

IoT テストベッドのための Bluetooth エミュレーション基盤の設計

湯村 翼^{1,2,a)} 明石 邦夫² 井上 朋哉^{1,2}

概要: 本論文では, IoT テストベッドのための Bluetooth エミュレーション基盤 BluMoon を提案する. Bluetooth では, Host と Controller は必ず Host Controller Interface(HCI) を通じて相互接続されるため, Psuedo Controller をソフトウェアで実装することで Bluetooth 機器の挙動を模擬する. 設計を評価するため, Bluetooth Low Energy の Advertise Packet を受信する Psuedo Controller と送信する Psuedo Beacon のプロトタイプ実装を行い, 提案手法が実装可能であることを確認した.

キーワード: Bluetooth, IoT, 無線ネットワーク, テストベッド

1. はじめに

ネットワークの普及と組込み機器の低価格化にともない, 様々な機器がネットワークに接続するようになった. この潮流は Internet of Things(IoT) と呼ばれる. IoT 機器の多くは無線通信機能を持ち, その主要な規格のひとつは Bluetooth[1] である.

Bluetooth は, PC とマウスの接続や, オーディオ機器とヘッドホンの音楽伝送など, 様々な用途に用いられる. 低消費電力で通信が可能な Bluetooth Low Energy(BLE) の登場以降, スマートフォンとその周辺機器との接続にも用いられる. また, 屋内測位のための Beacon としても用いられるようになった.

Bluetooth をはじめとする無線通信の検証を行うには, 多数の実機が必要となるが, 実機を用意するコストは大きい. さらに, 動作検証によって不具合が出た際, 不具合が引き起こされた条件の再現をしたいという要求があるが, 周囲の無線環境などの物理的要因が関連するため, 再現を行うことが非常に困難である.

本研究では, IoT テストベッドのための Bluetooth エミュレーション基盤 BluMoon を提案する. Bluetooth のテストプラットフォームでは, Bluetooth の Host および Application の実装がテスト環境でも同様に動作することが要求される. BluMoon では, Bluetooth Controller をハード

ウェアを用いずにソフトウェア実装による擬似 Controller に置き換えることでこの要求を実現する.

2. BluMoon

2.1 Bluetooth の構成

Bluetooth は, Host と Controller という階層構造をとる(図 1,2). Host と Controller の間は, Host-Controller Interface(HCI) というインタフェースが定義される. Bluetooth を利用するアプリケーションを開発する場合, 通常, Host の機能を利用し, 上位層にソフトウェアで実装される.

Host は, 用途に合わせてアプリケーション開発がしやすくなるよう, Generic Access Profile(GAP), Attribute Protocol(ATT), Generic Attribute Profile(GATT), Security Manager Protocol(SMP) といった様々なプロファイルやプロトコルが用意されている. また, Logical Link Control and Adaptation Protocol(L2CAP) という, リンクに関する制御を行うレイヤを持つなど, 複雑な構成となっている.

Controller には Bluetooth Basic Rate/Enhanced Data Rate(BR/EDR), Bluetooth Low Energy(LE), Alternate MAC/PHY(AMP) に対応したものがあり, それぞれ規格が異なる. BR/EDR は, Bluetooth バージョン 2.0 で採用され, クラシック Bluetooth と呼ばれることも多い. LE は Bluetooth バージョン 4.0 で採用された低消費電力のプロトコルである. AMP は, IEEE802.11 などの他の無線通信規格のハードウェアを Bluetooth Controller として用いるための規格である. BR/EDR と LE の 2 つの規格が Bluetooth の主要な規格で, 両方に対応するようデュアルスタック実装されている Controller も多い.

¹ 国立研究開発法人情報通信研究機構
NICT

² 北陸先端科学技術大学院大学
JAIST

a) yumu@nict.go.jp

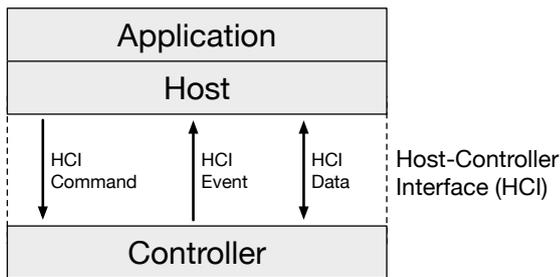


図 1 Bluetooth の階層構造と Host-Controller Interface.

