

博士論文

人間とコンピュータと物理の
インタラクションを統合的に取り扱う
Cyber-Physical-Human Interaction の研究

湯村 翼

主指導教員 丹 康雄

北陸先端科学技術大学院大学
情報科学研究科

2019 年 9 月

概要

人間とコンピュータのインタラクションのあり方は長年研究され、Human-Computer Interaction(HCI) と呼ばれる一分野として体系的にまとめられる。人とコンピュータのインタラクションの単純なモデルとして入出力という概念が用いられ、情報のやりとりは入出力装置を経由して行う。代表的な研究成果であるディスプレイ、マウス、GUI といった発明は、現在に至るまで広く活用されている。当初は単純な機構であった入出力機器は、深度センサ、ウェアラブルデバイス、3次元表示ディスプレイなど、様々な目的と仕組みの機器が登場している。また、精度向上やコスト削減のため、同じ目的でありながら異なる物理量を計測する機器も登場している。一方、コンピュータの世界はインターネット上に構築された情報空間に閉じず、物理世界の影響を考慮した Cyber-Physical System(CPS) として研究が進められてきた。組み込み機器の制御により、センサが物理量を計測し、アクチュエータが物理空間に影響を及ぼすという、人間が直接的には介在しないフィードバックループを形成する。

このように発展したコンピュータの世界において、物理空間における影響は無視できない。そこで、人間とコンピュータと物理のインタラクションを統合的に取り扱うために、Cyber-Physical-Human Interaction(CPHI) という新たな概念を提唱する。本論文では、CPHI のパターンとして Cyberization from Human (人間からのサイバー化), Realization to Human (人間への意識化), Human-Physical Merging(人間と物理の情報融合), Physical Augmentation (物理的拡張) の4項目を提案する。CPHI について考察するため、SleepTyping, PICALA, Augmented Typing, BluMoon, AOBAKO という5つの研究を行い、CPHI における位置づけを確認した。

本論文では、CPHI の概念をまとめ、物理現象を模倣するシミュレーションおよびエミュレーションを利用することでテストプラットフォームを構築することを提案した。物理現象の模倣部をモジュラブルで入れ替え可能な設計としたため、模倣する物理モデルをより高精細なものに置き換えることによって要件に合わせた物理空間の模倣が可能である。物理モデルを可換にする設計は、CPHI 開発支援プラットフォームの設計の普遍的な考え方として役立つと考える。

目次

第 1 章	序論	1
1.1	Human-Computer Interaction	1
1.2	Cyber Physical System	3
1.3	Cyber-Physical-Human Interaction	3
1.4	本論文の構成	4
第 2 章	Cyber-Physical-Human Interaction	5
2.1	関連する概念	5
2.2	Cyber-Physical-Human Interaction の提案	7
2.3	physical を考慮したシステム開発	10
2.4	本論文における CPHI の位置づけ	11
第 I 部	Cyber-Physical-Human Interaction の インタフェース	18
第 3 章	人間の行動取得と解像度: Sleep Typing	19
3.1	はじめに	19
3.2	提案手法	20
3.3	評価	23
3.4	考察	27
3.5	おわりに	27
第 4 章	人間の感情の情報化と可視化による周囲への伝達: PICALA	28
4.1	はじめに	28
4.2	設計と実装	29
4.3	実証実験	33
4.4	ログ解析	35
4.5	主観評価	36
4.6	考察	38
4.7	関連研究	38

4.8	おわりに	39
第5章	物理空間への重畳演出による人間の感情への干渉: Augmented Typing	41
5.1	はじめに	41
5.2	関連研究	42
5.3	Augmented Typing	43
5.4	実装	44
5.5	評価実験	47
5.6	デモンストレーション	54
5.7	議論	55
5.8	おわりに	57
第II部 Cyber-Physical-Human Interaction の システム開発		58
第6章	無線通信の仮想化: BluMoon	59
6.1	はじめに	59
6.2	Bluetooth Low Energy	61
6.3	関連研究	64
6.4	BluMoon の設計方針	67
6.5	BluMoon の設計と実装	70
6.6	評価	73
6.7	ケーススタディ	79
6.8	おわりに	82
第7章	エミュレーションとその物理化によるテストプラットフォーム: AOBAKO	84
7.1	はじめに	84
7.2	AOBAKO	85
7.3	ビーコンの物理化	87
7.4	まとめ	91
第8章	結論	92
8.1	議論	92
8.2	おわりに	92
謝辞		94
参考文献		95

目次

1.1	Sketchpad. 画像は文献 [2] より引用.	2
1.2	oN-Line Systems. 画像は文献 [4] より引用.	2
1.3	Dynabook のコンセプト図. 画像は文献 [14] より引用.	2
1.4	iPhone. 画像は Web サイト [9] より引用.	2
1.5	Kinect. 画像は文献 [12] より引用.	2
1.6	Playstation VR. 画像は Web サイト [13] より引用.	2
1.7	本論文の構成.	4
2.1	ラディカル・アトムズのコセプト. 図は文献 [27] より引用.	6
2.2	Human-in-the-Loop Cyber-Physical Systems の概念図. 図は文献 [31] より引用.	8
2.3	Cyber-Physical Human Systems の Human Service Capability Description モデル. 図は文献 [32] より引用.	8
2.4	cyber, physical, human を扱う概念の比較. (a)Human-in-the-Loop Cyber-Physical Systems (HiLCPS): human を起点とし cyber と physical を経由するフィードバックループを形成する, (b)Cyber-Physical Human Systems (CPHSs): CPS で扱う cyber と physical に加えて human も要素として扱うシステム, (c) 本論文で提案する Cyber-Physical-Human Interaction (CPHI): cyber と human のインタラクションにおける physical の影響も考慮対象とする	12
2.5	(a) ネットワーク層における OSI7 階層モデル. (b)CPHI における階層モデル.	13

2.6	CPHI の 4 パターン. C がサイバー/コンピュータ, P が物理, H が人間を示す. (a)Cyberization from Human (人間からのサイバー化): 人間の行動や思考をコンピュータが扱うことのできるデジタルデータへ変換, (b)Realization to Human (人間への意識化): コンピュータが処理した結果を人間へ伝達して人間はそれを意識として認識する, (c)Human-Physical Merging (人間と物理の情報融合): 人間の行動情報に物理空間の物理量を組み合わせてコンピュータへ伝達し意味付けをする, (d)Physical Augmentation (物理的拡張): コンピュータが物理空間を拡張することを通じて人間へと情報を伝達する.	14
2.7	CPHI の 4 パターンにおける CPHI モデル. C が cyber, H が human を示す. 間の青色の矩形及び矢印が, 情報伝達に用いられる物理量とその伝達方向である. (a) キーボードによる情報入力モデル. これは Cyberization from Human の事例となる. (b) ディスプレイによる情報揭示モデル. これは Realization to Human の事例となる. (c) タッチインタフェースを備えたデバイスによるコンテキストウェアアプリケーションのモデル. これは Human-Physical Merging の事例となる. (d) メガネ型デバイスによる Augmented Reality のモデル. これは Physical Augmentation の事例となる.	15
2.8	CPHI の 4 パターンの関係性.	16
2.9	スマートフォンにおける CPHI モデル.	17
3.1	SleepTyping の CPHI モデル.	19
3.2	センサとして使用したシリコンキーボード	20
3.3	システム構成図	21
3.4	システムの概観	21
3.5	描画システムの構成	21
3.6	描画システムのスナップショット	22
3.7	(a) 頭と腰, (b) 頭と足, (c) 肩と腰の 3 パターンのキーボード配置における, 体の位置とキー押下の関係.	23
3.8	睡眠時の寝返りの際のキー押下状況とその様子	25
3.9	AM2 時から 8 時までの 6 時間一晩を通して計測した結果. (a) 頭部のキーボードの計測結果. 押下イベントを 1 つの点として散布図として示し, 補助のためにイベント間を直線で結んでいる. 縦軸が押下したキーの位置の x 座標. 横軸が時刻. (b) 体部のキーボードの計測結果. 3 つのキーボードでの取得データを結合してひとつのグラフに示している. その他のグラフの設定は (a) と同じ. (c) スリープスキャンで計測し分析した睡眠ログ. グラフはスリープスキャン付属のアプリケーションより引用. 縦軸は睡眠の深さを覚醒, REM, 浅睡眠, 深睡眠の 4 段階で表している. 横軸は時刻.	26

4.1	PICALA の CPHI モデル. 上段の Cyberization from Human では, スマートフォン等のタッチインタフェースを想定し, 使用する物理量を静電容量とした.	28
4.2	PICALA の設置に用いたクリップ式電球ソケット. (a) 消灯時, (b) 点灯時 .	30
4.3	PICALA の聴講者用ボタンのユーザインタフェース. (a)PC ブラウザ (b) スマートフォン	30
4.4	PICALA のシステム構成図 (LAN モデル)	31
4.5	PICALA のシステム構成図 (インターネットモデル)	32
4.6	LAN モデルとインターネットモデルにおける, ボタン押下から照明点灯までの遅延時間の計測結果. エラーバーは標準誤差を示す.	33
4.7	実証実験で設置した PICALA. (a)WISS2014 でスクリーン下に設置した様子. (b)WISS2014 で演台に設置した様子. (c)wakate2015 でスクリーン下に設置した様子. (d)EC2015 でスクリーン下に設置した様子.	34
4.8	各研究会でボタンの押された割合の比較. 積み上げ式の棒グラフにて表示. .	35
4.9	ある講演におけるボタン押下の記録. 時間を 10 秒ごとに区切ってボタン押下回数をカウント. 積み上げ式の折れ線グラフにて表示.	35
4.10	アンケートの選択式設問の回答結果. エラーバーは標準誤差を示す. n=23(WISS2014), n=21(wakate2015), n=14(EC2015)	37
5.1	Augmented Typing の CPHI モデル.	41
5.2	5 種類の映像効果. 3 枚の図は効果の時間経過を示し, それぞれ上から下へ進む.	44
5.3	Web ページとして実装した演出システム.	46
5.4	システムのハードウェア構成	46
5.5	システム構成の全体像	47
5.6	効果映像 (wave) を投影した様子	48
5.7	効果映像の印象評価の結果. 直線の上下端は外れ値を除いた最大値、最小値、矩形の上下端は第三四分位点、第一四分位点. 矩形内の罫線は中央値, ×点は平均値. 矩形外の点は外れ値を示す.	49
5.8	効果音の印象評価の結果. 読み方は図 5.7 と同じである.	49
5.9	効果映像の印象の比較.	50
5.10	効果音の印象の比較.	50
5.11	particle における粒子数と拡散範囲の最適値のプロット. グループ A(拡散範囲が 100 前後のグループ) とグループ B(拡散範囲が 110 以上の右肩上がり)のグループ) に分けて線形近似を行った. グループ A の線形近似式は $y = 0.0349x + 94.332$ (相関係数 $R^2 = 0.19899$), グループ B の線形近似式は $y = 0.7593x + 113.96$ (相関係数 $R^2 = 0.94539$) である.	51
5.12	ray における速度と拡散範囲の最適値のプロット.	51

5.13	character における速度の最適値のプロット.	52
5.14	wave における速度と拡散範囲の最適値のプロット. 線形近似式は $y = 11.678xx + 161.39$ (相関係数 $R^2 = 0.31252$) である.	52
5.15	Maker Faire Singapore2016 での展示の様子	54
5.16	日本語入力支援システム	56
5.17	トポロジ表示によるタイピング学習支援システム	56
6.1	CPHI の物理における問題.	59
6.2	Bluetooth の階層構造.	62
6.3	接続確立までの HCI コマンドおよび HCI イベント, BLE フレームのシーケンス図.	63
6.4	BLE コントローラの状態遷移図.	63
6.5	BLE フレームのデータ形式.	64
6.6	既存のエミュレーション・シミュレーション手法の模倣ターゲットの比較.	66
6.7	BluMoon Frame (BM-FRM) のプロトコルフォーマット.	69
6.8	BluMoon における電波干渉の模倣手法の解説図. t_0 と t_1 が BM-FRM の到着時刻, t_{rcv} は BLE フレーム受信に要する時間である. (a)BM-FRM を受信した時刻から t_{rcv} が経過するまでに他の BM-FRM を受信しなければ, 電波干渉なしとみなしてその BM-FRM を受理する (b) 他の BM-FRM を受信した場合には電波干渉が発生したとみなしてその両方の BM-FRM を破棄する.	69
6.9	BluMoon の設計構成図.	72
6.10	BM-HCI の構成とそのインタフェース. Linux の Bluetooth スタック BlueZ に含まれる VHCI と, HCI によって接続する. この HCI 接続には, UNIX ドメインソケット <code>/dev/vhci</code> が用いられる. BM-MGR とは, コントローラ内インタフェースで独自フォーマットのデータをやり取りする.	72
6.11	Geneve にカプセル化し UDP セグメントとして格納した BM-FRM.	73
6.12	Location Information (LC-INF) の記述例. この例では, ノード ID1~4 の 2次元座標を指定している.	73
6.13	hciconfig コマンドの実行結果.	74
6.14	アダバイズパケット送信時の CPU 使用率. 横軸は 1 秒あたりのアダバイズ送信数. 縦軸は CPU 使用率. 値は BM-HCI, BM-MGR, BM-CNC のプロセスごとに算出される.	75
6.15	アダバイズパケット受信時の CPU 使用率. グラフの見方は図 6.14 と同様.	76
6.16	受信電波強度計測を実施した実環境. (a) 送信機と受信機の配置. (b) 屋外の計測環境. (c) 屋内の計測環境.	77
6.17	距離に応じた受信電波強度の計測結果. 横軸の距離は対数軸である. 各点は実測値, 直線は実測値の対数近似を示す.	77

6.18	受信率計測を実施した実環境.	78
6.19	アダバサイズパケット送信における T_{adv} , T_{ai} および T_d の関係.	78
6.20	干渉機の台数に応じた受信率の計測結果.	79
6.21	実験の設定.	80
6.22	実験を実施した実環境. (a) 全体像. (b) 受信機. (c) 送信機.	80
6.23	BLE ビーコンを用いた近接判定実験の結果. 近接判定と RSSI 実測データを上下に並べている. (a) 実環境. (b) BluMoon.	81
7.1	AOBAKO の全体像	85
7.2	AOBAKO のシステム構成	86
7.3	AOBAKO DESK.	88
7.4	AOBAKO SCOPE.	88
7.5	AOBAKO BOX.	88
7.6	AOBAKO BOX のシステム構成.	89
7.7	AOBAKO BOX の内部構造.	90
7.8	物理化されたビーコンの受信.	90

第 1 章

序論

1.1 Human-Computer Interaction

1.1.1 ユーザインタフェース

現在、コンピュータは日常生活に欠かすことの出来ない存在である。今日我々が使用するコンピュータは、無数の進化の上に成り立っている。コンピュータは、演算装置、制御装置、記憶装置、入力装置、出力装置によって構成され、これらはコンピュータの五大要素と呼ばれる。これは John von Neumann の報告書 *"First Draft of a Report on the EDVAC"*[1] に起源を持つ。コンピュータの発展の歴史とともに、人間とコンピュータのインタラクションのあり方は長年研究され、Human-Computer Interaction(HCI) と呼ばれる一分野として体系的にまとめられる。人とコンピュータのインタラクションの単純なモデルとして入出力という概念が用いられ、情報のやりとりは入出力装置を経由して行う。

入力装置と出力装置は、ユーザインタフェースとして進化してきた。現在我々が日常的に使っているグラフィカルユーザインタフェース (Graphical User Interface:GUI) は、Ivan Sutherland が提唱した Sketchpad[2](図 1.1) を起源に持つ。その後、Douglas Engelbart が開発した oN-Line System(NLS)[3][4](図 1.2) では、ディスプレイやマウスなど現在普及している入出力インタフェースを備える。Alan Kay が提唱した Dynabook[5](図 1.3) は、タッチパネルインタフェースを備えたパーソナルコンピュータのコンセプトを掲げた。

これらは後に商用製品にも影響を与える。GUI は Mac OS[6] や Windows[7], X Window System[8] などによって次々と実装され、コンピュータインタフェースのスタンダードとなった。また、コンピュータの入力装置としてはキーボードとマウスの組み合わせが最も一般的なものになった。iPhone[9](図 1.4) と iPad[10] の発売によって、2010 年代にはタッチパネルインタフェースも十分に普及している。その他にも、深度センサを用いた入力装置である Kinect[11][12](図 1.5) や、コンシューマゲーム用ヘッドマウントディスプレイ (Head Mounted Display:HMD) の PlayStation VR[13](図 1.6) など、様々な装置がコンピュータへの入出力に使われている。

2 第1章 序論



図 1.1. Sketchpad. 画像は文献 [2] より引用.



図 1.2. oN-Line Systems. 画像は文献 [4] より引用.

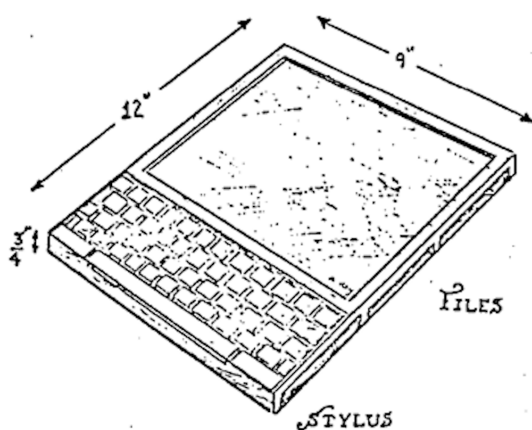


図 1.3. Dynabook のコンセプト図. 画像は文献 [14] より引用.



図 1.4. iPhone. 画像は Web サイト [9] より引用.



図 1.5. Kinect. 画像は文献 [12] より引用.



図 1.6. Playstation VR. 画像は Web サイト [13] より引用.

