

BlueCoDE: Bluetooth Coordination in Dense Environment for Better Coexistence

Weiping Sun[†], Jonghoe Koo[†], Seongho Byeon[†], Woojin Park[‡], Sangsoon Lim[‡], Daehyun Ban[‡], and Sunghyun Choi[†]

[†]Department of ECE and INMC, Seoul National University, Seoul, South Korea

[‡]Software R&D Center, Samsung Electronics, Seoul, South Korea

{weiping, jhkoo, shbyeon}@mwnl.snu.ac.kr, {woojin1.park, ssoon.lim, daehyun.ban}@samsung.com, schoi@snu.ac.kr

Abstract—Dense Wi-Fi and Bluetooth (BT) environments become increasingly common so that the coexistence issue between Wi-Fi and BT is imperative to solve. In this paper, we propose BlueCoDE, a coordination scheme for multiple neighboring BT piconets, to make them collision-free and less harmful to Wi-Fi. BlueCoDE reuses BT's existing PHY and MAC design, thus making it practically feasible. We implement a prototype of BlueCoDE on Ubertooth One platform and corroborate the performance gain via analysis, NS-3 simulations, and prototype-based experiments. Our experimental results show that with merely 10 legacy BT piconets, neighboring Wi-Fi network becomes useless achieving under 1 Mb/s throughput, while BlueCoDE enables the Wi-Fi throughput always remain above 12 Mb/s. We expect BlueCoDE to be a breakthrough solution for coexistence in dense Wi-Fi and BT environments.

I. INTRODUCTION

Cross-technology interference is emerging as a prominent problem at increasingly crowded 2.4 GHz Industrial, Science, and Medical (ISM) band. Wi-Fi and Bluetooth¹ (BT), the two most widely used technologies at 2.4 GHz ISM band, share the same spectra but are built on top of entirely different Physical (PHY) and Medium Access Control (MAC) layer design; Wi-Fi is Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)-based static channel technology [1] while BT is Time Division Multiple Access (TDMA)-based frequency hopping technology [2]. How to make them coexist better has been a major technical challenge since their introduction [3].

Highly-dense Wi-Fi and BT coexisting environments become increasingly common due to the accelerated deployments of Wi-Fi hotspots and the growing popularity of BT-enabled peripherals, e.g., increasing number of commuters enjoy audio streaming via BT headsets at a subway station covered by dozens of Wi-Fi networks. Accordingly, it is practically essential to deal with the coexistence problem in the highly-dense environments [4].

The severity of the heterogeneous interference between Wi-Fi and BT has been largely mitigated by BT's Adaptive Frequency Hopping (AFH) capability, which renders the heterogeneous interference harmless by dividing their respective

frequencies, i.e., BT hops within the BT channels free from Wi-Fi interference [2]. The majority of the previous work dealing with the problem of Wi-Fi and BT coexistence so far is to enhance the AFH capability by adding more intelligence and agility, especially to the channel classification and the collision avoidance mechanisms [5–10]. However, AFH is hardly effective in the environments where all the BT channels are thoroughly covered by a great number of highly-loaded Wi-Fi networks such that spectral separation is impossible. We refer to this scenario as *Wideband Heterogeneous* (WBH) environment.

In the environment where spectral separation between Wi-Fi and BT is feasible, homogeneous interference among multiple nearby BT devices becomes a salient problem. Normally, the homogeneous interference is largely alleviated by the respectively independent and pseudo-random hopping patterns. However, the spectral separation resulting from the AFH operation reduces the spectrum available for sharing among the multiple BT devices, thus giving rise to much severer homogeneous interference. We refer to this scenario as *Narrowband Homogeneous* (NBH) environment. Many approaches have been proposed to deal with the problem [11–13], while none of them is able to eliminate the homogeneous interference completely.

In this paper, we propose a light-weight coordination scheme for multiple nearby BT devices, called BlueCoDE, which addresses the above challenges. In BlueCoDE, a *coordinator* controls the hopping sequences² of multiple BT devices by simply manipulating their device addresses and clock values, while reusing BT's existing PHY and MAC. This leads them to a collision-free state in both WBH and NBH environments by making their hopping sequences *parallel* over time, i.e., each pair of the multiple BT devices always maintain constant non-zero offsets regarding their hopping channels over time.

Furthermore, BlueCoDE can make the multiple BT devices less harmful to the neighboring Wi-Fi networks in WBH environment, by locating their hopping channels as closely as possible so that, at a given moment, the range of the adversely

¹In this paper, Bluetooth means classic Bluetooth, which is widely used today for hands-free, speaker, mouse, etc., not Bluetooth Low Energy (BLE).

²Hopping sequence is a sequence of hopping channel numbers over time.

impacted Wi-Fi channels shrinks. Unlike the majority of the previous efforts, which try to reduce the number of collisions between Wi-Fi and BT, our goal is to reduce the wireless resources wasted in the collisions; we find out the root cause of the waste is the disparate channel bandwidths between Wi-Fi and BT. Note that our strategy is complementary rather than competitive with the previous efforts.

In fact, only individual BT devices are considered in the problem-solving by the existing efforts, and hence, multiple BT devices are not seen as a whole. To the best of our knowledge, we are the first to reveal and materialize the huge potential of the performance enhancement of Wi-Fi by the coordination among multiple neighboring BT devices. Moreover, to our best knowledge, we are the first to evaluate the performance of a Wi-Fi and BT coexistence scheme in a real wireless environment through proof-of-concept prototyping.

In summary, we claim the following four contributions.

- We find out a simple way to manipulate hopping sequences of multiple BT devices to make them collision-free.
- We develop a method to make multiple BT devices less harmful to Wi-Fi via hopping sequence manipulation, and identify the rationale analytically.
- We propose BlueCoDE as a general framework to exploit aforementioned method in NBH and WBH environments.
- We corroborate the performance gain delivered by BlueCoDE via simulation and prototype-based experiments.

The rest of this paper is organized as follows: in Section II, we discuss the related work, and in Section III, we provide a brief introduction to the BT operation and the coexistence problems addressed in this paper. In Section IV, we introduce the method used in hopping sequence manipulation, and in Section V, we give an overview of BlueCoDE. Two practical issues regarding BlueCoDE are addressed in Section VI. Then, the performance of BlueCoDE is evaluated via analysis, experiments, and simulation in Sections VII and VIII. Finally, we conclude the paper in Section IX.

II. RELATED WORK

Coexistence between Wi-Fi and BT: The adverse effect of BT piconets on the neighboring Wi-Fi networks is analyzed in [14]. The analytic results show that just a single fully-loaded interfering BT piconet reduces Wi-Fi throughput by 42%, emphasizing the vulnerability of Wi-Fi to the BT interference. The heterogeneous interference between Wi-Fi and BT can be mitigated by many algorithms [5–10, 15], which add more intelligence and agility to the spectral separation and traffic scheduling mechanisms. However, these algorithms consider only individual BT piconets, and pay no attention to an important question: *Is there room for further improvement when multiple BT piconets are involved?* We are the first to address this question with a positive answer. Note that the intelligence and agility of these algorithms can be easily integrated with BlueCoDE to achieve further improvement as long as they conform to the BT standard.