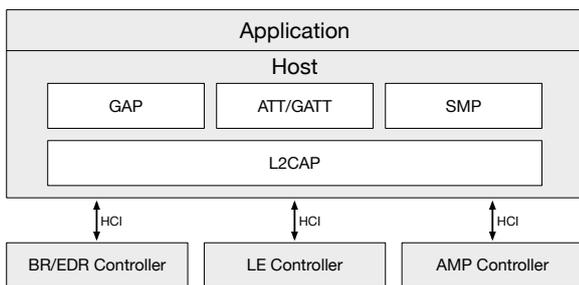


図 2 Bluetooth の構成概要.

HCI には、Host から Controller へ命令を送信する HCI Command, Controller から Host へ応答や通知を送信する HCI Event, データを双方向に送受信する HCI Data の 3 種類の通信方式がある。これらは、HCI Frame という形式で送受信される。

2.2 提案手法

Bluetooth では、Host と Controller は必ず HCI を経て相互接続される。Controller の挙動を模倣すれば、Controller やその先のリモートデバイスの挙動を自由に定義することができ、Host は HCI を介して通常の Bluetooth 通信と同等に扱うことができる。そこで、BluMoon では、ソフトウェアで実装された Psuedo Controller を設置する。

3. 設計

3.1 設計の概要

BluMoon では、Bluetooth Controller としてソフトウェア実装された Psuedo Controller を用いる。Host による Controller や外部機器に対する制御はすべて Psuedo Controller を経由する。そのため、模倣に関するあらゆる機能をすべて Psuedo Controller が担うこともできる。例えば、リモートのデバイスの振る舞いを Psuedo Controller の機能の一部として実装することができる。しかし、テストプラットフォームとして構築することを考えると、あらゆる機能を Psuedo Controller に実装する密結合な構成は、テ

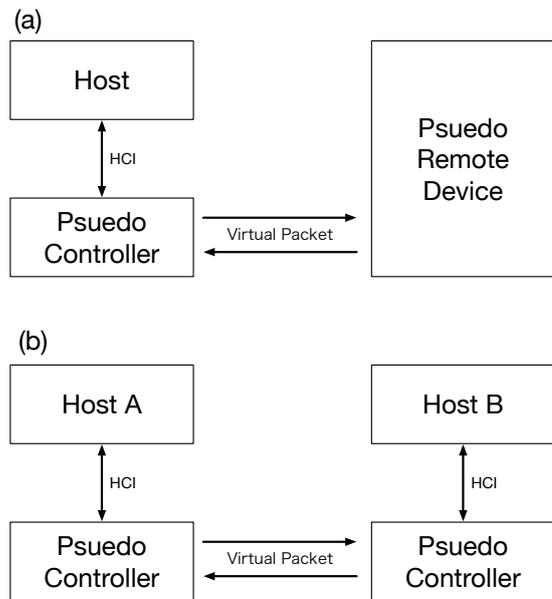


図 3 BluMoon の設計概要.

ストごとに異なる Psuedo Controller が必要となるので好ましくない。BluMoon では、リモートデバイスの振る舞いに関する機能は、Psuedo Remote Device が担う構成をとり、Psuedo Controller との間は Virtual Packet で通信を行う (図 3(a))。

また、Virtual Packet の仕様を定めることにより、Psuedo Controller 同士の通信も可能となる。これにより、複数の Host を用いたアプリケーションテストが可能となる (図 3(b))。

3.2 Psuedo Controller の設計

主要な Bluetooth Controller には BR/EDR と LE があるが、本論文では LE を対象として Psuedo Controller 設計 (図 4) について述べる。BR/EDR については述べないが、基本的な設計は共通である。

Psuedo Controller は、HCI Command を待ち受ける Command Receiver, HCI Event を送り出す Event Sender, Virtual Packet を待ち受ける Virtual Packet Receiver, Virtual Packet を送り出す Virtual Packet Receiver, これらの送受信を制御する Psuedo Controller Manager, 状態やパラメータを保存する Database にて構成される。

Command Receiver が Host からの HCI Command を受け取ると、Psuedo Controller Manager は Command を解析し、その内容に応じた処理を実行する。状態やパラメータのアップデートをとまなう場合は、新しい値をデータベースに書き込む。LE では、standby, advertising, scanning, initiating, connection の 5 つの状態を遷移する。