1.1.2 感覚

人間には、周囲の状況を知覚する能力が備わる。特に視覚、聴覚、触覚、味覚、嗅覚は五感とよばれる。五感は紀元前の Aristoteles による分類を起源とする [15] が、人間の感覚はこれらに限らない。例えば、皮膚感覚には触覚以外にも痛覚や温度覚もある。これらの感覚は、感覚器から神経を通じて脳に送られることで認識する。例えば、眼球の網膜に投影された像は、視神経を通じて脳の後頭葉の視覚野へと送られる。

感覚によって得た情報は脳で処理されるが、これは必ずしも実際の物理量とは一致しない。有名な例は錯視である。また、人間が刺激を知覚する量 (感覚量) は、実際の刺激量の対数に比例する [16]。つまり、 S を刺激量、 P を感覚量とした時、

$$P = k \log S \quad (1.1)$$

で表される。ここで、 k は定数である。これは Weber-Fechner の法則と呼ばれる、精神物理学の基本法則である。

人間は、感覚を単なる信号として扱うことにとどまらず、その存在や意味を獲得することができる。James Jerome Gibson は、人間とモノの間にある関係性をアフォーダンスという概念によって提唱した [17]。このアフォーダンスという単語は、後に、Donald Arthur Norman によって、形状などの物理的性質によってモノの操作方法や特徴を示すことという意味が付与された [18]。今日では、この意味の方がよく知られている。

1.2 Cyber Physical System

Cyber Physical System (CPS) は、物理空間とサイバー空間を統合的に取り扱う概念 [19] で、米国の報告書 *President's Council of Advisors on Science and Technology (PCAST)* [20] によってはじめて提唱された [21]。CPS の技術要素は多岐にわたる。その中でもとりわけ重要とされるものに、組込みシステム、リアルタイムシステム、分散システム、フィードバック制御、無線ネットワークなどが挙げられる。CPS は文脈によって様々な解釈され、Internet of Things (IoT) や Machine-to-Machine (M2M) という概念と対比されたり同一視されたりすることもあるが、CPS では機器のみではなく物理空間自体にも焦点を当てていることが大きな特徴である。

1.3 Cyber-Physical-Human Interaction

発展したコンピュータの世界において、物理空間における影響は無視できない。そこで、本論文では、人間とコンピュータと物理のインタラクションを統合的に取り扱うための新しい概念として Cyber-Physical-Human Interaction (CPHI) を提唱する。本論文で CPHI の概念をまとめ、人間とコンピュータのインタラクションにおける物理空間の影響を考慮することにより、目的のインタラクションに対するセンサやアクチュエータの物理要件を定めたり、衝突、

4 第1章 序論

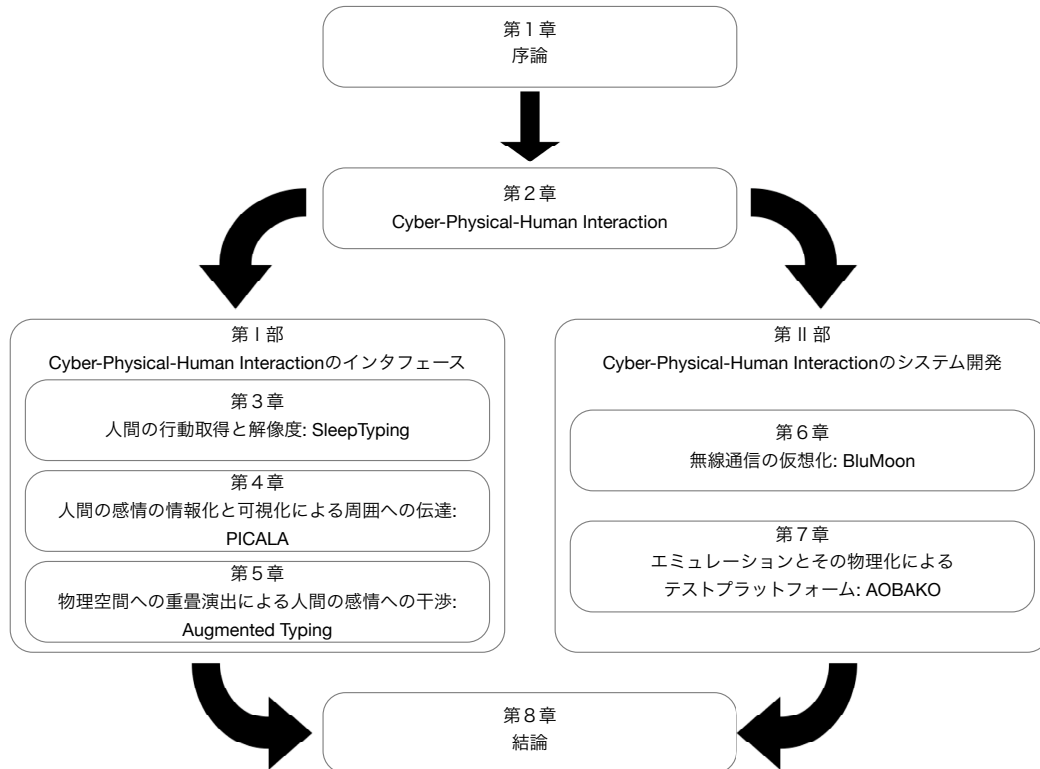


図 1.7. 本論文の構成.

干渉、遮蔽といった物理的な問題をインタラクションの設計段階で発見したりすることができるようになる。

1.4 本論文の構成

本論文は、図 1.7 に示すとおり構成で記述する。第 2 章では、本論文で提唱する CPHI の詳細と関連研究を記述する。その後、研究事例を 2 部構成で記述する。第 I 部は CPHI のインタフェースに関する研究について記述する。第 3 章では、市販のシリコンキーボードを用いた睡眠時姿勢センサ SleepTyping の事例を記述する。第 4 章では、プレゼンテーションにおいて聴講者のリアクションをリアルタイムに可視化するシステム PICALA の事例を記述する。第 5 章では、キーボードのタイピング体験を光と音の効果によって拡張する Augmented Typing の事例を記述する。第 II 部は Cyber-Physical-Human Interaction のシステム開発に関する研究について記述する。第 6 章では、ソフトウェアテストのための Bluetooth Low Energy (BLE) エミュレータ BluMoon について記述する。第 7 章では、BLE を用いたアプリケーションのためのテストプラットフォーム AOBako について記述する。最後に、第 8 章では、これらの研究と本研究で提案する CPHI との関係性をまとめ、本研究の結論を記述する。

第 2 章

Cyber-Physical-Human Interaction

2.1 関連する概念

2.1.1 Cybernetics

機械や生物などの対象分野にかかわらず制御工学と通信工学の問題が統一的に扱おうという着想のもと、制御と通信の理論の全領域を示す学術分野がサイバネティックス (cybernetics) と名付けられた。これは舵手を意味するギリシャ語から取られた。デジタルやコンピュータを指す接頭語 cyber の語源にもなる。サイバネティックスに関わるあらゆる分野の研究者、例えば計算機科学者や物理学者、応用数学者、神経生理学など様々な分野の研究者を集めて議論され、この会議はサイバネティックス会議と呼ばれた。サイバネティックスに関する議論と知見は Norbert Wiener によって "*Cybernetics: or Control and Communication in the Animal and the Machine*" [22] としてまとめられた。

2.1.2 Ubiquitous Computing

Mark Weiser は、1991 年に "*The Computer for the 21st Century*" [23] を発表した。この文章では、生活のあらゆる場面にコンピュータが埋め込まれる 21 世紀のコンピューティングの姿が描かれ、タッチパネル式の情報端末、ウェアラブル型デバイスによる機器の自動操作、機器同士の無線通信などの技術が予想された。このコンセプトは、「遍在する」という意味の単語 "ubiquitous" を用いてユビキタスコンピューティング (ubiquitous computing) と名付けられた。このコンセプトはコンピュータ技術の発展に大きな影響を与え、2019 年の現在、コンセプトが社会に着々と実装されつつある。

2.1.3 Context Aware

コンテキストアウェア (context aware) は、状況を考慮した、という意味の接頭語で、コンテキストアウェアアプリケーション (context aware application) やコンテキストアウェアシステム (context aware system) という形で用いられる。コンテキストアウェアの概念は Schilit

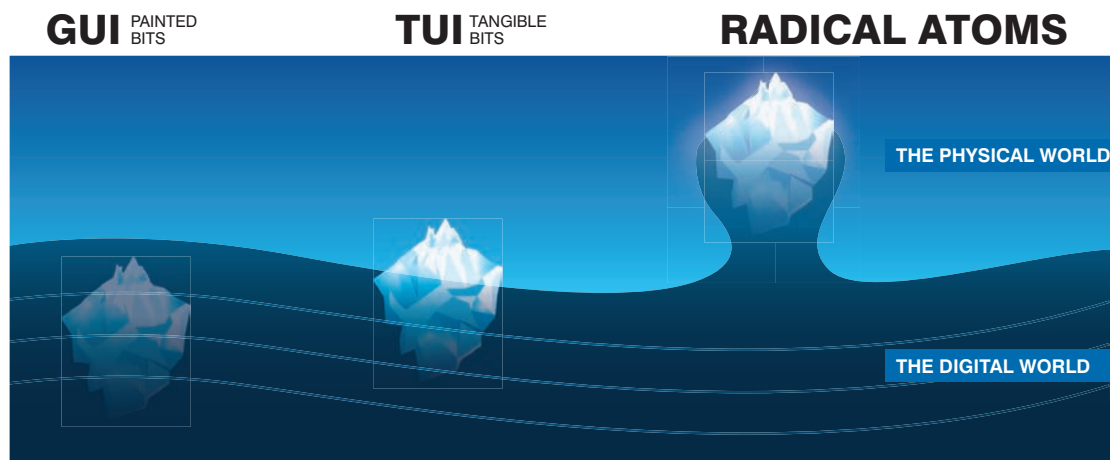


図 2.1. ラディカル・アトムズのコセプト。図は文献 [27] より引用。

and Theimer[24] で初めて登場し、その後、Dey[25] によって詳細にまとめられた。

Knappmeyer *et al.* のサーベイ論文 [26] では、コンテキストとは、エンティティとデジタル世界との間の相互作用に関する特徴を提供するあらゆる情報であると定義される。ここでのエンティティとは、ユーザ、アプリケーション・サービス、デバイス、または空間的に結合されたスマートな場所などを指す。さらに、コンテキストの要素として

- Spatial context : 場所, 建物など
- Temporal context : 時間, 曜日など
- Device context : 画面サイズなど
- Network and communication context : ネットワーク遅延など
- Environmental context : 物理環境. 騒音, 光の強さなど.
- Individuality and user profile context : ユーザの嗜好, 関心, 習慣などに関するコンテキスト
- Activity context : 現在どのタスクに関わっているのか, 次に何をするのか
- Mental context : 心の内部状態に関する情報. 感情, 気分, ストレス
- Interaction context : ユーザ間のインタラクション, ユーザとアプリケーションのインタラクションなど

の 9 つの例を挙げている。ただし、これがコンテキストの全てではないとも述べている。

2.1.4 Radical Atoms

物理空間の制御も含めたコンピュータと人間のインタラクションの事例として、石井裕が発表したラディカル・アトムズ (radical atoms)[27] がある。ラディカル・アトムズは、コンピューショナルに変形と再構成できる動的物質と人間とのインタラクションのためのビジョンである。このコンセプトのもと、inForm[28] や Transform[29] といった動的再構成可能な物質

によるインタフェース研究が進められた。石井は過去に情報を触れるユーザインタフェースに落とし込むタンジブル・ビット (tangible bits)[30] というコンセプトを提案している。ラディカル・アトムズは、その発展形である (図 2.1)。

2.1.5 Human-in-the-Loop Cyber-Physical Systems

Schirner *et al.*[31] は、Human-in-the-Loop Cyber-Physical Systems (HiLCPS)(図 2.2) を提唱した。HiLCPS では、ボディセンサや脳センサのような、今後発展するようなインタフェースを用いることを想定している。人間が脳波や筋電でサイバー空間の組込みシステムに情報を送り、組込みシステムはロボットアームなどのアクチュエータやディスプレイなどにより物理空間中の物体の状態を変化させる。その物理空間の変化は、人間が読み取ることができる。このようなフィードバックループを形成する。

HiLCPS は、本論文で提唱する CPHI と同様に cyber, physical, human を取り扱う概念である。しかし、HiLCPS では、人間の脳波や筋電を利用して human から cyber への情報伝達することを想定しており、その経路における physical については着目していない。CPHI では、human から cyber と、cyber から human の双方向において physical の振る舞いを考慮する。この点で、HiLCPS と CPHI は異なる概念である。これらの違いは図 2.4(a),(c) に示す。

2.1.6 Cyber-Physical Human Systems

Sowe *et al.*[32] は、Cyber-Physical System の中で人間もシステムの一部として取り扱う Cyber-Physical Human Systems (CPHSs) を提唱した。この研究では、人間をシステムとして取り扱うために、人間を機能としてモデル化した (図 2.3)。このモデルには Human Service Capability Description (HSDC) モデルという名前が付けられている。

CPHSs は、CPHI と同様に cyber, physical, human を取り扱う。しかし、CPHSs はこれらをそれぞれシステムの要素として扱っている。一方で、CPHI は physical を経由したインタラクションに着目している。この点で、CPHSs は CPHI とは異なる概念である。これらの違いは図 2.4(b),(c) に示す。

2.2 Cyber-Physical-Human Interaction の提案

人間は脳で、コンピュータは処理装置や記憶装置で、それぞれ情報を扱う。興味深いことに、人間とコンピュータのいずれも、電気信号を用いて情報伝達を行う。脳の情報を読み取る方法の一つが、神経細胞の活動によって生じる電位変動を計測するというものである。これは脳波あるいは脳電図 (electroencephalogram:EEG) と呼ばれる。EEG ような人間の脳の情報を、コンピュータとのインタフェースに用いたものはブレインマシンインタフェース (Brain-Machine Interface:BMI) と呼ばれる。究極の BMI とは、脳から直接コンピュータへ情報を入力し、また、コンピュータからの出力を直接脳に注入するものであろう。しかしながら、現状では、BMI で取り扱うことのできる情報は極めて限定的である。そのため、人間と

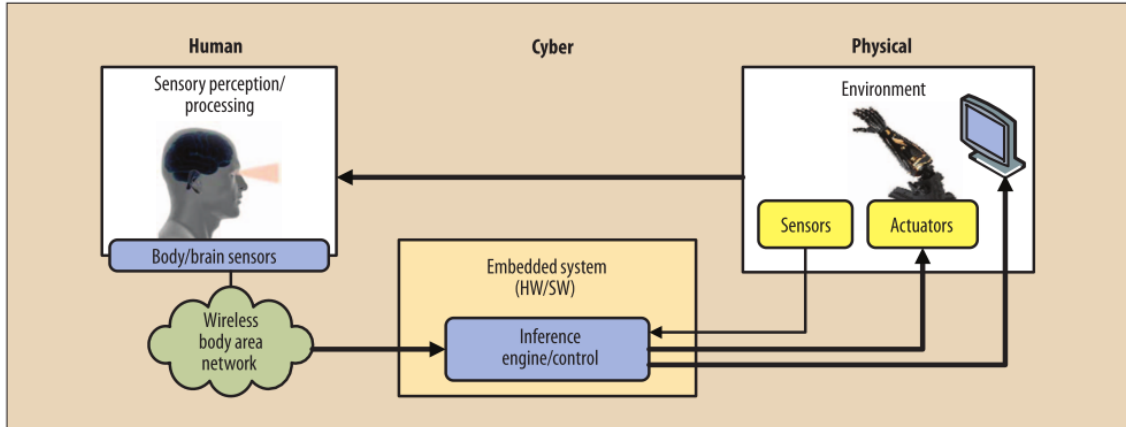


図 2.2. Human-in-the-Loop Cyber-Physical Systems の概念図. 図は文献 [31] より引用.

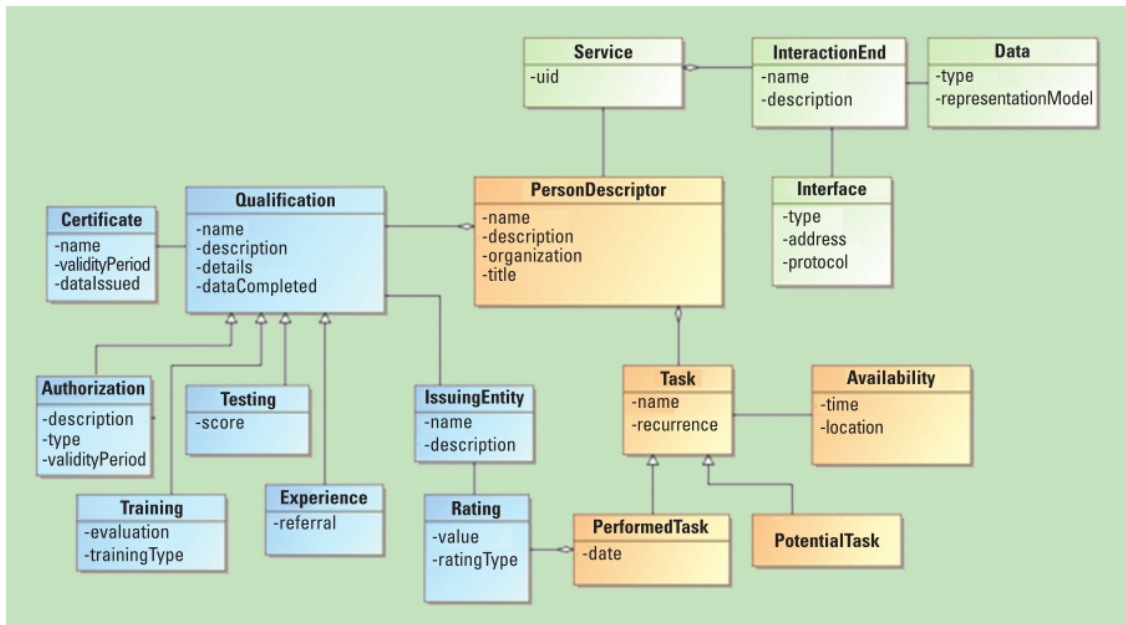


図 2.3. Cyber-Physical Human Systems の Human Service Capability Description モデル. 図は文献 [32] より引用.

コンピュータのインターフェースは、必ず何かしらの物理現象を通じたものになっている。

また、ネットワークでは、コンピュータ同士の通信のモデルとして OSI7 階層モデルが広く用いられている (図 2.5(a)). OSI7 階層モデルでは、最下層のレイヤー 1 は物理層と呼ばれる。有線通信では電気信号、無線通信では電磁波の変化を介して遠隔のコンピュータ同士が通信を行う。このように、情報伝達において必ず物理現象を通じたものになることは、コンピュータ同士の通信からも伺うことができる。

本論文では、このような、人間とコンピュータが物理を介して相互作用を行う概念を Cyber-Physical-Human Interaction (CPHI) として提唱する。CPHI では、cyber と human のインタラクションにおいて、インタラクション層と物理層の 2 層モデルを考える (図 2.5(b)). 従

来考えられてきた HCI はインタラクション層に着目しているが、CPHI では実際に情報のやり取りを行う物理層も含めて cyber と human のインタラクションを考える。

本論文では、CPHI のパターンとして

- Cyberization from Human (人間からのサイバー化)
- Realization to Human (人間への意識化)
- Human-Physical Merging (人間と物理の情報融合)
- Physical Augmentation (物理的拡張)

の 4 つを提案する。

2.2.1 Cyberization from Human

Cyberization from Human (人間からのサイバー化) は、人間の行動や思考をコンピュータが扱うことのできるデジタルデータへ変換することを指す。これは、従来の HCI における、コンピュータへの入力に相当する。ただし CPHI では情報伝達の際に必ず物理現象を経由することに着目する。代表的な入力装置であるキーボードは、人間が指でキーを押下した圧力を検知し、その情報をコンピュータへと伝える。

2.2.2 Realization to Human

Realization to Human (人間への意識化) は、コンピュータが処理した結果を人間へ伝達し、人間の意識として認識するまでのフローである。これは、従来の HCI における、コンピュータからの出力に相当する。ここでも情報伝達の際には、その間に何らかの物理現象をとまなう。代表的な出力装置である液晶ディスプレイは、液晶素子に電圧を印加することで液晶分子の向きを変えることで画素の色を任意のものに制御し、人間はその光を視覚によって取り込む。

2.2.3 Human-Physical Merging

Human-Physical Merging (人間と物理の情報融合) は、人間の行動情報に、物理空間の物理量を組み合わせてコンピュータへ伝達することで、その行動情報に意味付けをする。先に記述した Cyberization from Human と似ているが、Human-Physical Merging では物理空間を利用した単なる情報伝達ではなく、それに加えて他の物理量を活用するところに特徴がある。既存の概念では、前節で記述した Context Aware が該当する。

2.2.4 Physical Augmentation

Physical Augmentation (物理的拡張) は、コンピュータが物理空間を拡張することを通じて人間へと情報を伝達する。先に記述した Realization to Human と似ているが、Physical Augmentation は単なる情報伝達ではない。Realization to Human では情報伝達のために

物理空間を用いたが、Physical Augmentation では、もともと意味や情報を持っていた物理空間に重畳することによって別の情報を付与する。既存の概念では、Augmented Reality や Mixed Reality に相当する。