Coexistence among BT piconets: There are several algorithms proposed to deal with the packet collisions among

multiple nearby BT piconets, while non of them is able to completely eliminate the collisions. In [11], multiple delayed versions of hopping sequences are allocated to multiple BT piconets, respectively, while non-zero offsets among these delayed versions of hopping sequences are not guaranteed, thus still introducing collisions. The scheme proposed in [12] divides 79 BT channels into several orthogonal channel sets and let each piconet randomly select a set to use. Apparently, the collision probability increases among the piconets selecting the same channel set, which is not suitable in highly-dense environments. Moreover, the authors in [13] demonstrate that the packet collisions can be reduced by making the slot boundaries of the multiple BT piconets synchronized, while the reduction effect is far from satisfactory. We are the first to devise a scheme capable of completely eliminating the packet collisions among multiple nearby BT piconets.

Coordination: Coordination for wireless devices has been heavily investigated given that many advanced techniques such as Multi-User MIMO (MU-MIMO) are often based on tight clock synchronization among devices, e.g., see [16–18]. Moreover, coordination-based approaches are often considered an efficient way to address network heterogeneity. For example, Zhang *et al.* [19] propose to use a dedicated coordinator to emit busy-tone concurrently with normal Zigbee transmission to enhance the Zigbee’s visibility to Wi-Fi. They also propose to enable light-weight coordination between Wi-Fi and Zigbee through the gaps between consecutive energy pulses [20]. Therefore, as a coordination-based, fully feasible approach, BlueCoDE is a solution clearly worth consideration in both industry and research communities as the wireless technology evolves.

III. PRELIMINARY

A. Bluetooth Basics

A BT piconet is a group of BT devices, consisting of a *master* and up to seven *slaves*. A time slot is 625 μ s long, and a BT packet can only be transmitted at the start of a time slot. A BT packet can occupy one, three, or five time slots. In this work, we only consider one-slot packet, and the solutions for longer packets remain as future work.

Each BT device has its own clock implemented as a 28-bit binary counter (CLK) that increases by two every BT time slot. In a piconet, CLKs of all the slaves are synchronized with that of the master, and slot transitions occur whenever CLK₁³ of the master toggles between 0 and 1. Besides, each BT device has a unique 48-bit device address (ADDR) allocated by the manufacturer. The master’s ADDR is shared among all the slaves in the same piconet. In this paper, unless stated otherwise, CLK and ADDR of a piconet correspond to those of the master of the piconet.

Frequency hopping: There are 79 BT channels spaced 1 MHz apart and ordered from number 0 to 78. Hopping sequence is determined by *hop selection kernel*, consisting of a series of binary operations as shown in Fig. 1. In particular, when

³CLK_{*i*} indicates the *i*th bit of CLK consisting of CLK_{0–27}.

TABLE I
INPUTS OF HOP SELECTION KERNEL IN CONNECTION STATE.

Input	Value	Input	Value
X	CLK_{6-2}	Y1	CLK_1
Y2	$32 \times \text{CLK}_1$	A	$\text{ADDR}_{27-23} \oplus \text{CLK}_{25-21}$
B	ADDR_{22-19}	C	$\text{ADDR}_{8,6,4,2,0} \oplus \text{CLK}_{20-16}$
D	$\text{ADDR}_{18-10} \oplus \text{CLK}_{15-7}$	E	$\text{ADDR}_{13,11,9,7,5,3,1}$
F	$16 \times \text{CLK}_{27-7} \bmod 79$	F'	$16 \times \text{CLK}_{27-7} \bmod \Omega$

AFH is disabled, the hopping channel number is selected from *Basic Channel Table* (BCT), which has a unified format that contains all the 79 channel numbers, where the upper half contains the even numbered channels and the lower half contains the odd numbered channels, both of which are in the order of increasing channel number. When AFH is enabled, only (relatively) good channels are used to avoid interference. The channels used in AFH are referred to as *used channels*; otherwise, *unused channels*. When the channel obtained from BCT (indicated as CH_b in Fig. 1) is an unused channel, channel re-mapping takes place to select a used channel, CH_u , by replacing r_4 and BCT with r_5 and *Used Channel Table* (UCT), containing only the used channel numbers, respectively. Note that in connection state,⁴ CLK, ADDR, and the number of used channels (indicated as Ω) determine the inputs to the hop selection kernel and, consequently, the hopping sequence as shown in Table I.

B. Problems of Interest

In this work, we consider the following two coexistence problems.

- **WBH environment:** In this case, all the BT channels are used evenly in the long-term, even when AFH is exploited, since they suffer from similarly strong Wi-Fi interference and there are no obviously good BT channels among them. Both Wi-Fi and BT performance is evaluated.
- **NBH environment:** After spectral separation, only part of the BT channels are available for sharing among multiple nearby BT piconets. The performance of the multiple coexisting BT piconets is evaluated.⁵

IV. PARALLEL HOPPING SEQUENCES FOR COLLISION-FREE BT COEXISTENCE

In BlueCoDE, multiple piconets operate using the hopping sequences that are parallel to each other so that different piconets will never select the same channel simultaneously. By manipulating ADDR and CLK (cf. Table I), we can generate a group of mutually parallel hopping sequences with the legacy hop selection kernel.

A. Definition

We define that piconet a 's hopping sequence S^a and piconet b 's hopping sequence S^b are *parallel* if and only if the elements of S^a and S^b satisfy the condition that $(s_i^a - s_i^b) \bmod \Omega$

⁴In other states such as inquiry state, the inputs of the hop selection kernel are different from those in the connection state.

⁵We do not evaluate Wi-Fi performance in this case because a perfect spectral separation between Wi-Fi and BT is a premise.

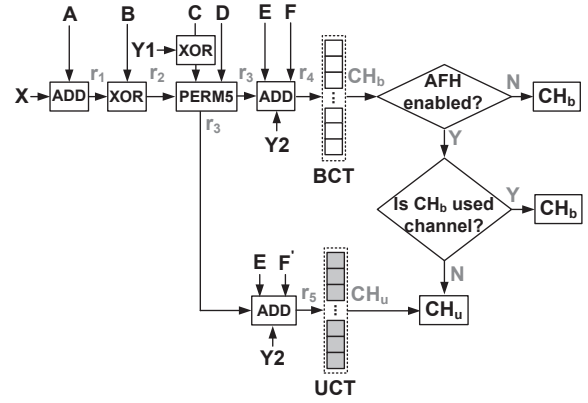


Fig. 1. Hop selection kernel.

is a non-zero ‘constant’ value regardless of the BT time slot index i . When AFH is not enabled, the number of used channels, Ω , is 79.

B. Conditions for Parallel Hopping Sequences

AFH-disabled case: In this case, the hopping sequences of multiple piconets are derived using a common BCT. To make each pair of the hopping sequences have a non-zero and constant offset over time, we let multiple piconets generate hopping sequences with the same inputs of hop selection kernel except \mathbf{E} , the time-invariant input of the last ADD operation (cf. Fig. 1). The conditions for parallel hopping sequences are listed as follows:

- 1) CLKs of multiple piconets should be synchronized.
- 2) ADDR₂₇₋₀ \ ADDR_{9,7,5,3,1} should be identical and ADDR_{9,7,5,3,1} should be different for multiple piconets.

Note that $\text{ADDR}_{13,11}$, which are also included in \mathbf{E} , are excluded in differentiation as they are also included in the derivation of \mathbf{D} .

AFH-enabled case: In this case, hopping channel can be generated either from BCT or UCT. This makes the situation more complicated and, consequently, two additional conditions are appended.

- 3) A common UCT should be used by multiple piconets.
- 4) In each BT time slot, all the piconets should use either BCT or UCT at the same time.