実際の Bluetooth Controller は無線電波によるパケットの送受信を行うが、BluMoon では、Psuedo Controller は

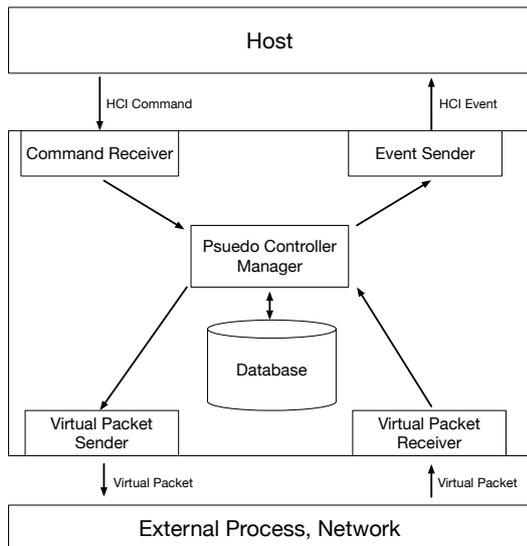


図 4 BluMoon の疑似 LE Controller の機能構成図。

プロセス間通信やネットワークを通じて Virtual Packet の送受信を行う。Psuedo Controller Manager は、状態とパラメータに応じて、Virtual Packet の送信を行う。

Virtual Packet Receiver が Virtual Packet を受け取ると、Psuedo Controller Manager は Virtual Packet を解析し、その内容に応じた処理を実行する。Virtual Packet の内容と、Database に保存された Psuedo Controller の状態とパラメータを照らし合わせ、HCI Event が発生する場合には HCI Event を生成して Event Sender を通じて Host へ送る。

3.3 Psuedo Remote Device の設計

Psuedo Remote Device は、Psuedo Controller の対向となる機器の挙動を模擬する。Psuedo Remote Device と Psuedo Controller のやりとりは Virtual Packet を通じて行うため、Psuedo Remote Device が行うことは Virtual Packet の生成と送信である。

送信する Virtual Packet の内容は、あらかじめシナリオ等の設定ファイルで定義したり、対向アプリケーションの挙動に応じて対応したりするようにする。例えば、一定周期で advertise を行う Psuedo Remote Device や、ペアリングのフローを模擬する Psuedo Remote Device などが実装できる。

3.4 Virtual Packet の設計

Virtual Packet は、IP ネットワークで送信し、BluMoon ヘッダとペイロードを格納する。宛先 IP アドレスは、Psuedo Controller や Psuedo Remote Device のものを指定する。ペイロードは、LE の場合 Link Layer のパケットフォーマットに従って記述する。Psuedo Controller が取

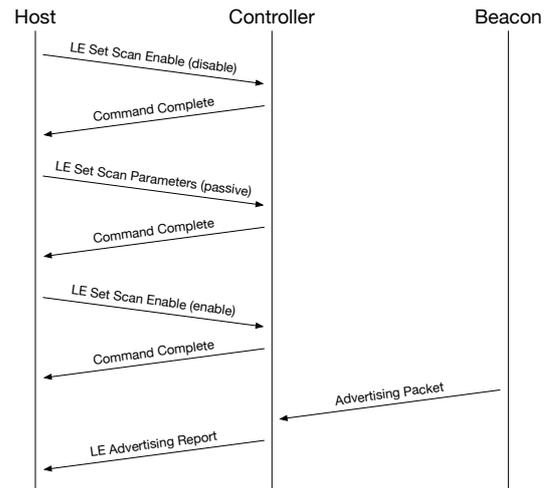


図 5 Advertise Packet を受信するための Passive Scan 設定のシーケンス

得できない値をエミュレーションで使用する際には、BluMoon ヘッダとして記述する。例えば、本来の Bluetooth では受信側の Controller で RSSI を計測するが、BluMoon では電波を用いないため、システムで指定する必要がある。

4. 実装

4.1 実装の概要

設計の妥当性を示すために、BLE Advertise Packet の送受信を行う Psuedo Controller と Psuedo Remote Device の実装を行った。これは BluMoon の一機能例であるが、Bluetooth では Host と Controller の両方が関係するすべての処理において HCI を経由するため、本実装は、他機能を実装する際の設計の妥当性も示すと考える。