2.2.5 Cyber-Physical-Human Interaction のモデル化

cyber と human の情報伝達において physical が果たす役割を整理するために、実際の情報伝達をモデル化したものを図 2.7 に示す。ここでは、キーボードによる情報入力、ディスプレイによる情報提示、タッチインタフェースを備えたデバイスによるコンテキストウェアアプリケーション、メガネ型デバイスによる Augmented Reality の 4 種類の事象についてのモデルを示す。(a) のキーボードによる情報入力では、human の操作によって発生した物理量変化を介して cyber へ情報を伝達する。(b) のディスプレイによる情報提示も、cyber の処理によって発生した物理量変化を介して human へ情報を伝達する。(a) と (b) の 2 パターンでは、human および cyber が情報伝達を意図して変化を起こした物理量のみを使用している。(c) のコンテキストウェアアプリケーションでは、human の操作による物理量の変化のほか、時間や場所といった物理量も利用する。(d) の Augmented Reality では、cyber の処理による物理量の変化の他に、もともと物理的に存在した物体も利用する。(c) と (d) の 2 パターンでは、このように、情報伝達のために変化させた物理量の他に存在する物理量も活用する。この 4 パターンは、伝達するソースの種類数と、情報伝達の向き (human から cyber または cyber から human) によって図 2.8 のように分類できる。

実際の CPHI を考えると、これらの 1 つのみに合致する場合は少なく、複数が組み合わさった複合的なものとなる。例えば、スマートフォンは多種のセンサとアクチュエータを搭載し、様々な物理量を用いてインタラクションを行う (図 2.9)。

2.3 physical を考慮したシステム開発

前節の 4 つのパターンでも示したとおり、CPHI では、すべての情報伝達において物理空間を活用する。そのため、物理空間で発生しうる問題は、そのまま CPHI の問題にもなりうる。具体的には、干渉や減衰などがある。

物理空間は排他制御を行ったりはしないため、制御の干渉が起こりうる。物理空間で扱う物理量の中でも、特に光、音、電波などは発生源から無指向で広がるため、干渉が問題となりやすい。電波干渉は、実際に起きている大きな問題の一つである。光の干渉を調停によって解決するアプローチも試みられている [33]。同様に光、音、電波は発生源からの距離に応じて強度が変化する。一般的には発生源から遠ざかるほど強度が小さくなる。この強度によって、情報伝達の成否に影響を及ぼす。屋内測位システムのような、この強度を利用するシステムもある。Human-Physical Merging や Physical Augmentation では、1 つのインタラクションにおいて複数の物理量を利用する。

このように、CPHI の概念を適用してシステム開発を再考すると、物理量の変化を考慮する

ことが重要である。しかし、物理現象を含めてシステム開発を行う場合、物理現象に置いて再現性がないことが大きな課題となる。不具合が生じた際に、ソフトウェアシステム検証であれば同じ条件で原因究明や再検証を行うことができるが、物理空間では条件を再現して検証することが困難である。この課題を解決する方法が、物理現象を模倣するシミュレーションやエミュレーションといった仮想環境を活用することである。本論文では、CPHI のためのシミュレーションやエミュレーションを活用したシステム開発について提案する。

2.4 本論文における CPHI の位置づけ

本論文では、SleepTyping, PICALA, Augmented Typing, BluMoon, AOBAKO という 5 つの研究について記述する。第一部では、CPHI のインタフェースに関する研究を列挙し、CPHI の 4 つのパターンのうち 3 つと比較する。

SleepTyping は、市販のシリコンキーボードをベッドに敷いて睡眠時姿勢センサとして活用する。これは、人間の行動情報を収集し分析するもので、Cyberization from Human に相当する。キーボードはもともと文字入力のためにキー押下の圧力を検知するものだが、人間の姿勢という異なる情報伝達のために同じ物理現象を用いた。

PICALA では、プレゼンテーションにおいて聴講者のリアクションをリアルタイムに可視化する。情報伝達には、会場に設置した電球の光の色を用いる。各聴講者のリアクションを収集するフローは Cyberization from Human にあたる。そして、その収集したリアクションを光の色によって可視化し、それを見た発表者や他の聴講者が光の色を見てリアクションを認識することは Realization to Human にあたる。

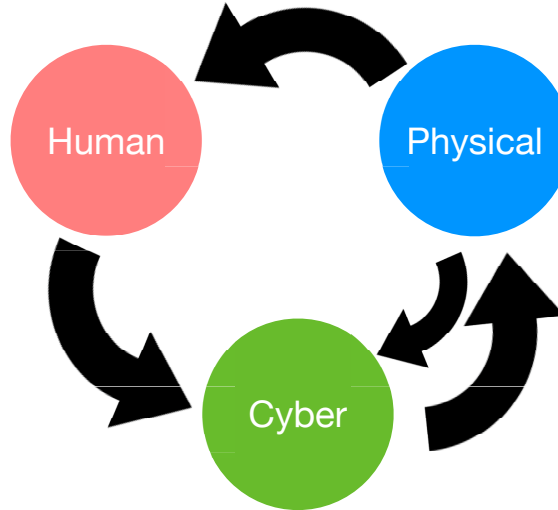
Augmented Typing は、キーボードのタイピング体験を光と音の効果によって拡張する。もともとそれ独自で成立していたタイピングという行為を、光と音という物理現象によって印象や感じ方を付与するものである。これは CPHI の Physical Augmentation にあたる。

第二部では、CPHI のシステム開発を行う場合の、エミュレータ等を用いた開発支援の提案を行う。このエミュレータでは、CPHI において重要である物理における問題を含有する。

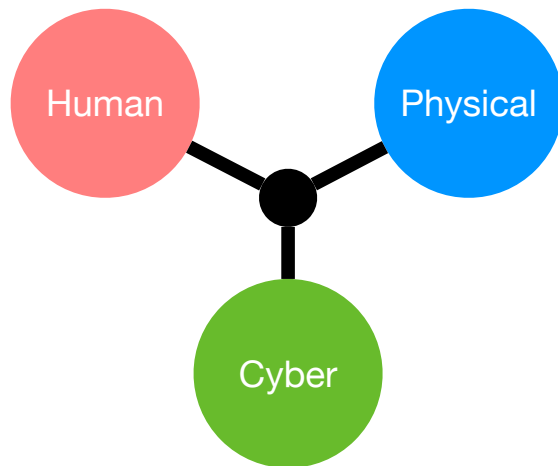
BluMoon は、Bluetooth Low Energy (BLE) の通信を有線ネットワーク上で模倣する、ソフトウェアテストのための無線エミュレータである。エミュレータとしての BluMoon の特徴が、電波干渉と受信電波強度の減衰を模倣することである。これらの物理における問題をエミュレータ内で模倣し、物理現象を計算的に発生させる。

AOBAKO は、BLE を用いたアプリケーションのためのテストプラットフォームである。BLE ビーコンと受信端末を任意の位置に設定し、その位置における電波状況を上述のエミュレータ BluMoon で計算し、さらにそれを物理空間で再現することで実機でテストを行うことができるものである。事例の一つとして BLE を用いたが、CPHI 志向アプリケーションを開発する際の新たなテストプラットフォームのコンセプトモデルとなる。

(a) Human-in-the-Loop Cyber-Physical System (HiLCPS)



(b) Cyber-Physical Human Systems (CPHSs)

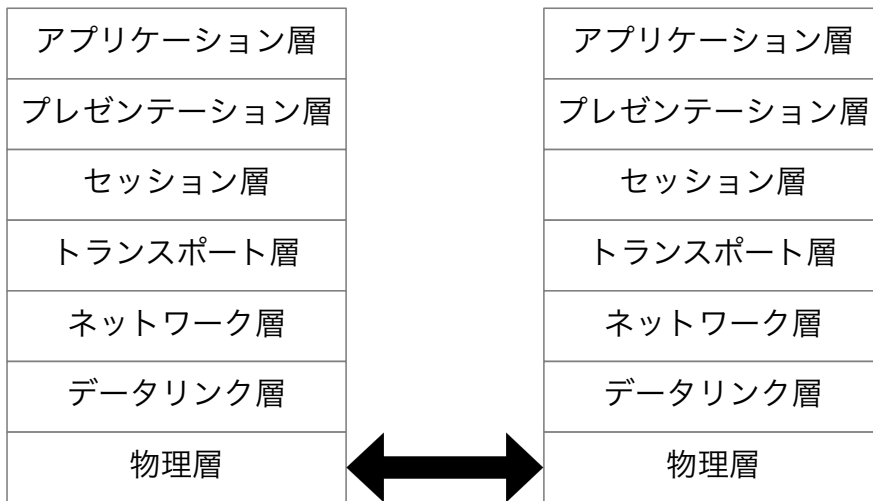


(c) Cyber-Physical-Human Interaction (CPHI)



図 2.4. cyber, physical, human を扱う概念の比較. (a)Human-in-the-Loop Cyber-Physical Systems (HiLCPS): human を起点とし cyber と physical を経由するフィードバックループを形成する, (b)Cyber-Physical Human Systems (CPHSs): CPS で扱う cyber と physical に加えて human も要素として扱うシステム, (c) 本論文で提案する Cyber-Physical-Human Interaction (CPHI): cyber と human のインタラクションにおける physical の影響も考慮対象とする

(a) ネットワークにおけるOSI 7階層モデル



(b) CPHIにおける階層モデル

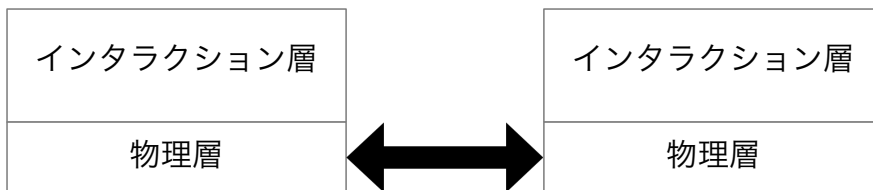


図 2.5. (a) ネットワーク層における OSI7 階層モデル. (b)CPHI における階層モデル.

(a) Cyberization from Human



(b) Rearization to Human



(c) Human-Physical Merging



(d) Physical Augmentation

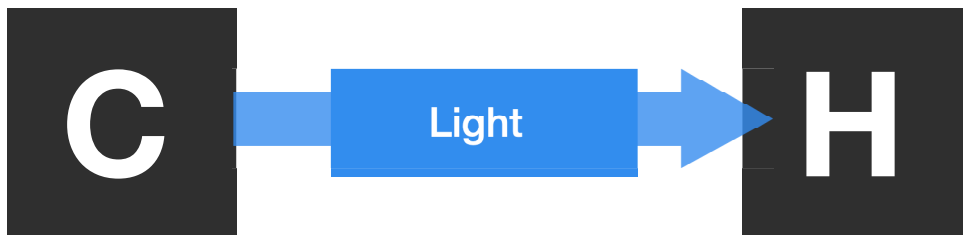


図 2.6. CPHI の 4 パターン. C がサイバー/コンピュータ, P が物理, H が人間を示す.
 (a)Cyberization from Human (人間からのサイバー化): 人間の行動や思考をコンピュータが扱うことのできるデジタルデータへ変換, (b)Realization to Human (人間への意識化): コンピュータが処理した結果を人間へ伝達して人間はそれを意識として認識する, (c)Human-Physical Merging (人間と物理の情報融合): 人間の行動情報に物理空間の物理量を組み合わせてコンピュータへ伝達し意味付けをする, (d)Physical Augmentation (物理的拡張): コンピュータが物理空間を拡張することを通じて人間へと情報を伝達する.

(a) Keyboard: Cyberization from Human



(b) Display: Rearization to Human



(c) Context-Aware: Human-Physical Merging



(d) Augmented Reality: Physical Augmentation

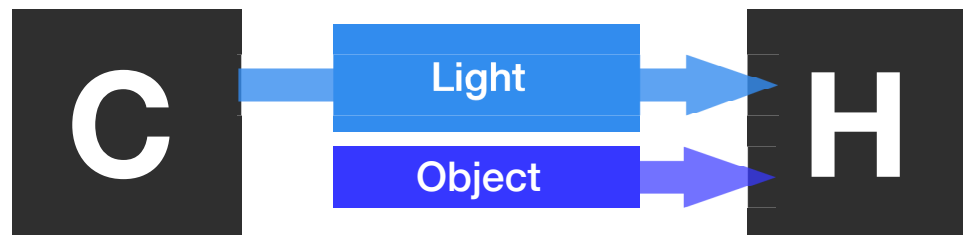


図 2.7. CPHI の 4 パターンにおける CPHI モデル。C が cyber, H が human を示す。間の青色の矩形及び矢印が、情報伝達に用いられる物理量とその伝達方向である。(a) キーボードによる情報入力モデル。これは Cyberization from Human の事例となる。(b) ディスプレイによる情報揭示モデル。これは Realization to Human の事例となる。(c) タッチインタフェースを備えたデバイスによるコンテキストウェアアプリケーションモデル。これは Human-Physical Merging の事例となる。(d) メガネ型デバイスによる Augmented Reality のモデル。これは Physical Augmentation の事例となる。

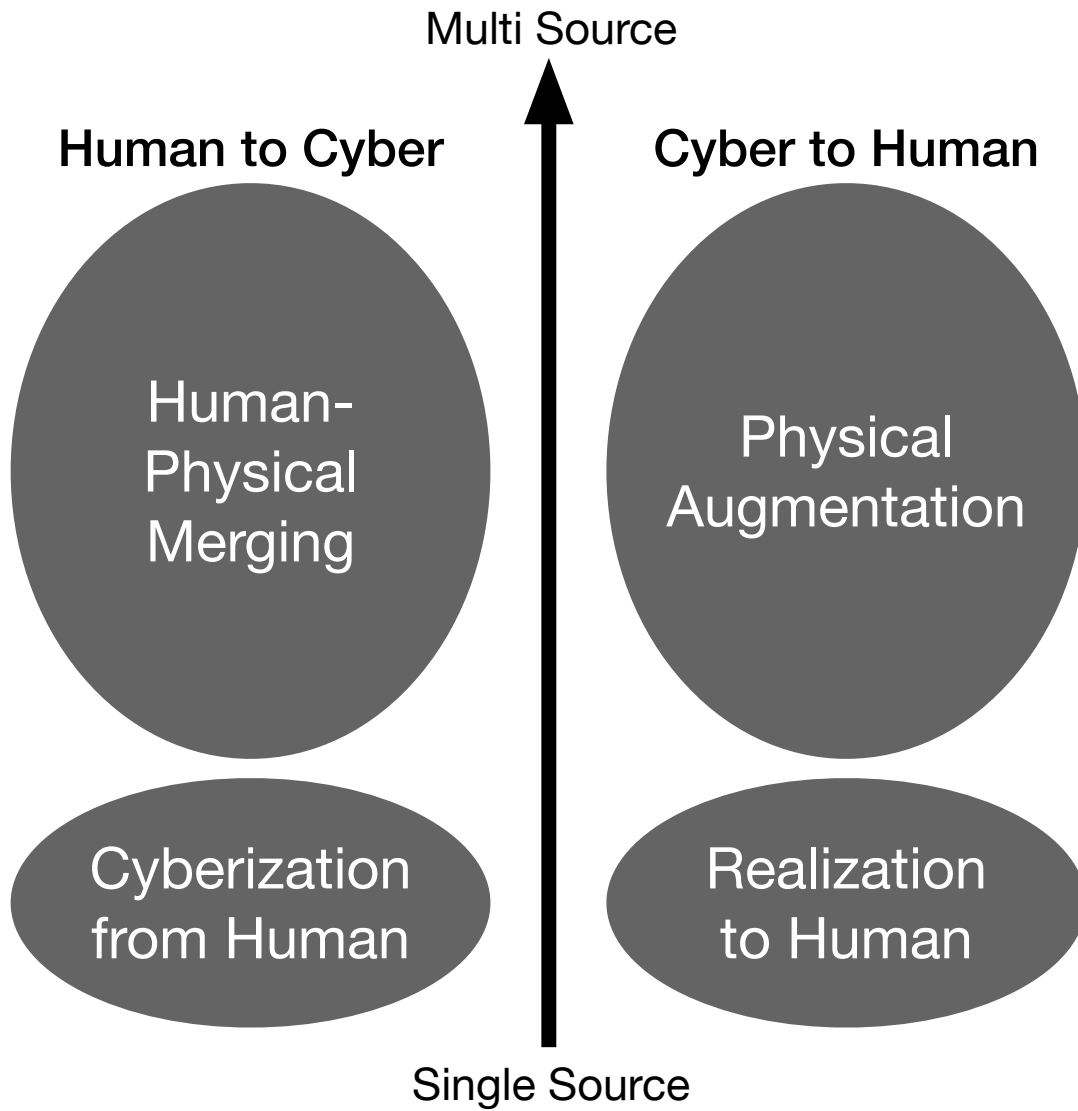


図 2.8. CPHI の 4 パターンの関係性.

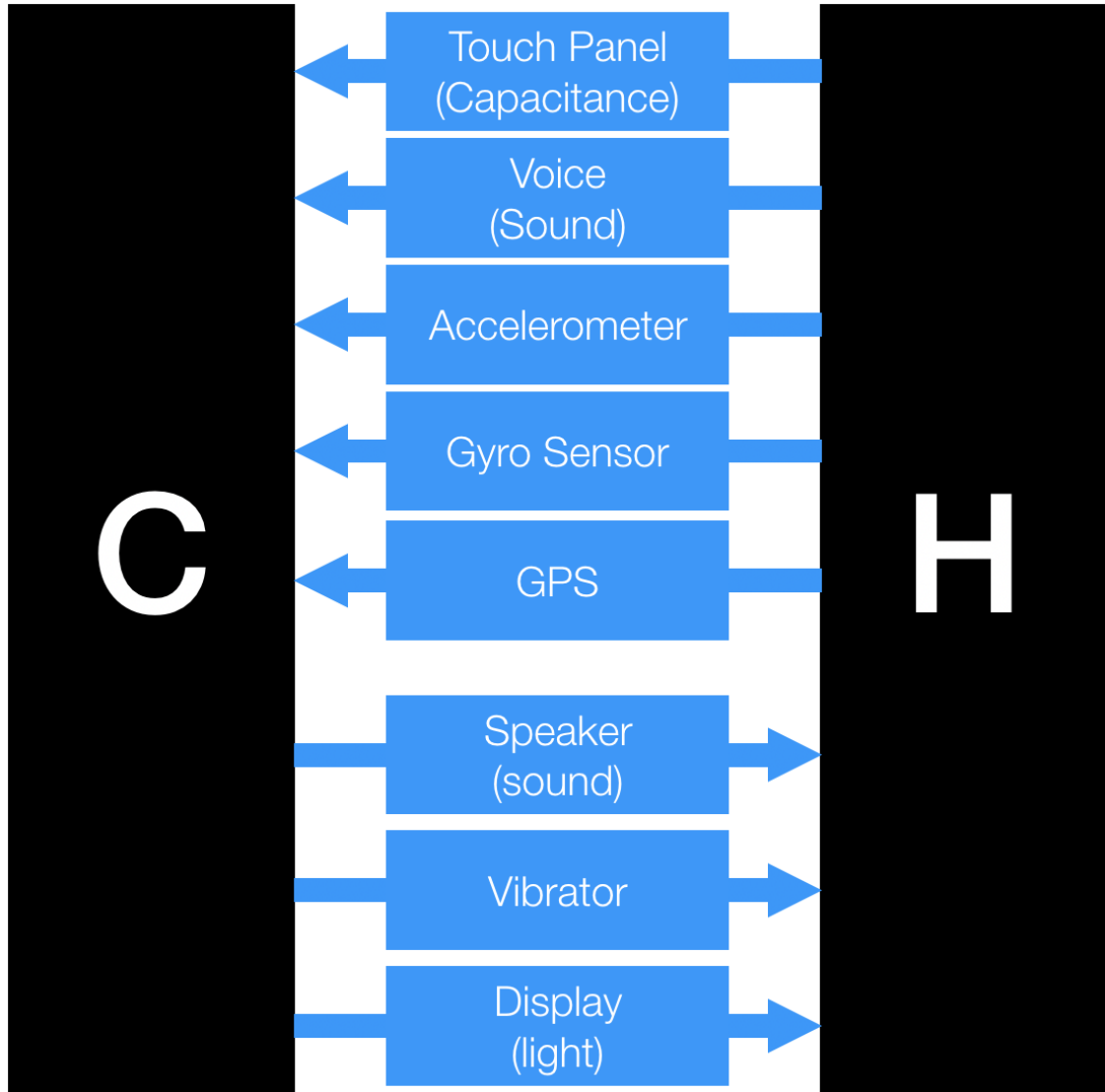


図 2.9. スマートフォンにおける CPHI モデル.