The necessity of the last condition arises from the fact that there is no guaranteed non-zero constant offset between the channels of a pair of piconets generated from BCT and UCT, respectively, even if all the first three conditions are satisfied. Note that the last condition cannot be satisfied all the time unless all the piconets are mandated to use only UCTs in AFH.⁶

Accordingly, we define two different modes of BlueCoDE: 1) Fully Standard-Compliant (FSC) mode, satisfying only the first three conditions and following the legacy hop selection kernel and 2) Almost Standard-Compliant (ASC) mode, in

⁶The role of BCT in AFH is to facilitate the slaves, which do not support AFH, to remain synchronized with the AFH-enabled master.

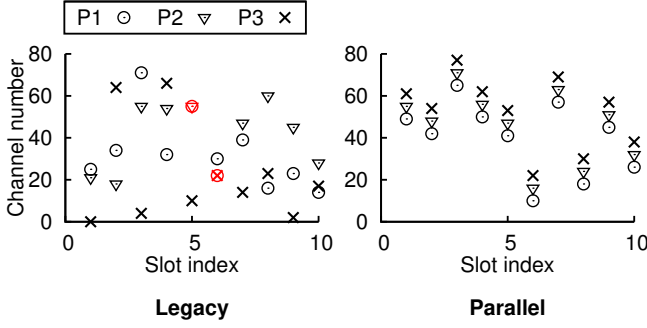


Fig. 2. Hopping sequence example.

which the use of BCT in AFH is prohibited to satisfy all of the four conditions.

In summary, if all the four conditions are satisfied, there are as many as $\min(2^5, \Omega)$ mutually parallel hopping sequences constrained by the number of different $\text{ADDR}_{9,7,5,3,1}$ values and the number of used channels, Ω . Fig. 2 shows the comparison between the legacy hopping sequences and the proposed parallel hopping sequences, emphasizing the collision-free property of the three mutually parallel hopping sequences generated by piconets P1, P2, and P3. Note that, in the case of parallel hopping sequences, the offsets between the channel numbers of each pair of piconets are congruent modulo Ω (79 in this case). This property is exploited by BlueCoDE to bond multiple piconets' hopping channels so that the resultant BT interference is less harmful to Wi-Fi compared with the legacy BT operation in WBH environment, as explained in Section VII.

V. BLUECODE OVERVIEW

A. Coordination

BlueCoDE coordinates multiple nearby piconets to make them satisfy the first three (in FSC mode) or four (in ASC mode) conditions for parallel hopping sequences. Accordingly, there should be a coordinator to control the CLKs, ADDRs, and UCTs of multiple piconets. The coordination procedure consists of the following four steps.

- Step 1. A master device will inform the coordinator of its identity information, i.e., its original device address.
- Step 2. Then, the coordinator replies with a list of control messages, i.e., a 28-bit temporal CLK (T_{CLK}), a 5-bit temporal $\text{ADDR}_{9,7,5,3,1}$ (T_{ADDR}),⁷ a UCT expressed as a 10-byte bitmap, where value 1 (0) indicates a used (unused) channel, and a 9-bit time offset (t_o)⁸ to indicate the time difference between the starting time of T_{CLK} and the moment when the control messages are received.
- Step 3. The master shares T_{CLK} and T_{ADDR} with the slave(s) in the same piconet and uses them to perform normal BT operations at t_o later.

⁷We assume that the other bits of ADDR have been shared among BT devices supporting BlueCoDE beforehand.

⁸ t_o is always less than the CLK resolution, i.e., 312.5 μs , such that 9-bit is sufficient to express it in the unit of μs .

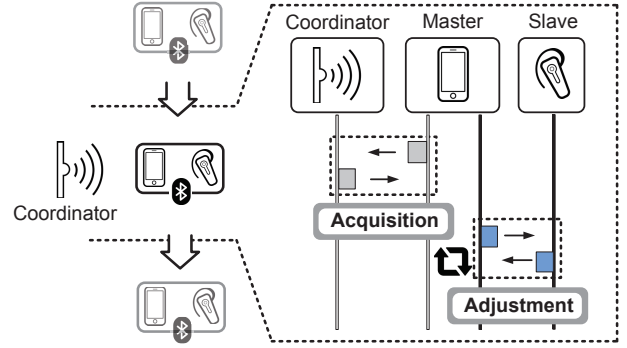


Fig. 3. Overview of BlueCoDE operations.

- Step 4. Repeat Steps 1 to 3 periodically with period τ to correct the synchronization errors due to the relative clock drift between the master and the coordinator (discussed in Section VI-B) and to confirm piconet's departure.

Besides, the coordinator has to detect Wi-Fi signals to determine whether it is in WBH or NBH environments. In a WBH environment, it will indicate all the BT channels as used channels in UCT, while in an NBH environment, it will deliver a common UCT containing only the BT channels free from Wi-Fi interference to multiple nearby piconets. Fig. 3 illustrates the coordination procedure, where Steps 1 and 2 are denoted as *acquisition* and Step 3 is denoted as *adjustment*.

B. Acquisition

Due to the fact that master device such as smartphone is normally equipped with multiple radio interfaces, e.g., LTE, Wi-Fi, and BT, various candidate technologies can be used to implement acquisition. While BlueCoDE is a general framework that can be applied to various scenarios, we here illustrate an example of the acquisition process under the assumption that the coordinator is a Wi-Fi AP and the master device is a Wi-Fi station equipped with Wi-Fi and BT combo chip, e.g., smartphone.⁹

Typically, Wi-Fi station (i.e., the master device in this example) is supposed to periodically conduct active scanning, i.e., exchange of probe request and probe response frames, to discover neighboring Wi-Fi APs [21]. The frame exchange herein can be used as a side channel to perform acquisition by embedding the identity information (involved in Step 1) and the control messages (involved in Step 2) into the vendor-specific fields¹⁰ of probe request and probe response frames, respectively. In particular, Wi-Fi AP (i.e., the coordinator) can disseminate T_{ADDR} and UCT to multiple neighboring Wi-Fi stations by simply embedding them in the probe response frames. On the other hand, the CLK synchronization is a non-

⁹This is the most common scenario, where BlueCoDE can be applied.

¹⁰Many Wi-Fi AP vendors implement proprietary features with vendor-specific Information Element (IE), which is defined in IEEE 802.11 standard to provide flexibility to the vendors for proprietary services. It can be easily embedded in probe request, probe response, and beacon frames by modifying driver codes of off-the-shelf Wi-Fi devices.

trivial task, and hence, we need to additionally incorporate the following two factors.

- 1) In most Wi-Fi and BT combo chips, such components as antenna (for 2.4 GHz radio), low noise amplifier, and power amplifier are shared by Wi-Fi (at 2.4 GHz) and BT modules, thus resulting in various TDMA-like solutions for the collocated coexistence problem, so that Wi-Fi and BT modules can alternately work [22–24]. These solutions are enabled by a dedicated internal communication link, e.g., shared register, bidirectional bus, etc., between them to share current status and timing information [22–24]. In particular, in the case of Qualcomm QCA6234 combo chip [23], the clock source of the BT is provided internally from the collocated Wi-Fi. Such trends will continue to make products more compact and cost-effective. Therefore, we argue that as long as multiple master devices' Wi-Fi clocks are time-synchronized, we easily achieve CLK synchronization by deriving the T_CLK and t_o internally based on collocated Wi-Fi clocks.
- 2) In practice, simply embedding T_CLK and t_o into vendor-specific field of probe response frame may cause unpredictable CLK synchronization errors among multiple piconets since the delay between the time that the vendor-specific field is embedded and the time that the probe response frame is transmitted over the air is random due to the nature of CSMA. To stringently achieve (microsecond-level accuracy) time synchronization, we can re-use Wi-Fi's *Timing Synchronization Function* (TSF) [1]. Each Wi-Fi AP should set the value of the *timestamp* of probe response frame (or beacon frame) to the value of its TSF timer—a 64-bit binary counter based on a 1-MHz clock with microsecond resolution—at the moment when the first bit of the timestamp is transmitted over the air, and each Wi-Fi station should update its local TSF timer based on the timestamp coming from the associated Wi-Fi AP; the received timestamp value should be adjusted by adding the Wi-Fi station's local processing delay passed since the first bit of the timestamp was received in order to achieve accurate time synchronization.¹¹ So, CLK synchronization can be achieved by deliberately letting multiple Wi-Fi stations associate with the coordinator or just overhear the timestamp value coming from the coordinator without association. Then, T_CLK and t_o are derived by $T_CLK = \lceil \frac{CLK_{tsf}}{312.5} \rceil$ and $t_o = \lfloor T_CLK \cdot 312.5 - CLK_{tsf} \rfloor$, respectively, where CLK_{tsf} is the timestamp value.