BLE では、ペアリング前に Peripheral(周辺機器)が存在を周知するために信号を定期的送信する Advertise という機能がある。Advertise のために送る信号は Advertise Packet と呼ばれる。Advertise はペアリング以外にも、測位に用いたり URL などの情報の通知にも活用され、これらを目的に設置された機器は Beacon と呼ばれる。そのため、本研究で実装した Psuedo Remote Device は Psuedo Beacon と呼ぶ。

BLE Advertise Packet には、接続可否、スキャン可否、宛先指定の有無によって ADV_IND, ADV_NONCONN_IND, ADV_SCAN_IND, ADV_DIRECT_IND の 4 種類がある。本研究では、プロトタイプ実装のため、シーケンスが最も単純な ADV_NONCONN_IND を受信するものを実装した。

Psuedo Controller は、BLE Advertise Packet の Virtual Packet を受け付け、受信した際にその情報を HCI Event として Host へ伝達する。Advertise を受け付けるには、Controller に対して Passive Scan の設定が必要となる。Psuedo Beacon は、BLE の Advertise Packet を定期的に Virtual

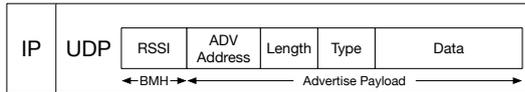


図 6 Psuedo Beacon が送信する仮想アドバタイズのフォーマット。BMH は BluMoon ヘッダ。

Packet として送信する。コマンドラインアプリケーションとして Python で実装した。RSSI と送信元 MAC アドレスは、コマンドの引数で指定して簡単に変更できるように実装した。Host, Controller, Beacon の送受信シーケンスを図 5 に示す。

これらの実装は、StarBED[2][3] で行った。StarBED の GroupN のノードに Ubuntu 16.04 LTS をインストールし、その上で実装した。

4.2 Psuedo Controller の実装

Psuedo Controller は、仮想 Bluetooth デバイスツールである btvirt を元に実装を行った。btvirt は、Linux の Bluetooth スタック BlueZ [4] で提供されるツール群に含まれる、オープンソースソフトウェアである。btvirt は、Bluetooth の仮想 Controller として振る舞い、HCI Command を受け付け、適切な HCI Event を返す。UNIX ドメインソケット経由で HCI Command とイベントの送受信を行い、USB や UART 等で接続された実機と同様に扱う事ができる。ただし、プロセス外からの命令を受け付ける機構は無く、HCI Event を任意のタイミングで発生させることはできない。本研究では、Psuedo Controller が BLE advertise の Virtual Packet を受信した際に、LE Advertising Report を Host に送る実装を行った。LE Advertising Report は、LEController が Advertise Packet を受信した際に Host に送られる HCI Event である。Virtual Packet 受信部は、UDP ソケットにより受信を待ち受ける。

4.3 Psuedo Beacon の実装

Virtual Packet のフォーマットに従って、UDP データグラムを送信するコマンドラインソフトウェアを Python で実装した。

本研究では、Virtual Packet の送信側である Psuedo Beacon が RSSI の値を指定し、BluMoon ヘッダとして格納する。RSSI は、Psuedo Controller が LE Advertising Report イベントを作成する際に利用される。