第I部

Cyber-Physical-Human Interaction の インタフェース

第 3 章

人間の行動取得と解像度: SleepTyping

3.1 はじめに

3.1.1 CPHI としての位置づけ

本節では、睡眠時の人間の体勢情報の取得に関する研究を通じ、CPHI の事例として考察する。体勢取得のためのセンサとして、圧力という物理量を用いる。これは、CPHI の 4 パターンのうち Cyberization from Human にあたる (図 3.1)。感圧センサには、汎用のシリコンキーボードを使用する。「キー入力」を目的としたデバイスを用いて、「体勢取得」という異なる情報を取得する。取得する物理量と精度が合致すれば、目的が異なるセンサでも流用可能であることを示す

3.1.2 背景

健康管理のために、さまざまなセンサ類の普及が進んでいる。その中でも、睡眠時の情報はとりわけ重要なデータである。睡眠時は浅い眠りと深い眠りを交互に繰り返すことが知られており、センサの情報を使えば寝返りの頻度などから推定することができる。

これまで、圧力センサを用いたもの [34]、脈波を用いたもの [35]、赤外線モーションセンサを用いたもの [36]、3 軸加速度センサを用いたもの [37]、WiFi の電場強度を用いたもの [38] など様々な睡眠センサが研究されてきた。

また、市販品としてもマットレスの下に敷いて体圧を計測するもの [39] から、電波を使って計測するもの [40]、加速度センサを用いて計測するもの [41]、腕につけるウェアラブルセンサ



図 3.1. SleepTyping の CPHI モデル。

で計測するもの [42] などのセンサが登場してきた。

これまで研究されてきた睡眠センサの技術は、高い精度を有するものもあるが、専門的な用途での使用に限定され、手軽に使うことはできない。

市販品のセンサは、一般向けで低価格ということもあり入手しやすいが、機能面や性能面に制限がある。日常生活において精度の良いデータを収集するためには、安価で高性能なセンサが求められる。そこで本研究では、安価な材料を使用して、配置や数を自由に変更できる睡眠時姿勢センサを開発した。

3.2 提案手法

3.2.1 設計と実装

感圧センサは、配置や数の柔軟な変更を可能にするため、安価で手に入りやすいものを使用したい。そこで、市販のシリコンキーボードを感圧センサとして使用した。キーボードは、多くのキーが並んでいてその押下を検出することができるので、配列型の感圧センサとして用いることができる。シリコンキーボードは柔らかいので、ベッドに敷いても違和感なく就寝することができる。

本研究ではサンワサプライ社製のシリコンキーボード (図 3.2) を使用した。

シリコンキーボードのキーの押下を監視し信号を取得する PC も、同様に安くて手に入りやすいものがある。そこで Raspberry Pi を使った。

システムの全体構成図を図 3.3 に示す。また、システムの概観を図 3.4 に示す。

キーボードの USB 端子は Raspberry Pi の USB ポートに接続する。Raspberry Pi では、キーの押下を監視する Ruby スクリプトが稼働している。この Ruby スクリプトの実装には libusb^{*1} を用いている。キーの押下を検知すると、キーの種類とキーボード番号から、押下されたキーの物理位置を計算する。あらかじめキーボードを実測してキーの種類と物理位置とのマッピングを計測済みで、物理位置の計算はこのマッピングが書かれたテーブルを参照することにより行う。計算された物理位置は、サーバへのアップロードされる。サーバへのアップロードは HTTP GET を用い、x 座標と y 座標をそれぞれパラメータに含める。



図 3.2. センサとして使用したシリコンキーボード

*1 <http://libusb.info/>

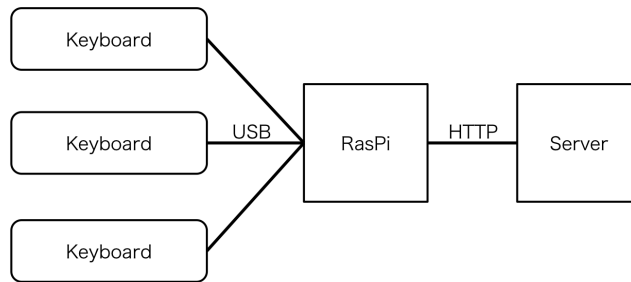


図 3.3. システム構成図

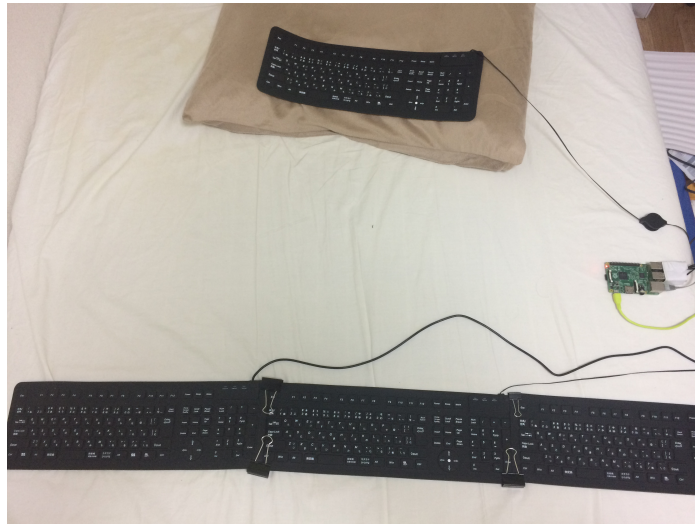


図 3.4. システムの概観

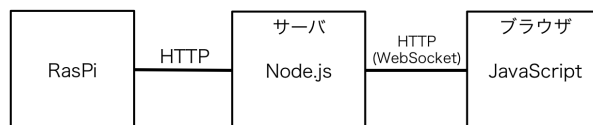


図 3.5. 描画システムの構成

サーバでは、node.js を用いて作成した HTTP サーバが稼働する。キー押下データのログファイルへの保存と、可視化のための描画を行っている。ログファイルは、キーの押下時刻、x 座標、y 座標、キーボード番号が csv 形式で保存される。

描画システムの構成図を図 3.5 に、図とスナップショットを図 3.6 に示す。

描画システムも node.js を用いており、ウェブブラウザで閲覧することができる。描画部分

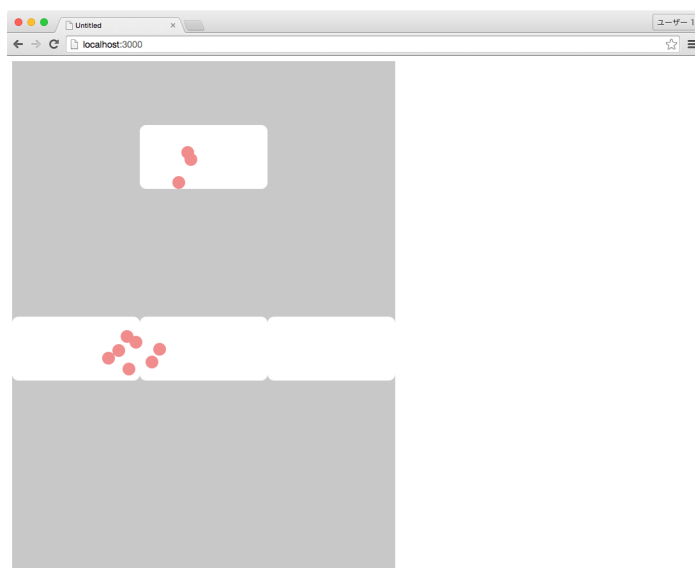


図 3.6. 描画システムのスナップショット

の記述には p5.js^{*2}を使用した。p5.js は JavaScript ライブラリで、Processing と非常に似た構文で記述できるためグラフィカルな用途に向いている。HTTP GET により送信されたキー押下の x 座標と y 座標に相当する位置に点を描画する。点は、描画が開始された瞬間が最も大きく、時間が経つにつれて小さくなっていく。点が同時最大描画数を超えると、古い点から順に消滅する。本論文の実験では同時最大描画数を 10 に設定した。HTTP サーバが受け取ったキー押下情報は、WebSocket 経由で描画部分へ受け渡している。そのため閲覧中のウェブページは即時更新される。WebSocket での通信を行うために、実装には socket.io^{*3}を用いた。

本論文の実験では、サーバと Raspberry Pi は同一 LAN 内に設置した。サーバをインターネット上に設置しどこからでもアクセス可能とする構成や、すべての機能を Raspberry Pi 内にて処理しオフラインで完結する構成など、使用環境の要求に対して柔軟に適応することができる。

3.2.2 システムテスト

本システムで任意の姿勢をとった時のキー押下状況を把握するため、寝ている様子の写真と比較する。シリコンキーボードの配置を変えて

1. 頭と腰
2. 頭と足
3. 肩と腰

*2 <http://p5js.org/>

*3 <http://socket.io/>

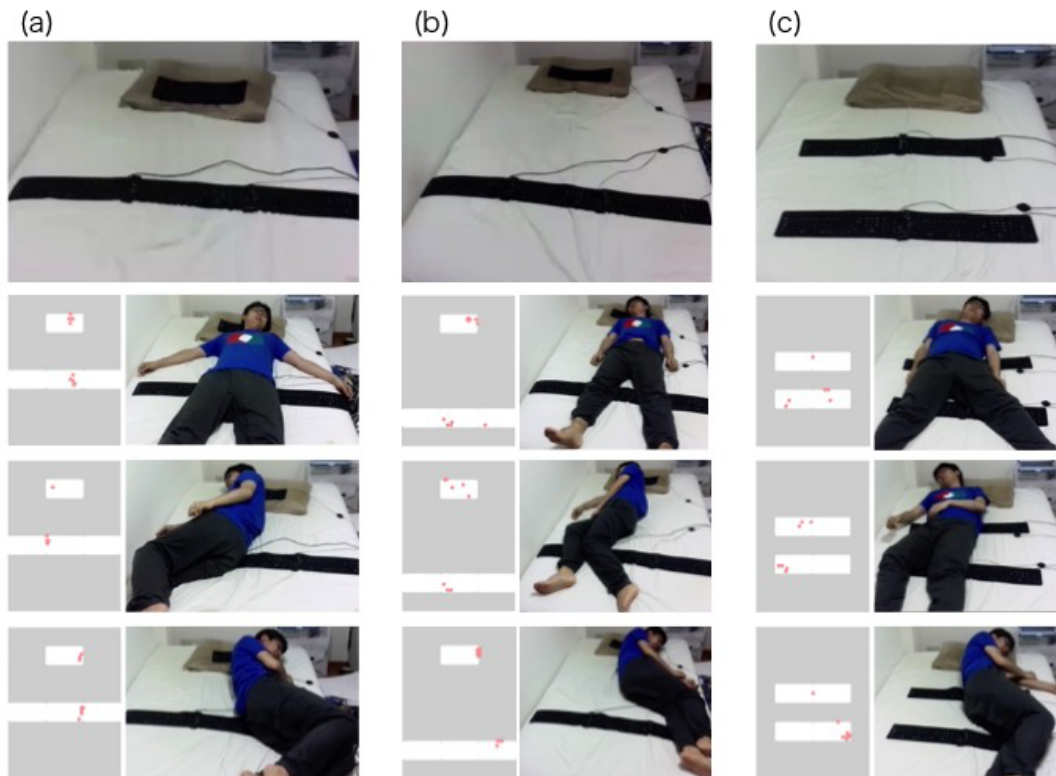


図 3.7. (a) 頭と腰, (b) 頭と足, (c) 肩と腰の 3 パターンのキーボード配置における, 体の位置とキー押下の関係.

の 3 パターンで比較を行った. それぞれのパターンにおいて, 体がベッドの中央, 左, 右に位置しているときの描画システムのスナップショットを図 3.7 に示す.

3.3 評価

実際の睡眠時に使用可能か検証するため, 睡眠時に使用する評価実験を行った. 評価は,

- ビデオにより撮影した映像との比較
- 他の睡眠センサとのデータの比較

の二つの方法で行った. これら二つの実験は別日に行った. 被験者は筆者 (湯村) で, 30 代成人男性である. 実験は自宅で実施した.

3.3.1 ビデオ撮影による評価

一晩を通じて睡眠時の様子を動画撮影し, 押下されるキーとの比較を行った. 比較は目視での確認により行った. 図 3.8 は, 4:58:13 から 4:58:21 の 8 秒間で寝返りをうった際の描画シ

システムと映像のスナップショットである。映像では、4:58:13 でベッドの右側で横を向いて被験者が、4:58:15~17 にかけて上を向きはじめ、4:58:17~19 にかけて膝を立てて真上を向いた仰向けとなっていた。4:58:21 に膝を伸ばして動作を終了する。このときの描画システムから、体が右から左へ移動したことがわかり、寝返りをうったことが推測できる。

その他の時間帯でも、キーの押下から推測できる姿勢は、実際の姿勢と一致していることがわかった。

3.3.2 他センサとの比較による評価

睡眠時における本システムと他センサとの比較評価実験を行った。比較対象のセンサには、タニタ社のスリープスキャンを選定した。これは、このセンサが [43] などの睡眠を計測する実験において多くの利用実績があるという理由である。

本システムを頭部とお尻付近に、スリープスキャンを腰付近に設置して併用した。計測時刻は AM2 時から 8 時までの 6 時間である。2 時すぎに入眠し、8 時前に起床した。その結果を図に示す。(a) のグラフは、頭部に設置したキーボードでの計測結果、(b) のグラフは体部に設置したキーボードでの計測結果である。(b) のグラフでは 3 つのキーボードでの取得データを結合してひとつのグラフに示している。それぞれのグラフはキー押下イベントを 1 つの点とした散布図となっており、補助のために各点間を直線で結んでいる。縦軸が押下したキーの位置の x 座標である。(a) では下限値が枕の左端、上限値が枕の右端となる。(b) では下限値がベッドの左端、上限値が枕の右端となる。(c) のグラフは、スリープスキャンで計測し分析した睡眠ログである。睡眠の深さが覚醒、REM、浅睡眠、深睡眠の 4 段階で表しているいずれのグラフも横軸は時間で、AM2 時から 8 時までの 6 時間のデータを示している。

(a),(b) の本システムのグラフでは、線が縦に伸びているところが体が大きく移動した、すなわち寝返りをうった時刻に相当する。一方、同様に線が横に伸びている時間帯は、体の動きがない状態で、深い眠りについていると考えられる。スリープスキャンでの睡眠の深さと体動の関係性は公開されていないが、体動が多いほど眠りが浅く、体動が少ないほど眠りが深いと判定していると推測する。

スリープスキャンのデータ分析では、継続して深い眠りについている時間帯が 2:40~2:50 頃と 5:50~6:30 頃の 2 回ある。これらのいずれの時間帯でも、本システムでは頭部及び体部が動いていないことがわかる。一方、ごく短い覚醒状態と継続した REM がそれぞれ 3 回ずつある。このときの体動をみると、寝返りをうった時刻と一致する。ただし、他の時間帯と比較して寝返りの回数が桁違いに多いわけではなく、本システムの計測データのみで眠りが浅いと単純に判定するのは難しい。

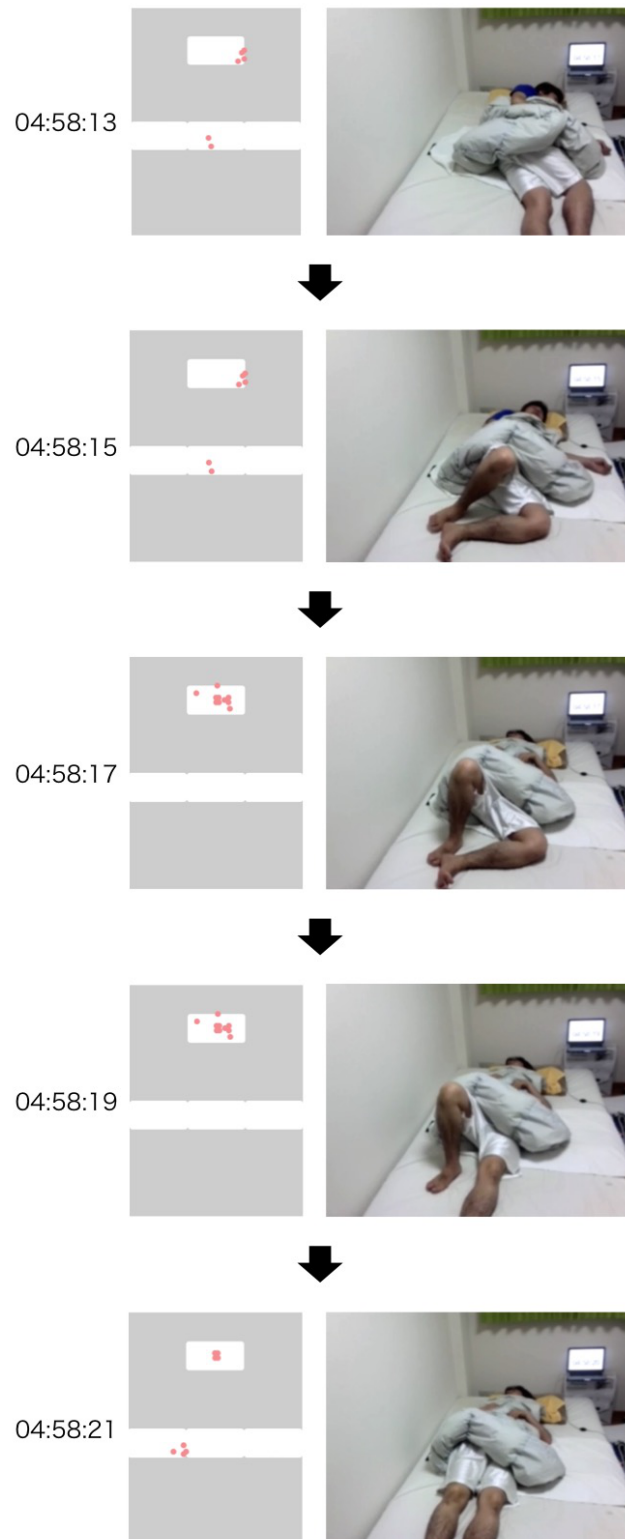


図 3.8. 睡眠時の寝返りの際のキー押下状況とその様子

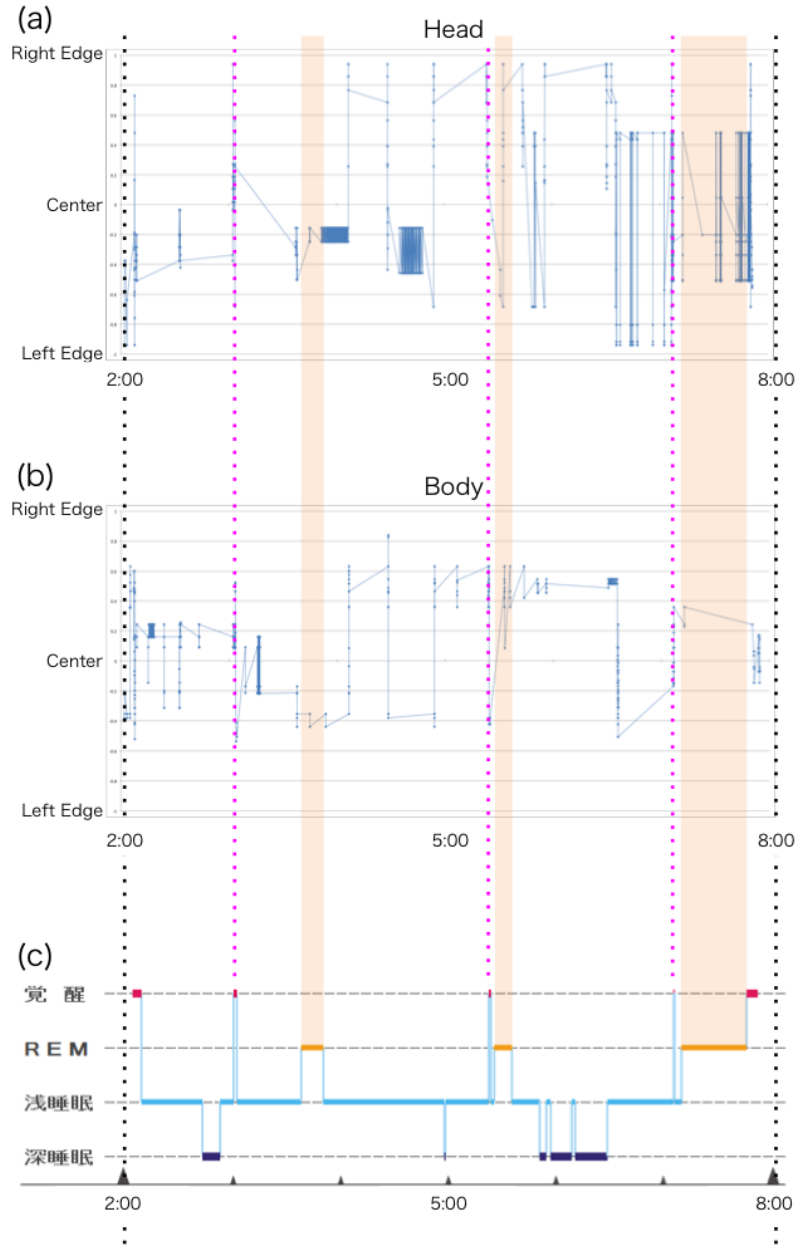


図 3.9. AM2 時から 8 時までの 6 時間一晩を通して計測した結果. (a) 頭部のキーボードの計測結果. 押下イベントを 1 つの点として散布図として示し, 補助のためにイベント間を直線で結んでいる. 縦軸が押下したキーの位置の x 座標. 横軸が時刻. (b) 体部のキーボードの計測結果. 3 つのキーボードでの取得データを結合してひとつのグラフに示している. その他のグラフの設定は (a) と同じ. (c) スリープスキャンで計測し分析した睡眠ログ. グラフはスリープスキャン付属のアプリケーションより引用. 縦軸は睡眠の深さを覚醒, REM, 浅睡眠, 深睡眠の 4 段階で表している. 横軸は時刻.