C. Adjustment

After the acquisition, the master needs to update its CLK, ADDR, and UCT to the new values. The UCT and CLK can be updated by *channel map update procedure* and *coarse clock adjustment procedure*, respectively, which are existing features in the BT standard. Regarding the ADDR, we have two options: 1) introduce a new protocol to enable ADDR

adjustment, or 2) remove all the paired slaves and re-establish the piconet. Apparently, the first option introduces less control overhead but necessitates a newly defined protocol, while the second option can be immediately supported by off-the-shelf BT products with extra control overhead. Either option can be used depending on the specific system requirement. Besides, if piconet has not been established yet, the master can carry out normal pairing procedure to set up a piconet based on the control messages obtained during the acquisition.

D. Piconet Management

In BlueCoDE, the coordinator carefully selects a proper $ADDR_{9,7,5,3,1}$ value for each piconet to make the overall BT interference less harmful to Wi-Fi in WBH environment. We conclude that when the hopping channels of the multiple piconets are more closely located in the frequency domain in each BT time slot, the resultant BT signals are less harmful to Wi-Fi, as further explained in Section VII. We also conclude that the channel offset between each pair of piconets should be at least 2 MHz to overcome the adjacent channel interference caused by in-band emission as explained in Section VI-A. As a result, in BlueCoDE, the coordinator allocates several consecutive $ADDR_{9,7,5,3,1}$ values to multiple piconets starting from all 0 sequence, and consequently, the hopping channels of the multiple piconets becoming an arithmetical progression with 2 MHz offset in each time slot due to the ordering rules of BCT and UCT described in Section III-A. Note that in NBH and WBH environments, the maximum number of piconets, which BlueCoDE can coordinate simultaneously, become $\min(2^5, \lfloor \frac{\Omega}{2} \rfloor)$ and $\min(2^5, \lfloor \frac{79}{2} \rfloor)$, respectively, due to the 2 MHz offset.¹²

In order to maintain the aforementioned hopping channel relations among nearby piconets, the coordinator should periodically inspect whether there is a new piconet that has not been coordinated or a piconet that has left the coordinator's coverage. When a new piconet joins, the next consecutive $ADDR_{9,7,5,3,1}$ is assigned to it. On the other hand, when a piconet has left, its $ADDR_{9,7,5,3,1}$ will be reused by the piconet¹³ currently with the largest $ADDR_{9,7,5,3,1}$ value so that no further ADDR adjustment is needed among other piconets.

VI. PRACTICAL ISSUES

A. In-band Emission

Two BT signals at different channels can severely interfere with each other due to the in-band spurious emission [2], which determines the minimal channel offset (i.e., the guard band) between each pair of the parallel hopping sequences to ensure interference-free condition in BlueCoDE.

The BT standard stipulates the interference rejection capabilities for each modulation type with respect to various types

¹¹On a commercial level, industry vendors assume the Wi-Fi TSF's synchronization to be within 25 μ s [25].

¹²We did not consider the case when the number of piconets exceeds the maximum supportable piconet number in both NBH and WBH environments, which remains as future work.

¹³The piconet will be informed of the new $ADDR_{9,7,5,3,1}$ through the next control messages received afterward.

TABLE II
INTERFERENCE REJECTION CAPABILITIES.

Interference Type	GFSK	$\pi/4$ -DQPSK	8DPSK	Ubertooth
Co-Channel	11 dB	13 dB	21 dB	10 dB
1 MHz offset	0 dB	0 dB	5 dB	0 dB
2 MHz offset	-30 dB	-30 dB	-25 dB	-20 dB
≥ 3 MHz offset	≤ -40 dB	≤ -40 dB	≤ -33 dB	≤ -41 dB

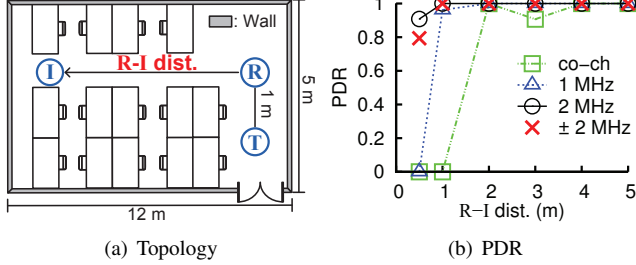


Fig. 4. In-band emission experiment topology and results.

of in-band interference as shown in Table II. Each value (in dB) indicates the minimum acceptable Signal-to-Interference Ratio (SIR) to meet the Bit Error Rate (BER) requirement (i.e., BER under 0.001) defined in the BT standard. In the cases of co-channel and 1 MHz offset interference, if the output power levels of a transmitter and an interferer are similar, the distance between the interferer and the receiver should be larger than that between the transmitter and the receiver to make SIR larger than 0 dB (cf. Table II). However, this condition cannot be always satisfied in dense environment. *Consequently, in BlueCoDE, we advocate the channel offset between each pair of piconets should be at least 2 MHz.*

We conducted a simple experiment using Ubertooth [26]—an open source BT platform equipped with cc2400 transceiver—to inspect the impacts of various types of in-band interference in the real world. The right-most column of Table II shows the minimum acceptable SIRs in the case of Ubertooth, which supports Gaussian Frequency Shift Keying (GFSK) only.

As illustrated in Fig. 4(a), the experiment was conducted in an office environment with three roles of Ubertooth devices, i.e., transmitter (indicated as ‘T’), receiver (indicated as ‘R’), and interferer (indicated as ‘I’). The distance between R and T is fixed at 1 m, analogous to the general case of BT usage, and the distance between R and I (R-I distance) varies from 0.5 m to 5 m. T transmits 1,000 BT packets to R with 0 dBm output power (i.e., the nominal output power of BT) at BT channel 39, (i.e., the channel in the midst, centered at 2.441 GHz), and Packet Delivery Rate (PDR) is measured at R. Each packet is 366-bit long without Forward Error Correction (FEC), analogous to an HV-3 packet format [2]. A packet is successfully delivered if all the bits of the packet are received correctly.

Interferer I generates GFSK-modulated pseudo-random binary sequence ceaselessly with 0 dBm output power at channels 37 (−2 MHz offset), 38 (−1 MHz offset), 39 (co-channel), 40 (+1 MHz offset), and 41 (+2 MHz offset), respectively, and the measured PDRs are averaged for each

absolute offset value. We also consider the case when two I’s located at the same position operate at channels 37 and 41 simultaneously (indicated as ± 2 MHz) to verify the performance of the piconet in BlueCoDE with two neighboring piconets always operating at the channels with −2 MHz and +2 MHz offsets, respectively.