4.4 実装による設計評価

Psuedo Controller を起動し、Psuedo Beacon からの仮想 Advertise Packet を受信し、各種 Bluetooth コマンドツ

```
$ hciconfig
hci0:   Type: BR/EDR  Bus: VIRTUAL
        BD Address: 00:AA:01:00:00:23  ACL MTU: 192:1
SCO MTU: 0:0
        UP RUNNING
        RX bytes:0 acl:0 sco:0 events:56 errors:0
        TX bytes:1016 acl:0 sco:0 commands:42 errors:0

$ sudo hcitool lescan
LE Scan ...
AA:BB:CC:DD:EE:FF BLUMOONTEST

$ sudo hcidump
HCI sniffer - Bluetooth packet analyzer ver 5.37
device: hci0 snap_len: 1500 filter: 0xffffffffffffff
< HCI Command: LE Set Scan Enable (0x08|0x000c) plen 2
  value 0x00 (scanning disabled)
  filter duplicates 0x01 (enabled)
> HCI Event: Command Complete (0x0e) plen 4
  LE Set Scan Enable (0x08|0x000c) ncmd 1
  status 0x00
< HCI Command: LE Set Scan Parameters (0x08|0x000b) plen 7
  type 0x01 (active)
  interval 10.000ms window 10.000ms
  own address: 0x00 (Public) policy: All
> HCI Event: Command Complete (0x0e) plen 4
  LE Set Scan Parameters (0x08|0x000b) ncmd 1
  status 0x00
< HCI Command: LE Set Scan Enable (0x08|0x000c) plen 2
  value 0x01 (scanning enabled)
  filter duplicates 0x01 (enabled)
> HCI Event: Command Complete (0x0e) plen 4
  LE Set Scan Enable (0x08|0x000c) ncmd 1
  status 0x00
> HCI Event: LE Meta Event (0x3e) plen 25
  LE Advertising Report
    ADV_NONCONN_IND - Non connectable undirected advertising (3)
    bdaddr AA:BB:CC:DD:EE:FF (Public)
    Complete local name: 'BLUMOONTEST'
    RSSI: -50
```

図 7 Psuedo Controller を起動し、Psuedo Beacon から Virtual Advertise Packet を受信した際の、各種状態確認ツールの実行結果。

ルの実行して結果を確認した。コマンドはいずれも BlueZ のユーティリティツール群に含まれる。実行結果を図 7 に示す。

hciconfig は、BluetoothController の状態を確認したり、起動や終了などの命令を実行できるコマンドツールである。各 Controller は、hciX というインタフェースとして表示される。Psuedo Controller を実行するノードにはもともと BluetoothController は接続されていないが、Psuedo Controller を実行すると hci0 というインタフェースとして現れる。

hcitool は、HCI に関するユーティリティコマンドツ

ルで、Controller に対して様々な命令を実行できる。lescan オプションでは、LE Set Scan Parameters と LE Set Scan Enable の HCI Command を Controller に送信して、BLE Advertise Packet のパッシブスキャンを開始する。Advertise Packet を受信して LE Advertising Report イベントを受け取ると、その概要を標準出力に表示する。

hcidump は、Host と Controller 間でやりとりする HCI フレームをキャプチャするコマンドツールである。hcitool lescan を実行した際に、HCI Command が送信され、その応答として Psuedo Controller が Command Complete イベント送信している手順を追うことが出来る。Virtual Packet を受信した際は、Psuedo Controller が LE Advertising Report イベントを送信する。hcitool と hcidump で表示される bdaddr, Complete local name, RSSI は Psuedo Beacon が指定し Virtual Advertise Packet に含まれている情報である。

5. 関連研究

5.1 Bluetooth エミュレータ

これまで、様々な手法で Bluetooth 通信の仮想化や Bluetooth 機器の模擬が行われてきた。

本論文では、Bluetooth のパケット情報を UDP データグラムとして送受信する方法を実装した。遠隔地にある機器同士で Bluetooth 通信を行うことを目的として、ネットワークを通じて Bluetooth の情報を送受信する方法がいくつか存在する。井波と丹 [5] は、ゲートウェイを設置して HCI Event 及び HCI Data を有線接続で転送する手法を提案した。Tsuda *et al.* [6] ではゲートウェイの機能を操作端末に組み込み、HCI メッセージを転送する手法を提案した。この手法は、Okada and Suzuki [7] で BLE に対応するよう拡張された。UbiGate [8] は、Session Initiation Protocol (SIP) を利用し、Bluetooth の拡張を行った。Service Discovery Protocol (SDP) で得た機器情報は、ゲートウェイを通じて SIP のプレゼンス情報に変換して管理される。UbiGate の機能を用い、Personal Area Network (PAN) 同士を接続するサービスとして UbiPAN [9] も考案された。

また、システム検証のために Bluetooth 機器の動作を模擬するものもある。BlueMonarch [10] は、Bluetooth アプリケーションをテストするためのシステムである。OBEX のファイル送信を、ペアリング不要な SDP 通信を用いて模擬するというもので、コンテンツ配信システムなどの評価を行う。Bluetooth アプリケーションの検証や評価を行うという目的は本研究に似ているが、ソフトウェアによる仮想化ではなく実際に電波を出すものであり、本研究とは対象が異なる。BLE Peripheral Simulator [11] は、Web Bluetooth の機能を試すための Android アプリである。バッテリー、心拍数、体温計の 3 種類のセンサ機器の動作を模倣し、BLE の peripheral 機器としてふるまう。