3.4 考察

カメラ撮影での映像との比較により、キー押下から推測される姿勢とよく一致することが明らかになった。その一方で、寝返りと推測できるキー押下イベントが、実際には手でキーボードを押しているというケースもあった。

また、スリープスキャンの分析との比較により、体がほとんど動かない深い睡眠の検出には有用であることがわかった。本研究ではまだ簡易的なデータ分析しか行っておらず、アルゴリズムの工夫などにより浅い睡眠の検出も可能となるかもしれない。

今回はスリープスキャン以外の既存技術との比較実験は行っていないが、ベッド上に置いて加速度を計測する方式や、ウェアラブルセンサーによる計測方式よりも、体圧を直接計測するスリープスキャン及び本システムの方が精度が高いと考えられる。また、映像撮影や Kinect などを用いて計測システムを構築した場合には体が布団に隠れてしまうため計測が難しいが、本システムでは問題なく測定できる。

3.5 おわりに

本研究では、市販のシリコン製キーボードを使用して、睡眠時に使用する姿勢センサを開発し、その有用性を検証した。安価な市販品を利用したことにより、低価格で高精度の睡眠センサとしての活用できることが見込まれる。睡眠に関する研究実や医療の現場など、成果を活用できる場所を模索したい。

今後は、取得したデータからより精度よく姿勢を推定する方法を継続して研究していきたい。使用事例を増やしてデータを蓄積できれば、データマイニングによる特徴検出についての研究が可能となる。姿勢推定の精度は、アルゴリズムによって改善できる余地が大いにありと考えている。また、姿勢のみならず、たとえば睡眠時無呼吸症候群 (SAS) を検出可能となれば、本システムの有用性は一気に増すと考えている。

第 4 章

人間の感情の情報化と可視化による 周囲への伝達: PICALA

4.1 はじめに

4.1.1 CPHI としての位置づけ

本節では、プレゼンテーションにおける聴講者の感情の周囲への共有に関する研究を通じ、CPHI の事例として考察する。聴講者の感情は、聴講者自身が PC やスマートフォンのブラウザを操作し選択する。ここは、CPHI の 4 パターンのうち Cyberization from Human にあたる。さらに、その感情は光という物理量を用いて照明色によって周囲へ伝える。これは、CPHI の 4 パターンのうち Realization to Human にあたる。このように、本節で提案する PICALA の CPHI モデルは Cyberization from Human と Realization to Human の組み合わせとなる (図 4.1).

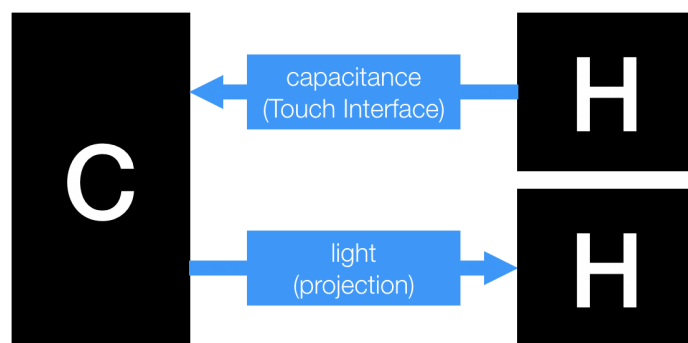


図 4.1. PICALA の CPHI モデル。上段の Cyberization from Human では、スマートフォン等のタッチインタフェースを想定し、使用する物理量を静電容量とした。

4.1.2 背景

学会発表や講演などのプレゼンテーションは、講演者が前に立ち、PCをプロジェクターに接続し、スクリーンにスライドを投影するという形式が定着して久しい。この形式での発表は、情報の伝達はほとんど講演者から聴講者への一方向となる。聴講者の感情が他の人に伝わる手段は拍手と笑い声に限られ、その場にいる聴講者同士で感情を共有することができない。この課題を解決するため、プレゼンテーションにおいて聴講者からのフィードバックを得るための様々な試みがなされてきた [44][45]。しかしいずれの手法も、入力が煩雑である、感情共有手法がプレゼンテーションを阻害するなどの課題を抱え、最善の手法は確立されていない。

本研究では、これらの先行研究で挙げられた課題を解消する手法として、スクリーン付近に色の変更が可能な照明を設置し、色に感情の意味を割り当て、照明色を変えることで聴講者が講演者や他の聴講者へ感情を共有するシステムを提案する。色を変えられる電球を用いてこのシステムのプロトタイプ実装を行い、3つの情報系研究会にて運用し、参加者に対してアンケートを実施してシステムの主観評価を行った。加えて、既存の手法との比較による定性評価を行った。

4.2 設計と実装

4.2.1 概要

PICALAの基本要素は、ユーザである聴講者がボタンを押下すると、プレゼンテーションのステージ上の電球の色が変化するシステムである。機器およびネットワークの構成は、LANモデルとインターネットモデルという2種類の構成を設計した。これらの実装はオープンソースソフトウェアとしてGitHubにソースコードを公開している*1*2。

4.2.2 照明

照明には、webAPIで電球の色相、彩度、明度を変えられる Philips社の hue*3を使用した。hueの電球の制御は、hueブリッジ経由で行われる。hueブリッジではwebサーバが稼働し、webAPIを備えている。APIアクセスに従ってhueブリッジは電球に制御コマンドを送信する。電球へのコマンド送信は ZigBee Light Linkによる無線通信が用いられる。実証実験では、3つの電球をクリップ式の電球ソケットにとりつけた(図4.2)。

*1 <https://github.com/yumu19/radihue>

*2 <https://github.com/yumu19/picala>

*3 hue: <http://www2.meethue.com/>

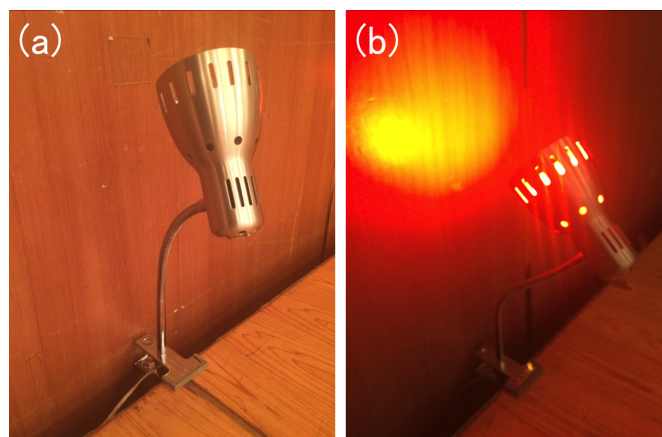


図 4.2. PICALA の設置に用いたクリップ式電球ソケット. (a) 消灯時, (b) 点灯時

4.2.3 ユーザインタフェース

ボタン押下は、ユーザが保有する PC やスマートフォンの web ブラウザ上で行う。ユーザは web ブラウザから特定の IP アドレスにアクセスし、表示されたボタンインタフェース (図 4.3) を開きながらプレゼンテーションを聴講する。ボタンは、「へえ〜(共感・納得)」「すごい(驚き・賞賛)」「笑(愉快)」「?(疑問)」の4つの感情を備える。ボタンの色はそれぞれ、へえ〜はオレンジ色、すごいのは赤色、笑は緑色、?は青色とした。これは、盛り上がりを暖色、盛り下がり寒色とし、「笑」は面白いことを表すネットスラングの「草が生える」に由来して緑とした。プレゼンテーションを聴講している際に何らかの感情が生じたときに、ユーザはいずれかのボタンを押下する。



図 4.3. PICALA の聴講者用ボタンのユーザインタフェース. (a)PC ブラウザ (b) スマートフォン

4.2.4 LAN モデルの構成

LAN モデルの機器構成およびネットワーク構成を図 4.4 に示す。LAN モデルは、会場に共用可能なネットワークが敷設されている場合を想定し、聴講者の PC・スマートフォン、サーバ、hue ブリッジを同一 LAN 内に設置する。ボタンを押下すると、PICALA サーバへ http アクセスを行う。ボタンの押下を受信する http サーバは node.js により実装した。このアクセスのパラメータには押下したボタンの種類が含まれる。受信すると、ボタンの種類に応じてパラメータを設定して hue ブリッジに web アクセスする。PICALA サーバと hue ブリッジはそれぞれ固定 IP アドレスを持つ。hue の電球へのアクセスは、hue の標準機能を使用し、hue ブリッジの webAPI 経由でアクセスする。

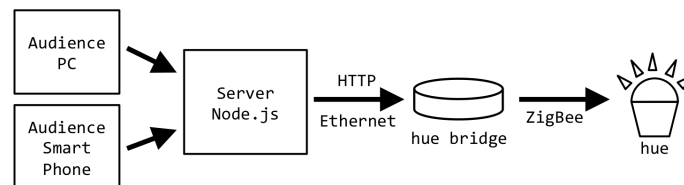


図 4.4. PICALA のシステム構成図 (LAN モデル)

4.2.5 インターネットモデルの構成

インターネットモデルの機器構成およびネットワーク構成を図 4.5 に示す。インターネットモデルでは、グローバル IP アドレスが付与されインターネットからの到達性を持つサーバを設置し、会場に共用可能なネットワークがない場合でも聴講者が各自で契約する LTE 回線経由で使用することができる。本研究のシステム実装ではサーバは Amazon EC2 を利用して設置した。聴講者の PC・スマートフォンからサーバへの通信は LAN モデルと同じである。hue ブリッジは講演会場に設置されるため、インターネットからのアクセスを可能とすることが難しい。そのため、サーバから hue ブリッジへの通信は、通信の橋渡しを行うブリッジ PC を経由する。ブリッジ PC のブラウザでは、サーバのブリッジ用 web ページを開いておく。このブリッジ用 web ページには websocket が組み込まれており、サーバへの API アクセスが生じた際に即時にブリッジ PC のブラウザ上 (クライアントサイド) で JavaScript を実行して hue ブリッジの API にアクセスする。websocket の実装は、Node.js で使用する JavaScript フレームワークの socket.io を利用した。

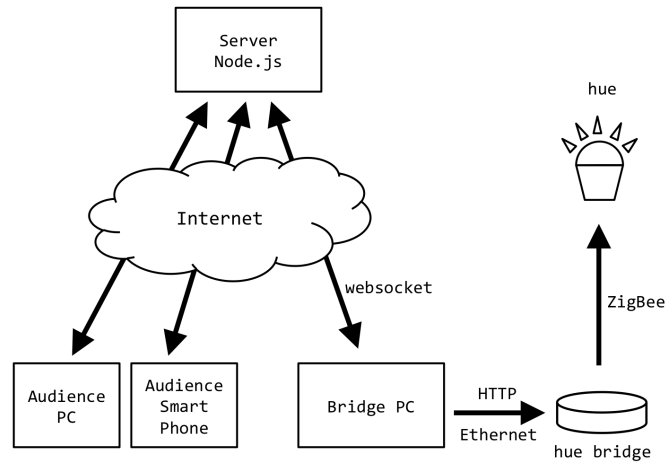


図 4.5. PICALA のシステム構成図 (インターネットモデル)

4.2.6 照明点灯アルゴリズム

本システムでは、ユーザの識別などは行っておらず、単純に http のアクセスのみで電球の制御を行う。一人がボタンを連打しても、大勢が一斉にボタンを押下しても、出力される結果は変わらない。通常時、電球は消灯*4し、ボタン押下すると、電球をボタンに応じた色に点灯する。点灯時間は 5 秒*5。

3つの電球を用意し、ボタン押下1回ごとに1つの電球の色を変えた。電球をそれぞれ A, B, C とすると、最初のボタン押下では電球 A に反映される。次のボタン押下では、電球 A の色はそのまま、電球 B に反映される。同様にその次のボタン押下では電球 A と電球 B の色はそのまま、電球 C に反映される。4 回目のボタン押下時は再び電球 A に反映される。このとき電球 A がまだ点灯している場合にも、色の設定を上書きする。

4.2.7 遅延時間の評価

聴講者がボタンを押下してから照明に反映されるまでの遅延時間を計測し、LAN モデルとインターネットモデルの遅延を比較した。計測に用いたシステム仕様を 4.1 に示す。経路 1 はスマートフォン-サーバ間、経路 2 はサーバ-ブリッジ PC 間の通信の経路を示す。無線 LAN は 802.11a でアクセスポイント AtermWR8700 を使用した。計測では、左手にストップウォッチを持ち、右手に持ったスマートフォンのブラウザからボタンを押下した、照明点灯は目視により確認し、遅延時間を測った。この計測をそれぞれのモデルで 30 回ずつ計測した、計測結果を 4.6 に示す。LAN モデルの遅延は平均 0.828 秒 (標準誤差 0.017 秒)、インターネットモデルの遅延は平均 1.368 秒 (標準誤差 0.069 秒) であった。LAN モデルの方が遅

*4 実際には hue の設定値において明るさ最小に設定

*5 WISS2014 での実証実験の一部では点灯時間を 1 秒を変更した。

延が小さいものの、インターネットモデルの遅延も十分小さく、どちらのモデルでも実用に大きな問題ないといえる。

表 4.1. 遅延計測の環境

	LAN モデル	インターネットモデル
サーバ OS	OS X 10.10.5	Ubuntu 14.04.2 LTS
サーバ仕様	MacBook Pro Retina 13-inch Late 2013 (2.4GHz Core i5)	Amazon EC2 t2.micro
Node.js	0.10.32	0.10.25
経路 1	無線 LAN	LTE(docomo)
経路 2	-	フレッツ光 + 無線 LAN

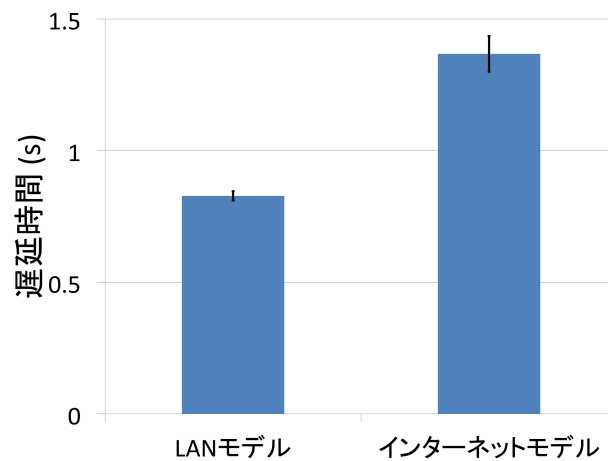


図 4.6. LAN モデルとインターネットモデルにおける、ボタン押下から照明点灯までの遅延時間の計測結果。エラーバーは標準誤差を示す。

4.3 実証実験

本システムを、第 22 回インタラクティブシステムとソフトウェアに関するワークショップ (WISS2014)*⁶、第 48 回情報科学若手の会 (wakate2015)*⁷、エンタテインメントコンピューティング 2015(EC2015)*⁸の 3 つの情報系研究会にて稼働し、実証実験を行った。

*⁶ <http://www.wiss.org/WISS2014/>

*⁷ <http://wakate.org/>

*⁸ <http://ec2015.entcomp.org/>

4.3.1 WISS2014

第22回インタラクティブシステムとソフトウェアに関するワークショップ(WISS2014)は2014年11月26日(水)~28日(金)の3日間、浜名湖ロイヤルホテルにて開催された。WISSでは、会議の場自体をよりインタラクティブにしていくWISS Challenge[46]という試みが行われており、本システムの実証実験もその一貫として実施した。WISS2014では3日間で招待講演を含めて合計22の口頭発表講演が実施された。口頭発表はすべてシングルトラックで開催され、聴講者数は約200名であった。設置位置は、ステージ上のスクリーン下(図4.7(a))と演台(図4.7(b))の2通りを試した。

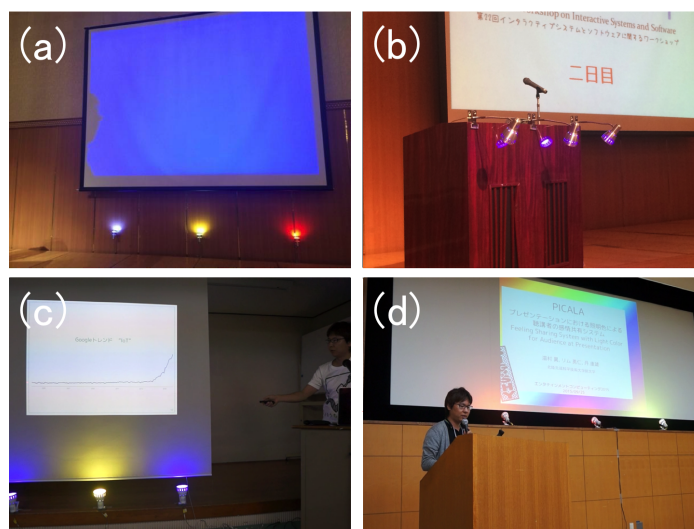


図4.7. 実証実験で設置したPICALA。(a)WISS2014でスクリーン下に設置した様子。(b)WISS2014で演台に設置した様子。(c)wakate2015でスクリーン下に設置した様子。(d)EC2015でスクリーン下に設置した様子。