Fig. 4(b) shows the PDR results. As expected, when I operates at co-channel and 1 MHz offset channel, R-I distances should be at least 2 m and 1 m for satisfactory PDR, respectively. On the other hand, if the channel offset is 2 MHz, even when there are two I’s, PDRs remain always fairly high regardless of the R-I distance, even though Ubertooth is supposed to perform much worse than off-the-shelf BT devices in this case.¹⁴

B. Clock Drift

Each BT device has its own clock supposed to advance with a clock rate of 3.2 kHz (cf. Section III-A). However, due to the imperfection of the hardware generating signal source such as crystal oscillator and the impact of the surrounding environments, e.g., temperature, humidity, etc., the clock rate fluctuates over time with a confined drift rate. Similarly, the coordinator’s clock also has its own drift rate. Considering the fact that the maximum clock synchronization errors between each pair of piconets should be less than the frequency hopping guard time I_S ¹⁵ (illustrated in Fig. 7) at the end of each BT time slot, all the masters should limit their clock synchronization errors under $\frac{I_S}{2}$ with respect to the coordinator’s clock. A simple algorithm is proposed to cope with the clock drift problem by adjusting the coordination period elaborated as follows.

- 1) Master m first uses the initial coordination period τ_0 for the first K coordination intervals. τ_0 is determined by the upper bound of the clock drift rate between the master and the coordinator. In the case of BT, the upper bound is 40 ppm [2], and hence, τ_0 becomes about 3.2 s.
- 2) When it receives control messages from the coordinator in the i th coordination interval, the relative clock drift rate between them during the i th interval is calculated as

$$r_d^{m,i} = \frac{|t_{\text{CLK}} \times (T_{\text{CLK}}^i - \text{CLK}^{m,i}) - (t_o^i + \Delta^i)|}{\tau_0} \quad (1)$$

where T_{CLK}^i , t_o^i , t_{CLK} , $\text{CLK}^{m,i}$, and Δ^i indicate T_{CLK} value in the i th control messages, t_o value in the i th control messages, CLK cycle (312.5 μ s), the CLK value of master m when receiving the i th control messages, and the actual elapsed time since the moment when the master’s clock becomes $\text{CLK}^{m,i}$, respectively.

- 3) After K intervals, the master calculates the average of the preceding K clock drift rates, \bar{r}_d^m , and the next coordination period is calculated as $\tau = I_S / (2\bar{r}_d^m)$.

¹⁴The minimum acceptable SIR of Ubertooth is higher than the three SIRs defined in the BT standard for 2 MHz offset case as shown in Table II.

¹⁵The guard time is at least 259 μ s in the case of one-slot packet.

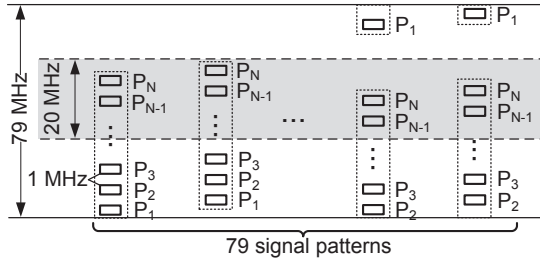


Fig. 5. BT and Wi-Fi overlapping patterns in BlueCoDE.

- 4) After that, the coordination period τ is updated whenever the master receives control messages based on the relative clock drift rate during the previous interval.

Note that the coordination period should be always upper bounded by the average sojourn time of a piconet (i.e., the time the piconet stays in coordinator's coverage) considering the mobility pattern in a specific venue.

VII. ANALYTICAL DISCUSSION

In this section, we analyze the performance gain delivered by BlueCoDE compared with the legacy scheme, i.e., the operation of each piconet is independent of the other piconets, in terms of the Wi-Fi performance in WBH environment and the BT performance in NBH environment.¹⁶ In the following analysis, for the sake of simplicity, we assume that 1) each piconet always transmits one-slot packet every BT time slot and 2) a Wi-Fi device in idle state detects an on-going BT transmission with probability 1 as long as the hopping channel of the BT transmission overlaps with the 20 MHz operating channel of the Wi-Fi device in the frequency domain.

A. Wi-Fi Performance in WBH Environment

The performance of Wi-Fi in WBH environment is mainly determined by two aspects: 1) Packet Error Rate (PER), representing how reliably an on-going Wi-Fi packet can be received by its receiver without error and 2) Channel Access Probability (CAP), representing how frequently a Wi-Fi device can grab the wireless medium to transmit its packet.

PER: We assume that a Wi-Fi packet is received in error if at least one BT packet collides with it in both time and frequency domains. Since, for a given Wi-Fi packet with a certain PHY rate, there is a fixed number of BT time slots overlapping with it in the time domain, we focus on the frequency domain performance.

With N piconets, the distribution of the number o_B of overlapping BT packets in the frequency domain during a BT time slot with BlueCoDE is derived by counting the number of overlapping BT packets for the 79 possible hopping patterns

¹⁶Since the BT performance in WBH environment is influenced by both Wi-Fi and BT devices in the neighborhood, it is difficult to derive a general analytic model for the analysis. We will evaluate it via experiments and simulation in Section VIII.

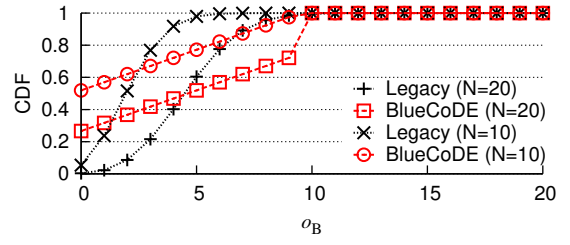


Fig. 6. CDF of the number of overlapping BT packets in a time slot.

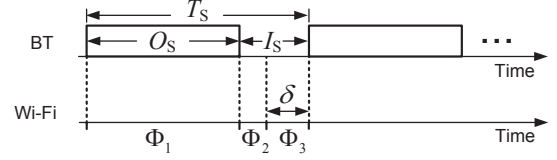


Fig. 7. The intervals for Wi-Fi's channel access within a BT time slot.

as illustrated in Fig. 5. It can be expressed with the following Probability Mass Function (PMF).

$$f_{bc}(o_B) = \begin{cases} \frac{61-2N}{79}, & o_B = 0, \\ \frac{4}{79}, & o_B = 1, 2, \dots, N-1, \\ \frac{22-2N}{79}, & o_B = N, \end{cases} \quad (2)$$

for $0 < N < 11$, and

$$f_{bc}(o_B) = \begin{cases} \max\left(\frac{61-2N}{79}, 0\right), & o_B = 0, \\ \frac{4}{79}, & o_B = 1, 2, \dots, 9, \\ \min\left(\frac{2N-18}{79}, \frac{43}{79}\right), & o_B = 10, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

for $11 \leq N \leq 32$.¹⁷ On the other hand, the distribution with the legacy scheme has the the following PMF.

$$f_l(o_B) = \binom{N}{o_B} \left(\frac{20}{79}\right)^{o_B} \left(\frac{59}{79}\right)^{N-o_B}, \quad o_B = 0, 1, \dots, N, \quad (4)$$

following a binomial distribution due to the independence among multiple piconets' hopping channels.