5.2 無線ネットワークエミュレータ

Bluetooth 以外の無線ネットワークプロトコルを対象とした無線ネットワークエミュレータも多数存在する。無線ネットワークエミュレータは、ソフトウェアによって実現するものと、専用ハードウェアによって実現するものがある。

ソフトウェアによる無線ネットワークエミュレータ QOMET [12] は、実世界のイベントシナリオから無線の特性を記述した deltaQ という設定ファイルを生成し、有線ネットワークでの通信に無線通信の特性を注入する無線ネットワークエミュレータである。QOMET は IEEE802.11 のエミュレーションを行うことを想定して設計された。QOMET をベースにして無線ネットワークエミュレーションテストベッド QOMB [13] や、動的ネットワークエミュレーションツール DynamiQ [14] も提案された。NETorium [15] は、大規模ネットワークを想定したソフトウェア無線ネットワークエミュレータである。NETorium は、Asteroid と Meteor によって構成される。Asteroid が仮想インタフェースを提供し、Meteor [16] が無線の特性を注入する。

一方、専用ハードウェアによるエミュレータには、Network Emulator II [17] などがある。専用ハードウェアによるエミュレータは、無線特性を高精度で注入できるが、専用ハードウェアを要するため実験環境のスケールアウトが難しい。

5.3 IoT テストベッド

大規模なシステム開発などで用いられる、実際の運用環境に近づけた試験用プラットフォームは、テストベッドと総称される。テストベッドの中でも、IoT に関するものは IoT テストベッドと位置付けられ、その事例は Gluhak *et al.* [18] にまとめられる。

IoT テストベッドの形態のひとつに、多数の無線ネットワークノードが並べられ、それらのノードを制御して実験を行う無線ネットワーク型の IoT テストベッドがある。この型のテストベッドには ORBIT [19], Kansei [20], KanseiGenie [21], WISEBED [22], SensLab [23] といったものがあり、各テストベッドでは 100~1000 程度の無線ノードが並べられる。

本研究で対象とした Bluetooth に関するテストベッドは、SeamBlue [24] の評価のために用いられたものがある。SeamBlue は BLE 接続をハンドオーバーする研究で、その評価のために、建物内にスマートフォンを配置して Bluetooth テストベッドが構築された。

また、実際の街中を使った実証実験環境も、広義の IoT テストベッドとして扱われる。この型のテストベッドには、OULU Smart City [25] や SmartSantander [26] がある。

6. 議論と今後の課題

6.1 テストベッドとしての効果

Bluetooth が活用される用途のひとつに、BLE Advertise を用いた測位システムがある。測位システムを導入する際には、正確な測位を行うためにビーコン設置位置を詳細に調整する必要があり、その作業には大きな労力がかかる。本研究の Bluetooth エミュレーション基盤を用いることにより、調整作業のコスト削減が期待できる。

また、スマートフォンと周辺機器の通信も、Bluetooth の用途として大きな割合を占める。Android エミュレータを用いたサービス検証基盤 [27] に本提案システムを組み込むことで、検証できる項目が増え、検証基盤の価値が高まると考えられる。Android では、バージョン 4.1 までは Bluetooth スタックとして本研究の実装でも用いた BlueZ が使用されていた。バージョン 4.2 以降は Bluedroid という Bluetooth スタックが用いられている。いずれのスタックもオープンソースであり、追加実装が可能である。

Web ブラウザから Bluetooth の通信制御を行う Web Bluetooth [28] という規格も存在する。PC 版 Chrome でも、2017 年 1 月にリリースされたバージョン 56 から Web Bluetooth API が正式にサポートされ、スマートフォン以外の機器からの接続が増えることが見込まれる。

6.2 電波伝搬シミュレーションとの連携

本論文では、無線環境の物理的要因の注入として、Advertise Packet に書き込まれる RSSI の値を任意に設定できるようにした。RSSI の値は、BLE での測位を行うための近接プロフィールに用いられる。RSSI は機器間の距離や遮蔽物に応じて変化するため、測位システムの検証を行う際にはより正確な RSSI の値が必要となる。機器の位置や遮蔽物を定義して電波伝搬シミュレーションを行うことで、正確な RSSI の値を注入し、より詳細な検証を行うことができると考えられる。また、より詳細なエミュレーションを実施するには、物理的要因の注入は、RSSI に限らずパケットの遅延やロス率にも反映されることが望ましい。