4.3.2 第48回情報科学若手の会

第48回情報科学若手の会(wakate2015)は2015年9月19日(土)~21日(月)の3日間、静岡県伊東市の山喜旅館にて開催された。情報科学若手の会は、20代を中心とした合宿形式の若手情報系研究者の集まりで、3日間で、1時間の招待講演、40分の若手特別講演の他、40分の一般講演が8件、25分のショート講演が7件、5分間のライトニングトークが10件実施された。参加者は50名程度であった。電球はスクリーンの直下に設置した(図4.7(c))

4.3.3 EC2015

エンタテインメントコンピューティング2015(EC2015)は、2015年9月25(金)~27(日)の3日間、札幌市教育文化会館および北海道大学学術交流会館で開催された。PICALAは、

EC2015 の初日に開催されたオーガナイズドセッション SIGSHY にて使用された。セッション時間は 100 分で、7 名の発表者が 10 分のショート発表または 20 分のロング発表を行った。聴講者は 50 名程度。セッションの冒頭に PICALA の説明を行った。スクリーンは地上から 1.5m 程度の高さに設置され、電球はスクリーンの直下に設置した (図 4.7(d))

4.4 ログ解析

実証実験を通じて全てのボタン押下の時刻と種類を記録した。各研究会でボタンの押された割合の比較を図 4.8 に示す。どのボタンもバランスよく押されているが、4 つのボタンのうち唯一ネガティブな意味合いの「？」は、他のボタンと比較して押下数がやや少ない傾向にある。

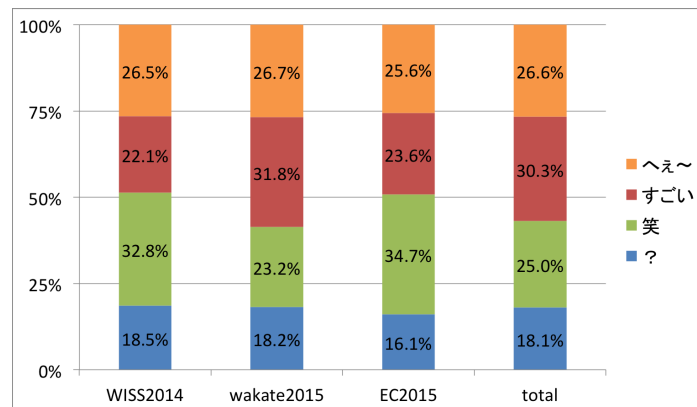


図 4.8. 各研究会でボタンの押された割合の比較。積み上げ式の棒グラフにて表示。

ある講演におけるボタン押下記録のグラフを、図 4.9 に示す。グラフは、時間を 10 秒に区切り、10 秒間に各ボタンが押下された回数を示している。ボタンの押下は常時まんべんなく発生しているわけではなく、特定の瞬間にほぼ同時に頻繁に押される傾向にあり、グラフにはいくつものピークが存在する。

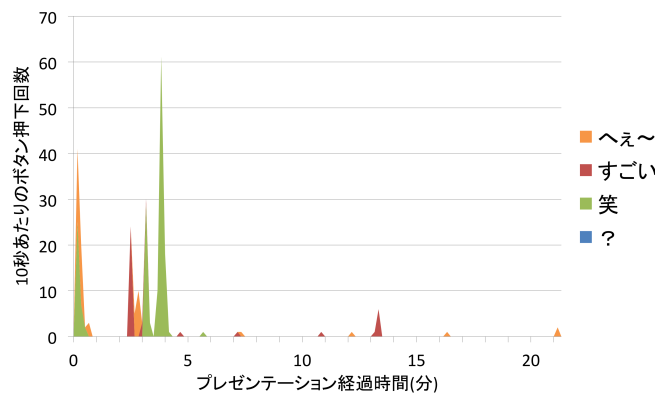


図 4.9. ある講演におけるボタン押下の記録。時間を 10 秒ごとに区切ってボタン押下回数をカウント。積み上げ式の折れ線グラフにて表示。

表 4.2. 使用者へのアンケートの質問項目

番号	質問	選択肢
Q1	PICALA のシステムは便利でしたか？	1. 便利でない ~ 5. 便利 (5 段階)
Q2	PICALA が便利/便利じゃない理由があれば教えてください	自由記述
Q3	PICALA の使い方はわかりましたか？	1. わかりにくい ~ 5. わかりやすい (5 段階)
Q4	PICALA がわかりやすかった/わかりにくかった点があれば教えてください	自由記述
Q5	PICALA は使いやすかったですか？	1. 使いにくい ~ 5. 使いやすい (5 段階)
Q6	PICALA の使いやすかった/使いにくかった点があれば教えてください	自由記述
Q7	PICALA は見やすかったですか？	1. 見にくかった ~ 5. 見やすかった (5 段階)
Q8	PICALA が見やすかった/見にくかった理由があれば教えてください	自由記述
Q9	PICALA はプレゼンを見るのに邪魔でしたか？	1. 邪魔だった ~ 5. 邪魔じゃない (5 段階)
Q10	プレゼンを見るのに邪魔だった理由があれば教えてください	自由記述
Q11	【講演者限定】PICALA は講演に邪魔でしたか？	1. 邪魔だった ~ 5. 邪魔じゃない (5 段階)
Q12	【講演者限定】PICALA を講演中に見ることができましたか？	見ることができた/何度か見ることができた/全く見ることができなかった
Q13	PICALA の感想があれば教えてください	自由記述
Q14	PICALA への要望や期待することがあれば教えてください	自由記述

4.5 主観評価

PICALA のユーザによる主観評価を行うため、WISS2014, wakate2015, EC2015 の参加者に対してのアンケートを実施し、結果をまとめた。

4.5.1 アンケート概要

アンケートは、Google フォームを利用した web での入力にて実施した。回答者数は、WISS2014 で 23 名、wakate2015 で 21 名、EC2015 で 14 名であった。アンケートの質問項目を表 4.2 に示す。アンケートの項目には選択式と自由記述式の両方がある。選択式の質問のうち、わかりやすさなど、程度を問うものについては 5 段階リッカート尺度とした。質問項目はすべて任意回答とした。Q3~Q6 は WISS2014 のみの質問で、一方、Q7~Q8 は wakate2015 と EC2015 のみの質問である。また、WISS2014 開催時のシステム名は「ラジ hue」だったが、もともとの質問文でラジ hue と記述していた部分を本論文では PICALA と記述している。アンケートの回答のうち、選択式質問の回答結果を図 4.10 に示す。

本論文において、アンケートの自由記述回答文の引用は、文中から必要な箇所のみを抜き出した、元の文の体裁を極力維持しているが、語尾や句読点は整形し、明らかな誤字は修正した。

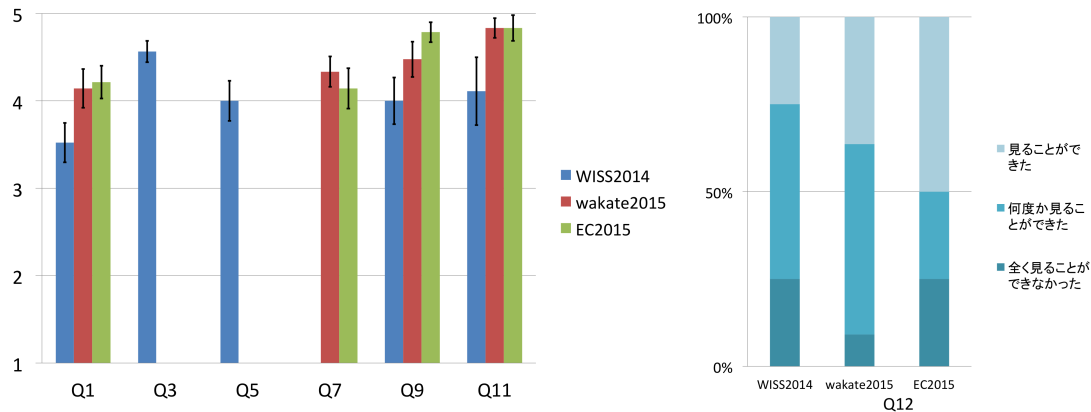


図 4.10. アンケートの選択式設問の回答結果. エラーバーは標準誤差を示す.
 $n=23$ (WISS2014), $n=21$ (wakate2015), $n=14$ (EC2015)

4.5.2 参加者間の感情共有

PICALA の主目的である感情共有について、アンケートからは「twitter のタイムラインよりも会場の一体感が得られた」「同じタイミングで同じ反応をしている人がいると満足感がある」「他の人がどう思っているのかがリアルタイムにわかるのは良い」といった感想が挙がり、聴講者同士での感情の共有ができていたことがわかった。また、聴講者と講演者のインタラクションについては、「講演中に簡単な感想を伝えられるのがとても良い」「発表相手に対してフィードバックを送れるのがよかった」といった、講演者へのフィードバックができることに好感を示す意見が挙げられた。また、「匿名で反応を返せるので正直に伝えやすい」といった、匿名性に利便性を感じている使用者もいた、フィードバックの匿名性は、チャットや Twitter にはない PICALA 独自の特徴である。

講演者側も、フィードバックを受け取ることができることに「リアクションがわかって嬉しかった」「発表中にみんなの気持ちがわかったという意見が挙げられ、発表中のフィードバックが実施されていることが確認できた。さらに、一部の講演者は「会場の反応を見て、発表者が説明を変えられることが便利だと思いました。」と、PICALA を活用したプレゼンテーションを実践している者もいた。

4.5.3 プレゼンテーションの阻害

PICALA の使用で最も懸念していたのは、光によりプレゼンテーションを阻害することである。Q9「PICALA はプレゼンを見るのに邪魔でしたか？」(1. 邪魔だった ~ 5. 邪魔じゃない) は 5 段階評価の回答は平均値 4.362(標準誤差 0.136) であり、この結果からプレゼンテーションを阻害してないと言える。「プレゼン中の声を妨げない点は、笑い声や拍手よりも優れていると感じた。」「画面上にコメントを流すような形式と違い、注目を強制されないのでプレ

ゼンの流れを阻害されなくてよかったです。」という好感触の感想もあった。ただし、プレゼンテーションを阻害していると感じていた聴講者も皆無ではなく、「ちかちかがうるさい」「プレゼンテーションの内容自体が難解で理解をしたいときに集中を削がれた場合があった」「プレゼンの視認性が落ちる事が何度かあった」「前の方の席だとちょっとまぶしかった」といった感想も挙げられた。

Q11「【講演者限定】PICALAは講演に邪魔でしたか?」(1. 邪魔だった ~ 5. 邪魔じゃない)は5段階評価の回答は平均値 4.600(標準誤差 0.141)であり、講演者に対してもプレゼンテーションの阻害はほとんどないと言える、

4.5.4 使いやすさ

Q3「PICALAの使い方はわかりましたか?」(1. わかりにくい ~ 5. わかりやすい)の5段階評価の回答は平均値 4.565(標準誤差 0.123)であり、全回答が4以上だったため、インタフェース設計については特に問題はなかったと言える。Q5「PICALAは使いやすかったですか?」(1. 使いにくい ~ 5. 使いやすい)の5段階評価の回答は平均値 4.000(標準誤差 0.229)であり、概ね使いやすかったという評価を得た。

4.6 考察

実証実験でのアンケート結果より、本システムが聴講者にとって有用であること、本研究の狙いどおり、聴講者から発表者へのフィードバックを実施できていることを明らかにした。

その一方、いくつかの課題が明らかになった。ひとつは照明色とその意味の対応付けである。ボタンの色と照明の色が同じであるため、頻繁にボタンを押下している聴講者であればすぐに覚えられるが、そうではない聴講者にとってはわかりにくい面がある。

もうひとつは、照明点灯中にボタンを押下した際に、自分のボタン押下が反映されたかどうかわかりにくいことである。これは、照明の数が増えれば解決できるであろう。本システムで使用したLED照明 hue は、潤沢に使用できるほど電球の価格は安くなく、さらに安価なLEDを用いて照明システムを構築することも検討の余地がある。

4.7 関連研究

プレゼンテーションにおいて、聴講者からのフィードバックを得るための様々な試みがなされてきた [44][45]。チャットを用いて聴講者間で意見交換しサブスクリーンに表示する手法は、RekimotoらがWISS'97で実施したのが先駆けであった [47]。この手法はDigital Backchannelsと呼ばれ [48]、今日まで、Backchan.nl [49] など、様々な手法が検討された [50]、近年ではTwitterを用いる方法 [51] も広まっている。

動画共有サイトのニコニコ動画 [52] では、動画視聴者が投稿したコメントが画面上を流れて動画投稿者や他の視聴者とのインタラクションを生み出している。このニコニコ動画のコメン

表 4.3. 関連研究との比較. 非侵襲性とは各手法がプレゼンテーションを阻害しないことを示す.

手法	入力手段	表現手段	非侵襲性	入力の容易さ	表現の多様性
チャット	文字入力	文字	○	△	△
ニコニコ動画	文字入力	文字	×	△	△
井上ら [54]	複合 UI(ボタン・文字・手書き)	自由描画	×	×	○
Borderless Canvas	手書き	自由描画	○	×	○
ラジへえ	ボタン選択	音	×	○	×
PICALA	ボタン選択	色	○	○	×

トを現実世界に拡張し、四方の壁面と天井が LED ディスプレイで構成されて会場全体にコメントが流れるニコファーレという施設も誕生した。このニコファーレを活用したニコニコ学会βという学会形式のイベントも開催された [53]。

これらのテキスト入力によるインタラクションは、自分の感情を文字に起こして投稿する必要があるため、投稿の敷居が高いことが課題である。また、聴講者にとっても、聴講しながらテキストを読むのは大変で、表示する場所によっては聴講の妨げになる場合もある。

チャット以外の手法でプレゼンテーションをインタラクティブにする試みも行われてきた。井上ら [54] は、聴講者が専用クライアントを操作して、プレゼンテーション中にリアルタイムにフィードバックできるシステムを提案した。Kurihara らの Borderless Canvas [55] は、聴講者の手元の端末に書き込んだ内容がサブスクリーンに表示され、聴講者を巻き込んだ議論を活発にするインタラクティブなシステムである。

加藤らのラジへえ [56] は、Web 上のボタン入力により数種類の声の効果音を流すことで感情をその場にいる講演者および聴講者全員に伝えるためのシステムである。ラジへえを用いればプレゼンテーションにおいて聴講者が感じた感想を瞬時に共有することができる。その一方で、感情を共有する度に効果音が流れるため、プレゼンテーションの妨げとなる可能性もある。

関連研究と本研究の比較を表 6.1 にまとめた。

4.8 おわりに

本研究では、照明の色を用いて聴講者の感情を表現する手法を提案し、設計と実装を行った。会場 LAN を使う場合とインターネット上のサーバを使う場合の 2 種類の設計を行い、どちらの場合も、ボタン押下から照明点灯までの遅延は非常に短い時間であった。PICALA を WISS2014, 第 48 回情報科学若手の会, EC2015 の 3 つの情報系研究会にて実証実験を行った。実証実験ではユーザへのアンケート調査による主観評価を行い、PICALA が目的としていた聴講者間および聴講者と講演者の感情共有ができていていること、懸念であったプレゼンテーションの阻害もそれほど影響がないことを確認した。

今後は、研究会でのプレゼンテーション以外での活用も目指す。アンケートでは授業で使いたいという意見も挙がっており、実証実験などで可能性を模索する。また、照明の色を用いた表現は、舞台演出で長年用いられている手法である。長年培われた舞台照明演出の技術を参考

40 第4章 人間の感情の情報化と可視化による周囲への伝達: PICALA

にし、よりよい照明の活用方法を検討する。PICALA ではスクリーン下に電球を設置したが、PC上でソフトウェアを稼働させ、スクリーン内に表示させるような手法も考えうる、

第 5 章

物理空間への重畳演出による人間の感情への干渉: Augmented Typing

5.1 はじめに

5.1.1 CPHI としての位置づけ

本節では、キーボードの打鍵にあわせて映像演出と音響演出を付与することでタイピング体験を拡張する研究を通じ、CPHI の事例として考察する。もともと、キーボードによるタイピングは、人間が情報を入力しコンピュータに伝える Cyberization from Human である。これに加えて、Augmented Typing では、タイピングに合わせて映像演出と音響演出を付与する。この演出は、物理的に存在するキーボードという機器に光と音を重畳する Physical Augmentation である。このように、本節で提案する PICALA の CPHI モデルは Cyberization from Human と Physical Augmentation の組み合わせとなる (図 5.1)。

5.1.2 背景

PC の文字入力デバイスは、キーボードが長年の主流である。キーボードは毎日長時間使用するもののため、こだわりを持って使用している人も多く、キー配列、押下時の反発力、

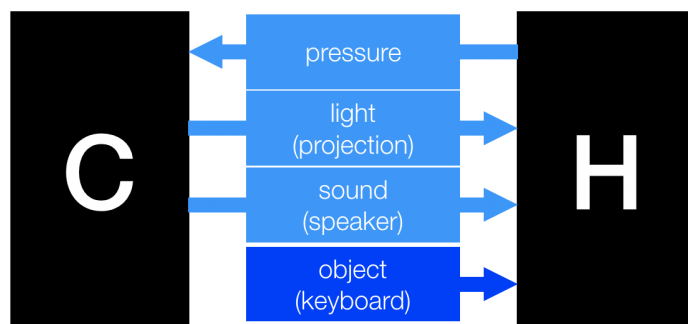


図 5.1. Augmented Typing の CPHI モデル.