Fig. 6 shows the Cumulative Distribution Functions (CDFs) of o_B for $N = 10$ and 20 . In both cases, BlueCoDE yields clean 20 MHz spectra (i.e., $o_B = 0$) with higher probability compared with the legacy scheme. The cross points between the CDF curves of BlueCoDE and the legacy scheme explain the rationale behind the superiority of BlueCoDE; *it makes an already occupied Wi-Fi channel collide with more BT packets simultaneously, thus yielding more unoccupied Wi-Fi channels compared with the legacy scheme*. Finally, the PERs with BlueCoDE and the legacy scheme are $1 - f_{bc}(0)$ and $1 - f_l(0)$, respectively.

CAP: Assuming that the expectation of the Wi-Fi's channel access delay is given by δ ,¹⁸ CAP is defined as the probability

¹⁷The maximum number of piconets BlueCoDE can coordinate simultaneously in WBH environment is 32 as explained in Section V-D.

¹⁸For 802.11 Distributed Coordination Function (DCF), δ is $95.5 \mu s$, i.e., Distributed Inter Frame Space (DIFS) ($28 \mu s$) plus the expected initial random backoff time calculated as Wi-Fi slot time ($9 \mu s$) multiplied by the average initial backoff counter value (7.5).

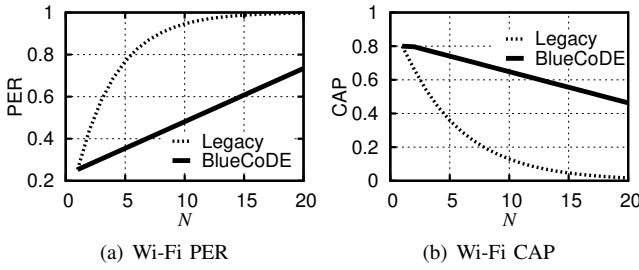


Fig. 8. Wi-Fi performance analysis in WBH environment.

that Wi-Fi channel is idle for at least δ since the arrival of a Wi-Fi packet, whose arrival time is uniformly distributed.

For the legacy scheme, CAP can be expressed as

$$CAP_l = \left(\frac{I_S - \delta}{T_S} + \frac{O_S + \delta}{T_S} \cdot \frac{59}{79} \right)^N, \quad (5)$$

where N , T_S , O_S , and I_S indicate the number of piconets, the duration of a BT time slot ($625 \mu s$), the actual transmission time of a one-slot BT packet, and the frequency hopping guard time at the end of each BT time slot, respectively. The base in (5) corresponds to the CAP when coexisting with a single piconet. The first term of the base indicates the probability that a Wi-Fi packet arrives during the time interval Φ_2 shown in Fig. 7, such that the corresponding Wi-Fi device can transmit the packet after δ with probability 1. Similarly, the second term is the probability that a Wi-Fi packet arrives during Φ_1 or Φ_3 , such that the corresponding Wi-Fi device can transmit the packet after δ with probability $\frac{59}{79}$. There is an exponent N since the operations of N piconets are independent of each other. Similarly, with BlueCoDE, the CAP becomes

$$CAP_{bc} = \frac{I_S - \delta}{T_S} + \frac{O_S + \delta}{T_S} \cdot f_{bc}(0). \quad (6)$$

Fig. 8 shows PER and CAP of Wi-Fi when coexisting with N piconets. We observe that BlueCoDE yields lower PER and higher CAP compared with the legacy scheme; the PER (CAP) with the legacy scheme grows (decays) rapidly and saturates, while it grows (decays) gradually with BlueCoDE, thus resulting in the expanding gaps between them until N becomes approximately 10.

While the analysis is rather simplistic, it essentially captures the rationale behind the superiority of BlueCoDE, capable of explaining the performance gains shown in Section VIII.

B. BT Performance in NBH Environment

As described in Section IV-B, in ASC mode, BlueCoDE eliminates collisions among multiple piconets completely in NBH environment. In FSC mode, the Collision Probability (CP), i.e., the probability that a piconet collides with one or more nearby piconets by selecting the same hopping channel,¹⁹

¹⁹In the analysis, we only consider the co-channel collisions.

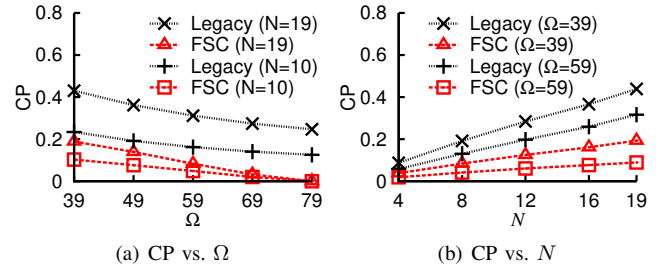


Fig. 9. CP comparison between BlueCoDE (in FSC mode) and the legacy scheme in NBH environment. Note that CP in ASC mode is always zero.

with the number Ω of used channels is

$$CP_{FSC} = 1 - \frac{\Omega}{79} \cdot \prod_{i=1}^{N-1} \left(1 - \max \left(\frac{80 - \Omega - i}{(79 - i)\Omega}, 0 \right) \right) - \left(\frac{79 - \Omega}{79} \right) \cdot \prod_{i=1}^{N-1} \left(1 - \max \left(\frac{\Omega - i + 1}{(79 - i)\Omega}, 0 \right) \right). \quad (7)$$

We also derive the CP with the legacy scheme by utilizing the statistical model proposed in [27]. Fig. 9 shows the CPs with the number Ω of used channels (Fig. 9(a)) and the number N of piconets (Fig. 9(b)), where the CPs with FSC mode are always much lower than those with the legacy scheme, due mainly to the considerable collision-free cases in FSC mode when all the piconets use either UCT or BCT at the same time.

VIII. EVALUATION

A. Evaluation Methodology

We implement a prototype of BlueCoDE on Ubertooth platform [26] and conduct experiments in an office environment (Fig. 10(a)). We make a Ubertooth device take the coordinator role.²⁰ At the beginning of each experiment, the other Ubertooth devices reside at a predefined BT channel (channel 39 in this case) and wait for a *trigger frame* transmitted by the coordinator, based on which the channel hopping operations of these Ubertooth devices are triggered simultaneously, thus achieving CLK synchronization as illustrated in Fig. 10(b), where $f(\cdot)$, $CLK(i)$, and $T_ADDR\{j\}$ indicate hopping sequence generation function, CLK value at the i th BT time slot since the trigger frame was received with initial value of T_CLK , and T_ADDR for the j th Ubertooth device, respectively. The control messages, i.e., T_CLK , T_ADDR , and UCT, are stored in advance in the firmware of each Ubertooth device.

We also conduct NS-3 simulations with random topologies to eliminate the correlation between the performance and network topologies. We add BT modules into NS-3 simulator [28], where mutual interference between Wi-Fi and BT is reflected based on the model proposed in [29]. In both the experiments and the simulations, we consider Wi-Fi's

²⁰Since the firmware of Wi-Fi and BT combo chips is proprietary, we are not authorized to modify them to achieve CLK synchronization in the way described in Section V-B.