6.3 他の無線通信プロトコルへの適用

本研究では、疑似ハードウェア機器をソフトウェア実装することで、Bluetooth エミュレーション基盤を構築した。本論文では、この手法を Psuedo Adapter Method と呼ぶ。

本研究で対象とした Bluetooth は、Host と Controller が HCI というインタフェースを通じて接続される。Host は主にソフトウェアとして実装され、Controller は主にハードウェアデバイスとして実装される。そのため、Controller が疑似アダプタの対象であることがわかりやすく、Psuedo Adapter Method を適用しやすい。

OSI 参照モデルでは、データリンク層がハードウェアとロジックの境界層として扱われる。さらにデータリンク層は MAC 副層と LLC 副層に分かれる。Psuedo Adapter としてソフトウェア実装する対象は、MAC 副層とその下層の物理層が適当と考えられる。

7. おわりに

本論文では、IoT テストベッドのための Bluetooth エミュレーション基盤 BluMoon を提案した。Bluetooth は Host と Controller が必ず HCI を通じて相互接続されるため、BluMoon では、Psuedo Controller をソフトウェアで実装することで Bluetooth 機器の挙動を模擬した。設計を評価するため、BLE Advertise Packet を受信する Psuedo Controller と送信する Psuedo Beacon のプロトタイプ実装を行った。仮想 Advertise Packet の送受信における、Psuedo Controller との HCI Command とイベントのシーケンスを確認し、提案手法の実現性を示した。今後は、電波伝搬シミュレーションと連携するなど、より詳細なエミュレーション基盤の構築を検討する。

参考文献

- [1] Bluetooth: <https://www.bluetooth.com/>.
- [2] StarBED: <http://starbed.nict.go.jp/>.
- [3] Miyachi, T., Nakagawa, T., Chinen, K.-i., Miwa, S. and Shinoda, Y.: StarBED and SpringOS architectures and their performance, *International Conference on Testbeds and Research Infrastructures*, Springer, pp. 43–58 (2011).
- [4] BlueZ: <http://www.bluez.org/>.
- [5] 井波正朗, 丹 康雄: Bluetooth ネットワークの有線拡張方式に関する検討, 電子情報通信学会技術研究報告. CS, 通信方式, Vol. 103, No. 415, pp. 47–52 (2003).
- [6] Tsuda, K., Suzuki, H., Asahi, K. and Watanabe, A.: Proposal for a Seamless Connection Method for Remotely Located Bluetooth Devices, pp. 78–79 (2014).
- [7] Okada, M. and Suzuki, H.: Implementation of seamless connection system for Bluetooth Low Energy devices in remote locations, *2017 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 344–345 (2017).
- [8] Bissyandé, T. F. d., Réveillère, L. and Bromberg, Y.-D.: Ubigate: A gateway to transform discovery information into presence information, *Proceedings of the 4th international workshop on Services integration in pervasive environments*, ACM, pp. 19–24 (2009).
- [9] Albert, J., Bissyandé, T. F., Bromberg, Y.-D., Chaumette, S. and Réveillère, L.: Ubipan: A bluetooth extended personal area network, *Complex, Intelligent and Software Intensive Systems (CISIS), 2010 International Conference on*, IEEE, pp. 774–778 (2010).
- [10] Smith, T., Saroiu, S. and Wolman, A.: BlueMonarch: A System for Evaluating Bluetooth Applications in the Wild, *Proceedings of the 7th international conference on Mobile systems, applications, and services*, pp. 41–54 (2009).
- [11] BLE Peripheral Simulator: <https://github.com/WebBluetoothCG/ble-test-peripheral-android/>.