ストロークの深さ、打鍵音などのさまざまな条件から自分の好みにあったものを使用している。ソフトウェアエンジニアの間では特にその傾向が強く、Happy Hacking Keyboard[57]やKinesis Contoured Keyboard[58]などの特徴的なキーボードを選択する人も多い。また、キートップを入れ替えたり、カバーを取り付けたりして自分好みにアレンジして使用することもある。

PCやスマートフォンでは、待ち受け画面の背景に好きな画像を設定したり、電話やメールの着信音を好きな音楽に設定するなど、ソフトウェアの面でのアレンジが普及している。また、ソフトウェアエンジニアが特によく使用する Emacs や Vim といったテキストエディタでは、設定ファイルを編集して見た目や機能をアレンジして使うことも一般的である。キーボードの特性やアレンジは、ハードウェアに依存しているが、ソフトウェアの面でアレンジすることができれば、より好みに合わせたアレンジが可能となる。例えば、打鍵にあわせて音声ファイルを再生して打鍵音として好きな音を鳴らしたり、キーボード上に効果映像を表示したりするような新たなアレンジの方法も生まれる。

そこで、本研究では、キーボードに映像演出と音響演出を加えることによって、タイピング体験を拡張するシステム Augmented Typing を提案する。Augmented Typing では、キーボードの打鍵に合わせて映像演出と音響演出を付与することによってタイピング体験を拡張する。

Augmented Typing のプロトタイプとして、プロジェクションマッピングを用いた映像演出と効果音を再生する音響演出を行うシステムを実装した。システムの効果を評価するため、演出に対する印象評価と、効果映像をアレンジする際の法則性の調査を行った。

5.2 関連研究

5.2.1 映像表示によるインタフェース拡張

キーボード上に映像を表示するデバイスおよびシステムは、過去にも提案され、市販されているものもある。Optimus[59]は、キートップに有機 EL ディスプレイが埋め込まれており、文字の言語を変えたり好きなアイコンを表示させたりすることができる。

Block et al.[60]は、通常の文字入力ではない機能をキーに与え、キーボードへのプロジェクションマッピングを行うことにより、入力インタフェースとしてのキーボードの機能拡張を行った。

MacBook Pro には、2016年モデルより Touch Bar[61]が搭載された。Touch Bar は、既存のキーボードやタッチパネル式ディスプレイとは異なり、限られた表示機能と入力機能を有し、使用するアプリケーションなどの状況によって機能を変化させる。

5.2.2 演出付与による体験拡張

本研究のプロトタイプ実装で使用するプロジェクションマッピングは、表示機能のない物体や建物の側面に映像を投影し、様々な演出を行う手法である。プロジェクションマッピング

は、鑑賞するだけで楽しめる演出として用いられることが多いが、体験を拡張する手法としても用いられる。

映像装置としてのピアノ [62] では、ピアノの鍵盤の上部と下部のスクリーンに投影した映像に合わせてピアノが演奏されたり、演奏に合わせて映像が変わったりする演出手法が実現された。近年では、特定の視点から立体的に見えるような錯視を利用した映像をピアノに投影する事例もある [63]。卓球でラリーやバウンドにあわせて演出を付与する PingPongPlus[64] や、ビリヤードでショットにあわせて演出を付与する OpenPool[65] といった、スポーツでの体験拡張にも利用される。

打鍵に合わせて文字が飛び出る演出をプロジェクションマッピングで実装した Magic Keyboard[66] や、VR 空間で実装した Visual IO[67] など、キーボードへの映像付与によりタイピング体験の拡張を行った事例もある。

5.2.3 演出付与による学習支援と作業支援

プロジェクションマッピングは、様々な技能の習得や作業支援にも利用される。Takegawa et al.[68] は、ピアノの習熟のため、鍵盤へのプロジェクションマッピングを行い、運指の監視によりピアノの習熟への効果を調べた。同じくピアノの習熟を目的とした P.I.A.N.O.[69] も、鍵盤および天板へのプロジェクションマッピングを行うシステムである。音楽ゲームのように、鍵盤に近づいてくる譜（矩形の領域）にタイミングを合わせて打鍵することで正しく演奏することができる。土井と宮下 [70] は、プロジェクションマッピングを用いて箏の演奏を支援するシステムを提案した。Favnavi[71] は、机上での作業を真上から映像で記録し、別の人の作業実施時には机上に投影することで、作業手順の再現と伝達を容易にしている。Projectron mapping[72] は、プロジェクションマッピングで電子部品を仮想的に光らせることによって電子回路の学習を行うシステムを提案した。プロジェクションマッピングは、立体物の造作支援にも使われる。Yoshida et al.[73] は建築物の造作支援に用い、トイブロックの組み立て支援を行う StudI/O[74] や、粘土の造作支援を行う CrayMore[75] など、支援対象も様々である。

5.3 Augmented Typing

キーボードに映像と音の演出を付与することにより、タイピングの体験を拡張することを Augmented Typing と名付ける。PingPongPlus[64] や OpenPool[65] がスポーツに演出効果を付与して盛り上げたように、キーボードの打鍵に演出効果を付与し、タイピングという単調な作業に高揚感や心地よさを加えることが Augmented Typing が実現したいコンセプトである。

打鍵にあわせて音声ファイルを再生して打鍵音として好きな音を鳴らしたり、キーボード上に効果映像を表示したりするような新たなアレンジの方法も生まれる。

映像演出を付与すれば、打鍵にあわせてキーボードが光ったり、押下したキーを中心に波紋が広がったり、オブジェクトを撃ち出したりすることができ、タイピングで高揚感を得ること

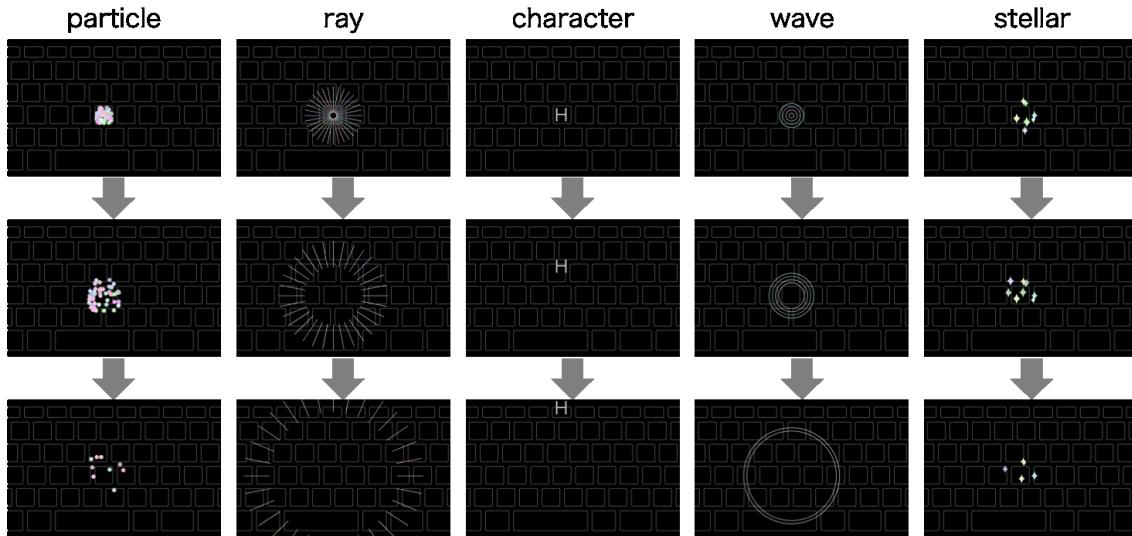


図 5.2. 5 種類の映像効果. 3 枚の図は効果の時間経過を示し, それぞれ上から下へ進む.

ができる. 音響演出を付与すれば, 打鍵に SF 風の効果音を加えて自分の好きな世界観に没入したり, 動物の鳴き声を加えて癒やしを得たりすることができる. さらに, 自分の好みにあった演出効果を選択して世界にひとつの自分のキーボードをつくりあげることによって, 所有欲も満たすことができる. 設定ファイルをオープンソースとして公開したり, 公開された他人の設定ファイルをさらに自分なりにアレンジし直すといったこともできるようになる.

Augmented Typing の概念を実現するためには, キーボードという入力装置に表示機能を付与する必要がある. 本研究でのプロトタイプ実装は, 次節で解説する通り, キーボードの上からプロジェクターで投影するプロジェクションマッピング方式を用いる. しかしこれは, Augmented Typing のコンセプトを実現する実装方法のひとつにすぎない. キーボードのキートップにディスプレイを搭載して映像演出を行うという方法や, キートップのみではなくキーボード全体をタッチパネル式ディスプレイにする方法も考えられる. Microsoft HoloLens[76] のような透過型ヘッドマウントディスプレイを使い, 実空間には何も表示せず, タイピングしている人だけが演出を体感できるという手法も考えられる.

5.4 実装

5.4.1 映像演出

Augmented Typing では, キーの押下に合わせてキーの周辺に効果映像を表示する. 効果映像は「粒子が飛び散る (particle)」「放射状の光線が出る (ray)」「文字が前方に飛び出す (character)」「波紋が広がる (wave)」「キー周辺に輝きが表示される (stellar)」の 5 種類を用意した. それぞれの効果映像は図 5.2 に示す. 通常時はキーの枠のみを表示し, キー押下時に追加で効果を描画する. 各演出で表示されるオブジェクトの色は, 投影時に目立つような白に近い色をランダムで選択した.

表 5.1. 効果音の音源.

種類	URL
keyboard	https://on-jin.com/o/100
typewriter	https://on-jin.com/o/201
digital1	http://www.kurage-kosho.info/mp3/button37.mp3
digital2	http://www.kurage-kosho.info/mp3/button38.mp3
wadaiko	http://soundeffect-lab.info/sound/anime/mp3/drum-japanese1.mp3
cat	http://pocket-se.info/se/cat3.mp3

5.4.2 音響演出

タイピング時の音響演出として、キーの押下に合わせて効果音を再生する。効果音は「キーボードのタイプ音 (keyboard)」「タイプライターのタイプ音 (typewriter)」「低音の電子音 (digital1)」「高音の電子音 (digital2)」「和太鼓 (wadaiko)」「猫の鳴き声 (cat)」の6種類を用意した。音源は、すべてフリー素材として配布されているもの(表 5.1)を利用し、再生時間の長い音源はタイピングに合うようにトリミングした。

5.4.3 ソフトウェア

キー押下にあわせて映像演出の描画と音響演出の再生をおこなうシステムを、JavaScriptを用い、1つのWebページとして実装した。Webページとして実装する利点は2つある。ひとつは、利用する場合に特定のOSや特別なソフトウェアを必要とせずJavaScriptが動作するブラウザがあればよいこと。もうひとつは、Webサーバで公開しておけばインターネットに接続可能であればどこからでも利用可能ということである。ソフトウェアの実装は、GitHubにてソースコードを公開しており^{*1}、Webページとしてもアクセス可能である^{*2}。

映像演出の記述には、JavaScriptライブラリ p5.js を用いた。p5.js は、Processing と同等の記法で記述できるライブラリで、図形の描画が簡単にできるため演出映像の記述に向いている。キーの押下は p5.js の keyPressed() 関数の呼び出しでフックできる。映像のフレームレートは 30fps に設定した。

演出を描画する部分の他に、ページ最上部にコントロールパネルを配置した(図 5.3)。コントロールパネルは html のフォームのパーツで構成され、キーボードおよびマウスで操作することができる。後述する評価実験で使用するため、映像と効果音の種類の選択と、映像演出のいくつかのパラメータのスライダーでの調整を可能とした。

^{*1} <https://github.com/yumu19/augmentedtyping>

^{*2} <https://yumu19.github.io/augmentedtyping/>

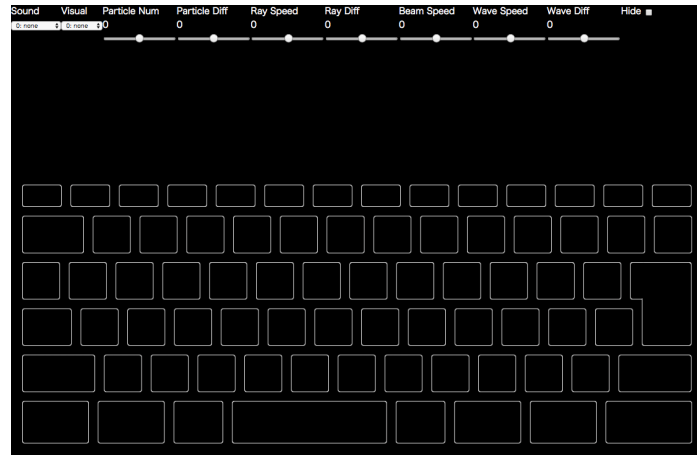


図 5.3. Web ページとして実装した演出システム.

5.4.4 ハードウェア

上述の演出システムを表示した PC をプロジェクタに接続し、プロジェクタを三脚で固定し、キーボードの真上から映像を投影する (図 5.4). キーボードは PC に USB 接続される. 効果音は PC から再生される. キーボードは Apple Keyboard テンキー付き JIS, PC は MacBook Pro Retina, プロジェクタは Dell M110(明るさは 300ANSI ルーメン) を使用した (図 5.5,5.6). 投影時の解像度は 1280×720 である.

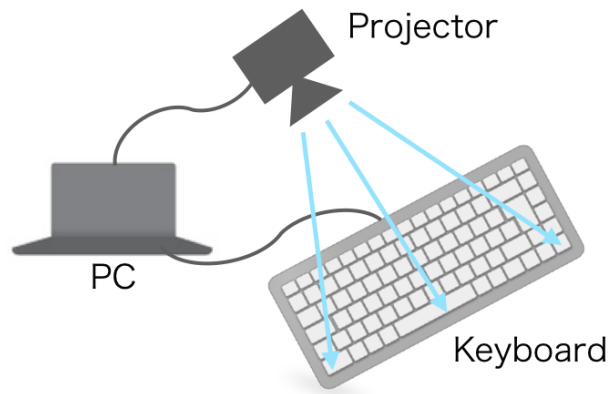


図 5.4. システムのハードウェア構成



図 5.5. システム構成の全体像

5.5 評価実験

5.5.1 実験概要

演出付与がタイピングに与える影響を明らかにするため、評価実験を行った。実験内容は「印象評価」と「パラメータ最適化」の2つである。印象評価では、SD法を用いて7つの印象評価軸に沿って印象を選択する。パラメータ最適化では、効果映像のパラメータを調整し、最も心地よいと思ったパラメータの数値を選択する。被験者は、Augmented Typingを用いたタイピングを体験しながら、アンケート用紙に回答を記入する。上記に加え、全ての効果映像および効果音について、アンケートの自由記述欄を設け、印象評価軸にとどまらない印象についての意見聴取を行った。この自由記述欄への記入は任意とした。

実験は、情報系の大学生および大学院生11名を対象とし、大学内の学生居室で実施した。映像演出と音響演出の評価を独立して行うため、映像演出の評価時には無音、音響演出の評価時には効果映像なしとした。音響演出の印象評価時には、ヘッドホンを着用した。

5.5.2 印象評価

前節で紹介した5つの効果映像と6つの効果音について印象評価を行った。川崎と井手口[77]では、動画映像から受ける印象を14種類の項目から評価した。これを参考にし、本研究では、迫力、軽快さ、温冷、鬱陶しさ、美しさ、躍動感という7種類を印象の項目とした(表5.2)。印象評価はSD法により実施し、各項目は7段階リッカート尺度の選択肢とした。

各演出の印象評価結果を図5.7~5.8に示す。印象のばらつきを箱ひげ図で示した。映像に比べ音響演出の方が全体的にばらつきが大きく、印象の個人差が大きい。鬱陶しさ、美しさの2項目は、優劣がある印象項目であるが、いずれの項目も映像演出の方が評価が高かった。

各演出の印象の比較を図5.9~5.10に示す。映像演出に対する印象が比較的均等になってい

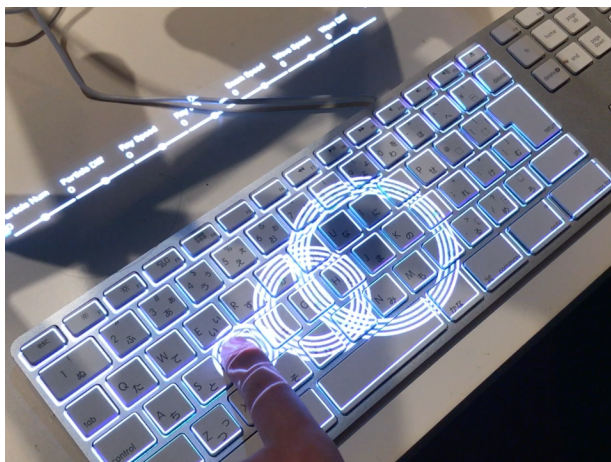


図 5.6. 効果映像 (wave) を投影した様子

表 5.2. 印象評価の項目.

印象項目	評価尺度 (1~7)
迫力	迫力がない~迫力がある
軽快さ	重々しい ~ 軽快
温冷	冷たい ~ 温かい
鬱陶しさ	うっとうしい ~ うっとうしくない
美しさ	美しくない ~ 美しい
躍動感	躍動感がない ~ 躍動感がある
派手さ	地味 ~ 派手

ることに対し、音響演出に対する印象は、効果音によって異なることが見て取れる。

5.5.3 効果映像の最適パラメータの調査

映像演出において、効果映像が与える影響の法則性を明らかにするため、心地よいと感じる効果映像のパラメータ調査実験を行った。

被験者は、効果映像のパラメータを変更し、もっとも心地よいと感じるよう調整した。パラメータ変更は、投影されたコントロールパネルのスライダーをマウスでドラッグすることで実施できる。最適化が終了したら、その数値をアンケート用紙に記入してもらった。

最適パラメータの調査は、particle, ray, character, wave の4つの映像効果を対象とした(表 5.3)。調整するパラメータは、粒子数、移動速度、拡散範囲のいずれかであり、映像効果によって異なる。

押下したキーを中心にして広がる映像効果は、キーからの位置がある閾値を超えると消滅する。この閾値のことを拡散範囲と呼び、単位は pixels。速度は、効果映像の一つのオブジェクトが単位時間あたりに移動する大きさと、単位は pixels/frame。なお、キーボード全体がおおよそ解像度 1280 × 640 に相当し、効果映像の再生フレームレートは 30fps である。

particle では飛び散る粒子の数と拡散範囲、ray では光線が飛び散る速度と拡散範囲、

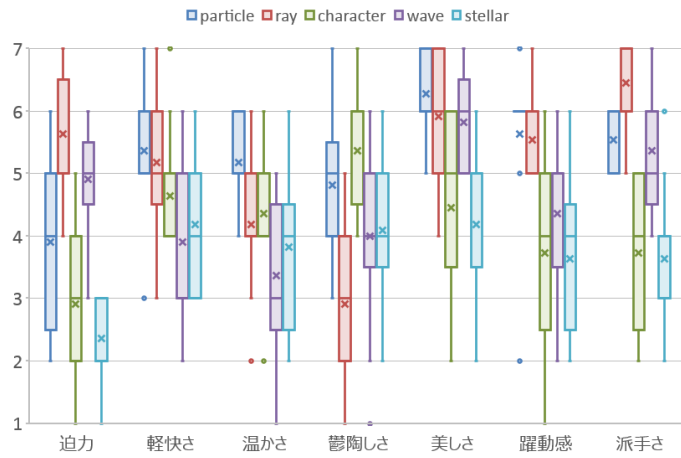


図 5.7. 効果映像の印象評価の結果. 直線の上下端は外れ値を除いた最大値、最小値、矩形の上下端は第三四分位点、第一四分位点. 矩形内の罫線は中央値, ×点は平均値. 矩形外の点は外れ値を示す.

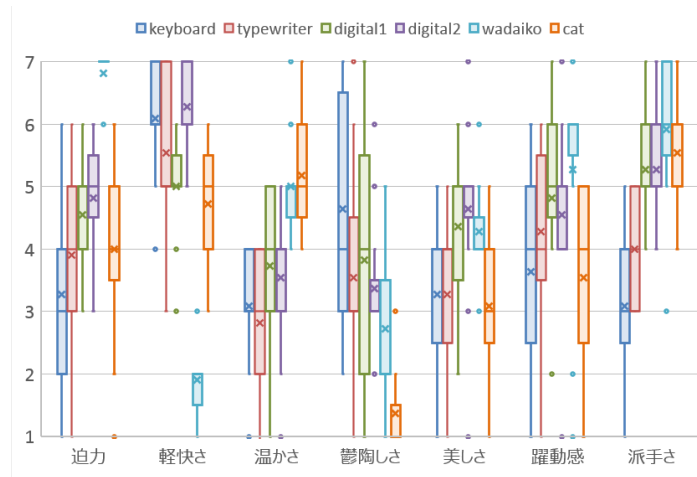


図 5.8. 効果音の印象評価の結果. 読み方は図 5.7 と同じである.

character では文字が飛び出す速度, wave では波紋が広がる速度と拡散範囲を可変パラメータとした.

表 5.3. 最適パラメータの調査で用いた効果映像とそのパラメータ

効果映像	パラメータ 1	パラメータ 2
particle	粒子数	拡散範囲
ray	速度	拡散範囲
character	速度	-
wave	速度	拡散範囲

各効果映像において, 被験者が最も心地よいと感じたパラメータの値をプロットした結果を図 5.2 に示す.

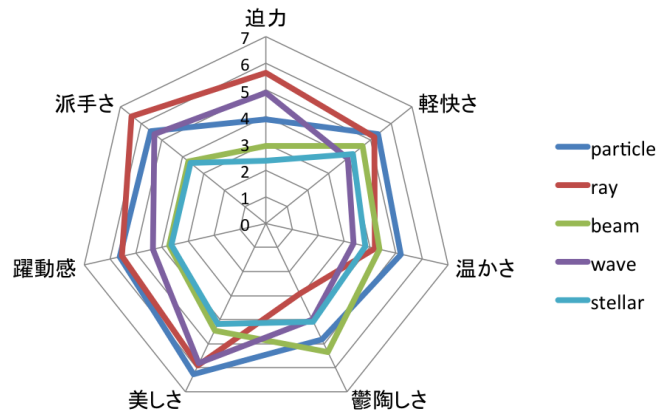


図 5.9. 効果映像の印象の比較.