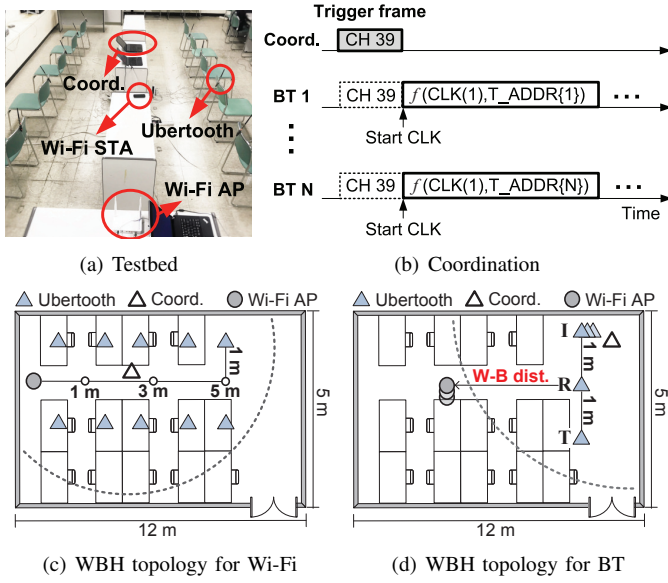


Fig. 10. Prototyping.

downlink traffic and fully-loaded BT traffic, unless stated otherwise.

B. Prototype-based Experiments

WBH experiment for Wi-Fi performance: Experiments are conducted under the topology shown in Fig. 10(c).²¹ A Wi-Fi station (Galaxy Nexus) receives fully-loaded UDP traffic from the associated Wi-Fi AP (ipTIME A2004NS-R), which controls PHY rates with an internal rate-adaptation algorithm. We also consider various representative BT traffic models, i.e., fully-loaded (1), a half-loaded (1/2), and a third-loaded (1/3) traffic models, which are analogous to the BT's SCO link with three different packet types (i.e., HV-1, HV-2, and HV-3), respectively.

Fig. 11(a) shows the application layer throughput of the Wi-Fi station at various distances to the AP in the cases when the neighboring Ubertooth devices run with the legacy scheme (indicated as 'L') and BlueCoDE (indicated as 'B'). The experiments are conducted in a clean wireless environment such that the Wi-Fi throughput is fairly high (up to 48 Mb/s) without BT interference. On the other hand, with BT interference, the Wi-Fi performance gets worse as the BT traffic load increases. We find BlueCoDE superior to the legacy scheme in all cases. In particular, with fully-loaded BT traffic (indicated as '1'), the Wi-Fi throughput becomes under 1 Mb/s with the legacy scheme as the distance increases, while BlueCoDE always provides throughput over 12 Mb/s. Fig. 11(b) shows the empirical CDF of the Modulation and Coding Scheme (MCS) indices²² of the Wi-Fi packets captured during the experiment.²³ It unfolds that with the legacy scheme, the AP

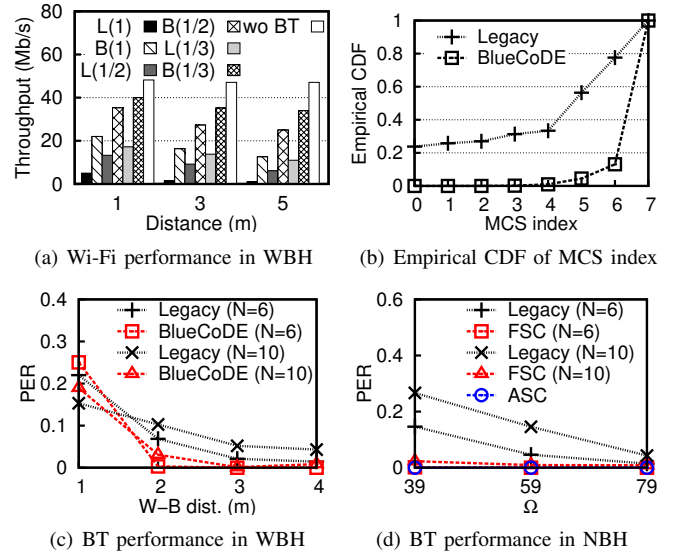


Fig. 11. Experimental results.

uses lower ordered MCS indices more frequently to deal with packet losses, thus further degrading the performance, since the lengthened Wi-Fi packet duration tends to incur more collisions with BT packets.

WBH experiment for BT performance: We employed another topology²⁴ for the evaluation of the BT performance in WBH environment as illustrated in Fig. 10(d), consisting of an Ubertooth transmitter (indicated as 'T'), an Ubertooth receiver (indicated as 'R') placed 1 m away from T, several collocated²⁵ Ubertooth interferers (indicated as 'I') placed 1 m away from R, and three collocated fully-loaded Wi-Fi APs at Wi-Fi channels 1, 6, and 11,²⁶ respectively.

Fig. 11(c) shows the PER of R, when the distance between the Wi-Fi APs and R (W-B distance) varies from 1 m to 4 m, with five ($N = 6$ including T) and nine ($N = 10$) I's. When W-B distance is 1 m, the Wi-Fi interference dominates the PER, and hence, the PER with BlueCoDE is higher than that with the legacy scheme due mainly to the fact that the higher CAP yielded by BlueCoDE (cf. Fig. 8(b)) allows more Wi-Fi packets to be transmitted, and hence, severer Wi-Fi interference is imposed to R. When W-B distance is larger than 1 m, the PERs with BlueCoDE become nearly zero while with the legacy scheme, PERs are still noticeable due to the collisions between T and I's. Note that if the APs are not always fully loaded, but have finite and fixed amount of data to transmit, e.g., stations download a certain number of files through the APs, R is affected less by Wi-Fi with BlueCoDE than with the legacy scheme regardless of W-B distances. This is because the legacy scheme will induce more Wi-Fi packet errors followed by several packet retransmissions with lower

²¹Although WBH environment needs several Wi-Fi networks to cover all the BT channels, we deploy a single Wi-Fi AP, since only the Wi-Fi performance, which is independent of the Wi-Fi networks operating at other non-overlapping Wi-Fi channels, is of our interest in this experiment.

²²Higher MCS index means higher but less reliable PHY rate.

²³Wi-Fi packets are captured by a laptop placed beside the AP.

²⁴We select a more densely deployed topology, where collisions between I's and T are exposed more explicitly.

²⁵In this section, 'collocated' means being placed at the same position.

²⁶The three non-overlapping Wi-Fi channels cover almost the entire BT channels.

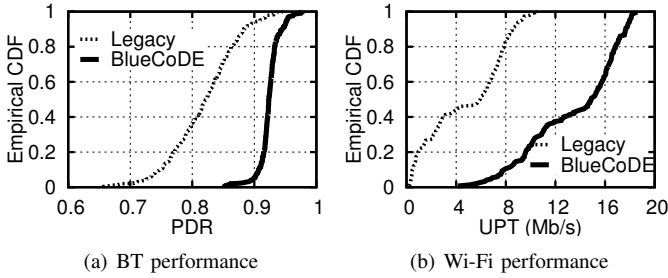


Fig. 12. NS-3 simulation results.

PHY rates,²⁷ thus incurring longer Wi-Fi channel occupancy time compared with BlueCoDE, as verified in Section VIII-C. **NBH experiment for BT performance:** The experiment was conducted under the topology similar to Fig. 10(d) except that there is no involved Wi-Fi AP, and the numbers of used BT channels, Ω , equal to 39, 59, and 79 were considered. The PERs of \mathbf{R} are shown in Fig. 11(d). As expected, BlueCoDE in FSC (ASC) reduces (eliminates) the collisions between \mathbf{T} and \mathbf{I} 's substantially (completely), thus always achieving much lower PERs than the legacy scheme.