- [12] Beuran, R., Nakata, J., Okada, T., Nguyen, L. T., Tan, Y. and Shinoda, Y.: A multi-purpose wireless network emulator: QOMET, *Proceedings - International Conference on Advanced Information Networking and Applications, AINA*, pp. 223–228 (2008).
- [13] Beuran, R., Nguyen, L. T., Miyachi, T., Nakata, J., Chinen, K. I., Tan, Y. and Shinoda, Y.: QOMB: A wireless network emulation testbed, *GLOBECOM - IEEE Global Telecommunications Conference* (2009).
- [14] Beuran, R., Yasuda, S., Inoue, T., Takano, Y., Miyachi, T. and Shinoda, Y.: DynamiQ: A Tool for Dynamic Emulation of Networks., *EAI Endorsed Trans. Self-Adaptive Systems*, Vol. 1, No. 2, p. e1 (2015).
- [15] Akashi, K., Inoue, T., Yasuda, S., Takano, Y. and Shinoda, Y.: NETorium: high-fidelity scalable wireless network emulator, *Proceedings of the 12th Asian Internet Engineering Conference*, ACM, pp. 25–32 (2016).
- [16] 明石邦夫, 井上朋哉, ラズバンベウラン, 篠田陽一: Meteor: 大規模ネットワーク実験環境における無線ネットワークエミュレータの設計と実装, *電子情報通信学会論文誌 B*, Vol. 98, No. 4, pp. 357–372 (2015).
- [17] ixia: Network Emulator II, <https://www.ixiacom.com/products/network-emulator-ii/>.
- [18] Gluhak, A., Krco, S., Nati, M., Pfisterer, D., Mitton, N., Razafindralambo, T. and Alexander Gluhak, Srdjan Krco, Michele Nati, Dennis Pfisterer, Nathalie Mitton, T. R.: A Survey on Facilities for Experimental Internet of Things Research, *IEEE Communications Magazine*, Vol. 49, No. November, pp. 58–67 (2011).
- [19] Raychaudhuri, D., Seskar, I., Ott, M., Gamu, S., Ramachandran, K., Kremos, H., Siracusa, R., Liu, H. and Singh, M.: Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols, *Wireless Communications and Networking Conference, 2005 IEEE*, Vol. 3, IEEE, pp. 1664–1669 (2005).
- [20] Ertin, E., Arora, A., Ramnath, R., Nesterenko, M., Naik, V., Bapat, S., Kulathumani, V., Sridharan, M., Zhang, H. and Cao, H.: Kansei: a testbed for sensing at scale, *2006 5th International Conference on Information Processing in Sensor Networks (IPSN 2006)*, No. ii, pp. 399–406 (2006).
- [21] Sridharan, M., Zeng, W., Leal, W., Ju, X., Ramnath, R., Zhang, H. and Arora, A.: Kanseigenie: Software infrastructure for resource management and programmability of wireless sensor network fabrics, *Next Generation Internet Architectures and Protocols*, pp. 275–300 (2010).
- [22] Chatzigiannakis, I., Fischer, S., Koninis, C., Mylonas, G. and Pfisterer, D.: WISEBED: an open large-scale wireless sensor network testbed, *International Conference on Sensor Applications, Experimentation and Logistics*, Springer, pp. 68–87 (2009).
- [23] Burin Des Rosiers, C., Chelius, G., Fleury, E., Fraboulet, A., Gallais, A., Mitton, N. and Noël, T.: SensLAB Very Large Scale Open Wireless Sensor Network Testbed, *Proc. 7th International ICST Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCOM)* (2011).
- [24] Hussain, S. R., Mehnaz, S., Nirjon, S. and Bertino, E.: SeamBlue: Seamless Bluetooth Low Energy Connection Migration for Unmodified IoT Devices, *Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks*, USA, Junction Publishing, pp. 132–143 (2017).
- [25] Ojala, T.: Open urban testbed for ubiquitous computing, *2010 WRI International Conference on Communications and Mobile Computing, CMC 2010*, Vol. 1, pp. 442–447 (2010).
- [26] Sanchez, L., Muñoz, L., Galache, J. A., Sotres, P., Santana, J. R., Gutierrez, V., Ramdhany, R., Gluhak, A., Krco, S., Theodoridis, E. and Pfisterer, D.: SmartSantander: IoT experimentation over a smart city testbed, *Computer Networks*, Vol. 61, No. January 2013, pp. 217–238 (2014).
- [27] 三輪信介: Gigandroid: 多数のスマートフォンを対象としたサービス検証基盤の提案, 第17回インターネットテクノロジーワークショップ(WIT2016) (2016).
- [28] Web Bluetooth: <https://webbluetoothcg.github.io/webbluetooth/>.