C. NS-3 Simulation

Simulation setup: We conducted NS-3 simulations with 100 random topologies to evaluate Wi-Fi and BT performances in WBH environment. In each topology, a coordinator with BT interface, 10 BT piconets, three Wi-Fi APs at channels 1, 6, 11, respectively, and three Wi-Fi stations associated with the three Wi-Fi APs, respectively, are randomly deployed within a 20 m by 20 m rectangular region. Each piconet consists of a master and a slave located 1 m apart. During each simulation run, we let each Wi-Fi AP transfer a 0.5 MB file to its Wi-Fi station with link-adaptation algorithm enabled [30], and conduct the coordination in the same way as we do in the prototyping.

Figs. 12(a) and 12(b) show the empirical CDFs of the BT's PDR²⁸ and Wi-Fi's User Perceived Throughput (UPT) defined as the size of the file divided by the time to transfer it, respectively. The PDR CDF curve with BlueCoDE is densely concentrated—most piconets achieve PDR larger than 0.9, while the curve with the legacy scheme is relatively flat, ranging from 0.7 to 0.9, verifying that BT is less affected by Wi-Fi with BlueCoDE when the amount of Wi-Fi data is fixed. As expected, BlueCoDE outperforms the legacy scheme in terms of UPT as well.

IX. CONCLUDING REMARKS

In this paper, we propose BlueCoDE to mitigate the performance degradation of Wi-Fi and BT in WBH and NBH environments. The performance enhancements of BlueCoDE compared with the legacy scheme are evaluated through analysis, simulation and prototype-based experiments.

It should be noted that Wi-Fi's vulnerability to neighboring BT piconets is much severer than expected; from both analysis

and experimental results, we learn that with merely 10 BT piconets, neighboring Wi-Fi network becomes useless achieving under 1 Mb/s throughput. That reminds us of the necessity of the solution to mitigate the severity of the problem. We expect BlueCoDE to be a breakthrough solution for coexistence in dense Wi-Fi and BT environments.

As future work, we plan to improve BlueCoDE further by supporting multi-slot BT packets and developing topology-aware hopping sequence manipulation in multi-coordinator environments.

ACKNOWLEDGMENTS

This work was supported by Software R&D Center in Samsung Electronics Co., Ltd., and the National Research Foundation of Korea (NRF) grant funded by Korea government (MSIP) (NRF-2015R1A2A2A01006750).

REFERENCES

- [1] *IEEE 802.11-2012, Part 11: Wireless LAN medium access control (MAC) and physical Layer (PHY) specifications*, IEEE std., Mar. 2012.
- [2] *Specification of the Bluetooth system*, Bluetooth SIG, Dec. 2014.
- [3] J. Lansford *et al.*, "Wi-Fi (802.11b) and Bluetooth: enabling coexistence," *IEEE Network*, vol. 15, no. 5, pp. 20–27, 2001.
- [4] Plantronics, "A new look at Bluetooth density for the office: Expanded simulation results deliver real-world opportunities for audio deployments," *White Paper*, 2013.
- [5] J. Li and X. Liu, "A frequency diversity technique for interference mitigation in coexisting Bluetooth and WLAN," in *Proc. IEEE ICC*, 2007.
- [6] *IEEE 802.15.2, Coexistence of wireless personal area networks with other wireless devices operating in unlicensed frequency bands*, Aug. 2003.
- [7] A. Hsu *et al.*, "Enhanced adaptive frequency hopping for wireless personal area networks in a coexistence environment," in *Proc. IEEE GLOBECOM*, 2007.
- [8] C. Chiasserini and R. Rao, "Coexistence mechanisms for interference mitigation between IEEE 802.11 WLANs and Bluetooth," in *Proc. IEEE INFOCOM*, 2002.
- [9] J. So and Y. Kim, "Interference-aware frequency hopping for Bluetooth in crowded Wi-Fi networks," *Electronics Letters*, vol. 52, no. 17, pp. 1503–1505, 2016.
- [10] T. M. Taher *et al.*, "Development and quantitative analysis of an adaptive scheme for Bluetooth and Wi-Fi co-existence," in *Proc. IEEE CCNC*, 2009.
- [11] N. Amanquah and J. Dunlop, "Improved throughput by interference avoidance in co-located Bluetooth networks," in *Proc. IET EPMCC*, 2003.
- [12] Z. Jiang *et al.*, "Reducing collisions between Bluetooth piconets by orthogonal hop set partitioning," in *Proc. IEEE RAWCON*, 2003.
- [13] I. Ashraf *et al.*, "Time-synchronised multi-piconet Bluetooth environments," *IEEE Proc.-Comm.*, vol. 153, no. 3, pp. 445–452, 2006.
- [14] —, "Impact of interfering Bluetooth piconets on a collocated-persistent CSMA-based WLAN," *IEEE Trans. Veh. Technol.*, vol. 58, no. 9, pp. 4962–4975, 2009.
- [15] A. C. Hsu *et al.*, "Coexistence Wi-Fi MAC design for mitigating interference caused by collocated Bluetooth," *IEEE Trans. Comput.*, vol. 64, no. 2, pp. 342–352, 2015.
- [16] H. Qiu *et al.*, "High-rate WiFi broadcasting in crowded scenarios via lightweight coordination of multiple access points," in *Proc. ACM MobiHoc*, 2016.
- [17] S. Yun *et al.*, "Multi-point to multi-point MIMO in wireless LANs," in *Proc. IEEE INFOCOM*, 2013.
- [18] H. V. Balan *et al.*, "AirSync: Enabling distributed multiuser MIMO with full spatial multiplexing," *IEEE/ACM Trans. Netw.*, vol. 21, no. 6, pp. 1681–1695, 2013.
- [19] X. Zhang and K. G. Shin, "Cooperative carrier signaling: Harmonizing coexisting WPAN and WLAN devices," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 426–439, 2013.
- [20] X. Zhang and K. G. Shin, "Gap sense: Lightweight coordination of heterogeneous wireless devices," in *Proc. IEEE INFOCOM*, 2013.
- [21] X. Hu *et al.*, "Is there WiFi yet?: How aggressive probe requests deteriorate energy and throughput," in *Proc. ACM IMC*, 2015.
- [22] Texas Instruments, "WiLink 8 solutions," http://processors.wiki.ti.com/images/1/13/WiLink8__TI_Coexistence_Tutorial_1.pdf, 2013.
- [23] Qualcomm, "QCA6234 integrated dual-band 2x2 802.11n + Bluetooth 4.0," 2016.
- [24] Broadcom, "BCM4325 Bluetooth and WLAN Coexistence," *White paper*, 2008.
- [25] Wikipedia, "Timing Synchronization Function," https://en.wikipedia.org/wiki/Timing_Synchronization_Function, 2017.
- [26] D. Spill, "Project Ubertooth," <http://ubertooth.sourceforge.net/>, 2015.
- [27] T. Lin *et al.*, "An improved packet collision analysis for multi-Bluetooth piconets considering frequency-hopping guard time effect," *IEEE J. Sel. Area. Comm.*, vol. 22, no. 10, pp. 2087–2094, 2004.
- [28] "The NS-3 network simulator," <https://www.nsnam.org>, NS-3 Consortium, 2015.
- [29] D. Halperin *et al.*, "Predictable 802.11 packet delivery from wireless channel measurements," *Proc. ACM SIGCOMM*, 2010.
- [30] "Minstrel rate control algorithm," <https://wireless.wiki.kernel.org/en/developers/documentation/mac80211/ratecontrol/minstrel>, Linux Wireless, 2016.

²⁷In Wi-Fi, a packet error will be followed by several packet retransmissions with more robust but lower PHY rates until the packet is received correctly due to Automatic Repeat Request (ARQ) and rate-adaptation mechanisms.

²⁸A PDR of a piconet during a simulation run becomes a PDR sample.