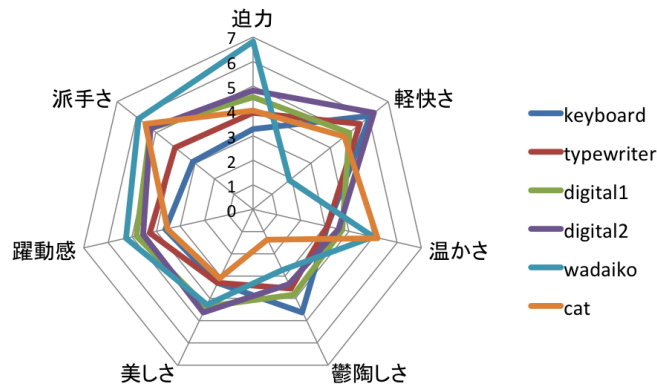


図 5.10. 効果音の印象の比較.

particle(図 5.11) では, 拡散が 100 前後のグループ (これをグループ A と呼ぶ) と, 拡散が 110 以上の右肩上がりのグループ (これをグループ B と呼ぶ) の 2 つに分かれた. グループ A は, 粒子数に依存せず拡散の範囲がほぼ一定なことから, 拡散を重要視している層であることを示唆する. 一方, グループ B は, 拡散が粒子数に比例することから, 粒子の分布密度が一定になることを好む層であることを示唆する. ray(図 5.12) では, 特徴的な分布にはなっておらず, 特性を見出すことはできなかった. character(図 5.13) は, パラメータが文字射出速度の 1 つのみのため, 1 次元プロットで示した. 分布の偏りが少なく, 人によって傾向が様々であることが示唆される. wave(図 5.14) では, ややばらつきがあるものの, 相関係数 $R^2 = 0.31252$ の線形近似で表すことが出来る. 拡散範囲が速度に比例することから, 波紋の生成から消滅までの時間が印象に影響を与える要因となっていることを示唆する.

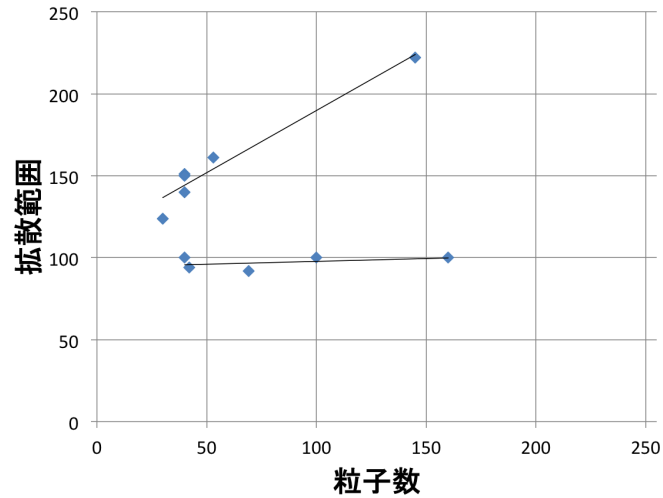


図 5.11. particle における粒子数と拡散範囲の最適値のプロット. グループ A(拡散範囲が 100 前後のグループ) とグループ B(拡散範囲が 110 以上の右肩上がりのグループ) に分けて線形近似を行った. グループ A の線形近似式は $y = 0.0349x + 94.332$ (相関係数 $R^2 = 0.19899$), グループ B の線形近似式は $y = 0.7593x + 113.96$ (相関係数 $R^2 = 0.94539$) である.

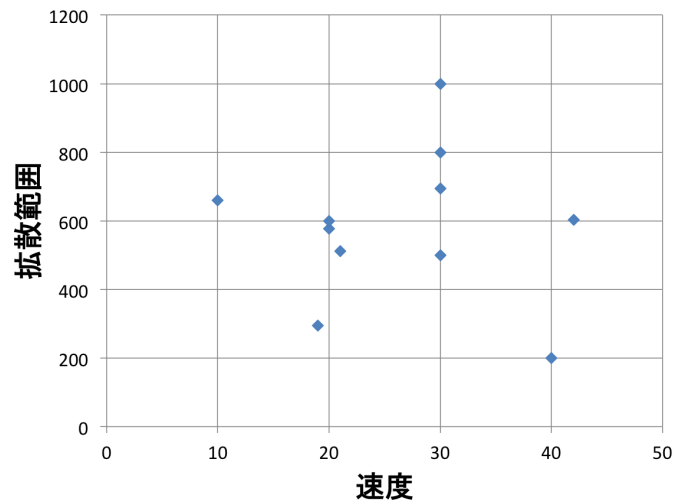


図 5.12. ray における速度と拡散範囲の最適値のプロット.

5.5.4 映像演出への感想

アンケートに記述された映像演出への感想を引用しながら, 映像演出への印象や課題をまとめる. () 内は, どの効果映像に対して記述された感想かを示す.

効果映像が前方向への移動である character では

- 他のエフェクトに比べ、タイピングを阻害することなくエフェクトを楽しむことができ



図 5.13. character における速度の最適値のプロット.

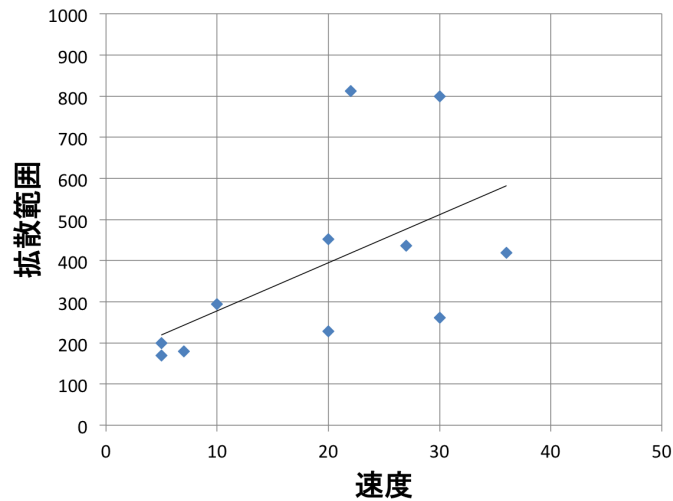


図 5.14. wave における速度と拡散範囲の最適値のプロット. 線形近似式は $y = 11.678xx + 161.39$ (相関係数 $R^2 = 0.31252$) である.

て良かった。(character)

という感想があり, 効果映像の表示する位置や効果の進行方向によって障害性が変わることを示唆している. 効果映像の移動がない stellar では

- もっと周りに広がっていくといいなと感じました (stellar)
- もう少し指からはなれてキラキラだったらよかったかも。(stellar)

という感想が挙げられた. 効果の位置と移動の有無は, タイピング体験にも影響を与えると考えられる.

障害性は, Augmented Typing で想定していた懸念点のひとつであり, 実際に

- 範囲を広げすぎると、他のキーが見つらなくなる (wave)

という感想がいくつか挙げられた。

タイピングはキーボードの上に手を載せて行うため、

- 指にかくれてエフェクトが見えにくい 押した段階で上にキラキラがきて欲しい (stellar)

という、キーボード上に表示された演出が見えないという問題点も指摘された。

キーボードのタイピングは、熟練すると、手元をあまり見ずに打つようになる。そのため、付与した映像演出が目に入らないという懸念がある。実験でもその点に触れ

- 画面見ながら打つとどう感じるのか気になりました (all)
- キーボードを見ないで視界の端に少しうつるくらいだと楽しい (wave)

といった感想が挙げられた。

その他に

- 高速、広範囲だとタイピングをはやくするととても楽しかった (ray)
- 軽快なのがすごくよかったです。(ray)

といった好感触な感想が挙げられた。

5.5.5 音響演出への感想

アンケートに記述された音響演出への感想を引用しながら、音響演出への印象や課題をまとめる。()内は、どの効果音に対して記述された感想かを示す。

音響演出に対する好印象の感想として

- よくきく音なので、これこそキーボードを叩く音だなって感じでした。(keyboard)
- iPhoneなどの仮想のキーボードとかにこういう音がつくと、打っている感じが出るので良い (keyboard)
- 作業がはかどるような感覚が得られそう。(keyboard)
- 見た目とすごくマッチしててよいです (digital1)
- 未来のPCを使っている感じがして楽しい。映画のワンシーンに入っている気分。(digital1)
- 未来感のある音とキーボードの見た目がマッチしていて、かなり気持ちいいフィードバックに感じました (digital2)

というものが挙げられた。これらの感想は、Augmented Typingがタイピング体験を実際に拡張していることがわかる。一方で、拡張によって違和感が発生した場合もあり、

- タイプライターの音がこのキーボードを打っているときに聴こえることに多少違和感が

54 第5章 物理空間への重畳演出による人間の感情への干渉: Augmented Typing

ありました。(typewriter)

- このキーボードの見た目、UI からこの音が出る気がしないので違和感がありました。(digital1)

という感想も挙げられた。

映像演出では、タイピング時に手元を見ないことが懸念点であったが、音響演出ではその問題は解決され、

- ブラインドタッチでも楽しいので個人的には好き (all)

という感想が挙げられた。

映像演出で発生したタイピングの阻害という問題点は音響演出でも同様にあり、

- たたくこと自体はたのしいが書くことを考えてタイピングするには不向きな気がした (wadaiko)
- やかましい (cat)

といった感想が挙げられた。特に wadaiko と cat でタイピングを阻害するという感想が多く挙げられ、鬱陶しさの印象評価でもその傾向は見られた。

5.6 デモンストレーション

Augmented Typing を体験した際の反応を観察して意見や感想、課題を収集するため、イベントや学会など様々な場所で計 15 回のデモンストレーション展示を行った (表 5.4)。特別な背景知識や文脈を必要とせず、身近な道具であるキーボードを用いた展示であることからか、老若男女問わず好印象の反応であった。特に子どもはタイピングに熱中し、長時間体験することもあった。日本国内にとどまらず、海外でも展示を行った (図 5.15)。

さらに、効果映像の character に相当する、文字を打ち出す演出を撮影し、その動画を Twitter に投稿したところ、35,000 以上リツイートされ、賞賛や驚嘆の反応が見られた。その

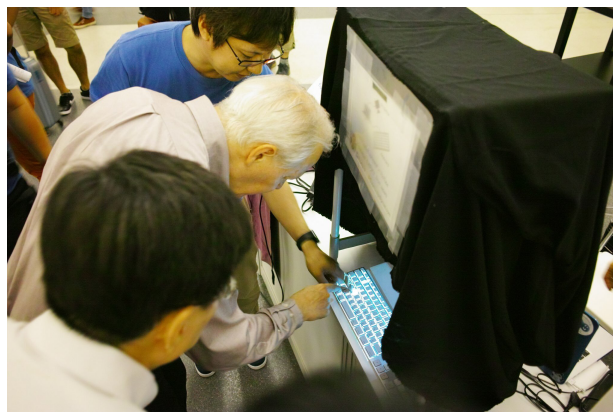


図 5.15. Maker Faire Singapore2016 での展示の様子

時の反応をツイートまとめサイト Together にまとめた [78].

5.7 議論

5.7.1 実装方法について

本研究ではプロジェクションマッピングを利用してシステムのプロトタイプ実装を行った。そのため、キーボードに投影するはずの効果映像が指に投影されてしまう。評価実験ではこの問題は特に指摘されなかったが、被験者が普段からプロジェクションマッピングの技術に慣れ親しみ特徴を把握しているため違和感を持たなかった可能性もある。また、現在のシステムではプロジェクターとそれを支える三脚が大きいため、日常で利用するには邪魔である。また、キーボード自体の価格に比べて非常に高価である。プロジェクターの小型化、短焦点化と低価格化が進めば、普及しやすくなるであろう。

キーボードのキートップにディスプレイを搭載して映像演出を行う場合、指に投影される問題は解決される。しかし、現在 Optimus[59] のように市販されているものもいくつかあるが、キーボードとしては高価なこともあり、普及しているとはいえない状況である。また、映像演出を付与できる領域がキートップに限られるという制約もある。

キートップのみではなく、キーボード全体をタッチパネル式ディスプレイにする方法も考えられる。スマートフォンやタブレット端末を使用する際など、タッチパネル式ディスプレイ上のソフトウェアキーボードでタイピングをする機会も増えてきた。また、それらの普及により製造コストも下がってきていることから、いまのキーボードに置き換わるキー入力専用デバイ

表 5.4. デモンストレーション展示を実施したイベント一覧。

イベント名	場所	展示日時
TMCN Vol.9	日本マイクロソフト本社	2014/10/01
GUGEN2014	富士ソフトアキバプラザ	2014/12/13-14
WIDE 合宿 2016 春	信州松代ロイヤルホテル	2016/03/10
Maker Faire Singapore 2016	Singapore University of Technology and Design	2016/06/25-26
NT 金沢 2016	金沢駅もてなしドーム地下広場	2016/07/16-19
Maker Faire Tokyo 2016	東京ビッグサイト	2016/08/07
秋葉原メイカーズ倶楽部	mogura	2016/08/07
NICOGRAPH 2016	富山大学	2016/11/04
NT 金沢 2017	金沢駅もてなしドーム地下広場	2017/07/08-09
AkiParty Tokyo 2017	mogura	2017/08/05
NT 金沢 2018	金沢駅もてなしドーム地下広場	2018/07/07-08
AkiParty Tokyo 2018	mogura	2018/08/04
Maker Faire Taipei 2018	華山 1914 創意文化園區	2018/11/03-04
NT 加賀 2018	アビオシティ加賀	2018/12/15-16
Asian CHI Symposium 2019	Scottish Event Campus	2019/05/05

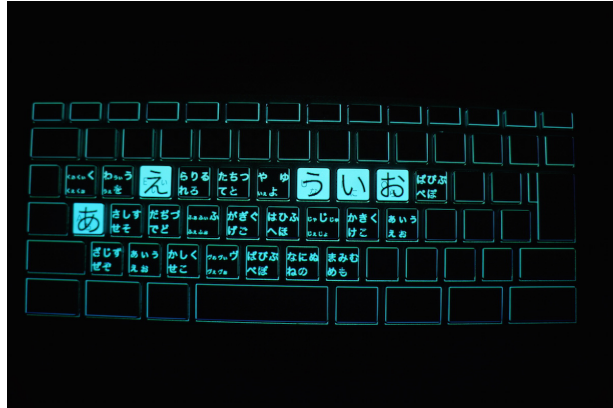


図 5.16. 日本語入力支援システム

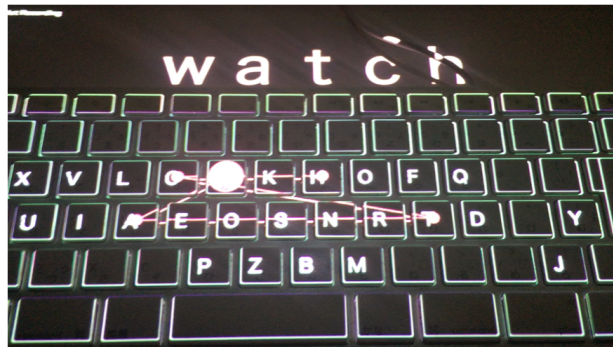


図 5.17. トポロジ表示によるタイピング学習支援システム

スとしてタッチパネル式ディスプレイが普及する可能性もある。タッチパネル式ディスプレイによるタイピングの欠点は、押下感がなく、タイピングのリズムが狂ったり、タッチタイピングがしにくいことである。今後、押下感を擬似的に発生させる 3D Touch[79] などの技術の発展により、タッチパネルで押下感のあるタイピングができるようになるかもしれない。また、押下感の欠如を補う技術として Augmented Typing を活用することも考えられる。

5.7.2 タイピング学習支援への応用

PC 初心者がタイピングを練習する方法として、タイピングソフトの使用が一般的である。しかし、タイピングソフトは PC の画面を見ながら入力を行うものであり、タッチタイピングを習得する人には向いているが、初心者がタイピングを修得するためには必ずしも最善の方法ではない。Augmented Typing を活用すれば、初心者に適した新たなタイピング学習方法を提供できるかもしれない。

まず、Augmented Typing によりタイピング自体が楽しい体験となれば、初心者がキーボードに触れるきっかけとなる。さらに、Augmented Typing を活用したタイピング学習支援の例を 2 つ挙げる。ひとつは、日本語ローマ字入力において、キートップにそのキーを押下すると入力される文字を表示する支援システムである (図 5.16)。タイピング初心者の障壁となる

日本語入力の学習を促進する。もう一つは、視覚情報によってタイピング学習を促進する支援システムである (図 5.17)。単語に含まれる文字のキー同士を線でつなぎ、単語をトポロジ (形状) で視覚的に把握することでキー配列の記憶定着を補助する。

なお、タイピング学習支援に関するシステムは、過去にもさまざまな研究で提案されている。辻本ら [80] は、拡張現実感を用いて、タイピング学習を支援するツールを開発した。田村ら [81] は、タイピング中の運指を認識しシステムを開発し、適切な運指でのタイピングの促進に対する効果を測定した。Imamura et al.[82] は、タッチタイピングの練習の際に手元を見ていないかどうかを視線の検知により判定した。

5.8 おわりに

本研究は、Augmented Typing という映像と音の演出付与によるキーボードタイピング体験の拡張を提案し、そのシステムのプロトタイプ実装を行った。5つの効果映像と6つの効果音について7種類の軸で印象評価を行った。映像に比べ音響演出の方が、印象評価のばらつきが大きく、美しさは映像演出の方が高く、音響演出の方が鬱陶しいという傾向が現れた。効果映像のパラメータを調整して最適値を探る調査では、particle では特定の拡散範囲によって心地よさが決まる群と粒子の分布密度によって心地よさが決まる群がいること、wave では波紋の生成から消滅までの時間が印象に影響を与えることを示唆する結果を得たが、より信頼性の高い法則とするにはさらなる調査が必要である。本研究の目的であった、演出付与がタイピング体験を拡張することがわかる感想が多数挙げられた。また、効果演出の表示の方向や移動の有無が印象に影響を与えることを示唆する感想も得られた。懸念していた通り、演出付与がタイピングの阻害になる場合もあり、これは今後の課題として扱う。今後、印象項目や演出効果を増やして実験することにより、演出が与える印象や演出効果との関係性がより明らかになるであろう。

第II部

Cyber-Physical-Human Interaction の システム開発

第 6 章

無線通信の仮想化: BluMoon

6.1 はじめに

6.1.1 CPHI としての位置づけ

物理空間を考慮して CPHI のシステムテストを行うには、物理量の伝達の際に発生する問題 (図 6.1) を考慮する必要がある。本節では、物理空間の状態を情報空間で模倣し、物理量まで含めたソフトウェアテストを行うことを提案する。情報通信のprotocolsの一つである Bluetooth Low Energy のエミュレータの設計と実装を行い、その中で「受信強度」「電波干渉」という物理空間における問題を模倣する。

6.1.2 背景

ネットワークに繋がる機器は PC や携帯電話にとどまらず、家電機器やウェアラブルデバイスなどのさまざまな機器がその周辺機器やサーバとのデータの送受信をするようになった。機器のネットワーク接続において、Bluetooth Low Energy (BLE)[83] はよく使われる無線通信プロトコルのひとつである。BLE は、広く普及する近距離無線通信規格であり、低消費電力で通信可能であるため組み込み機器への導入に適する。また、Global Positioning System (GPS) が利用できない屋内での測位を目的としたビーコンとしても用いられる。

BLE を用いたシステムの開発は、無線通信を扱うことによる特有の困難さを伴う。BLE は



図 6.1. CPHI の物理における問題.

電波を用いた無線通信であるため、送受信機の位置や周囲の環境に応じてふるまいが変わる。例えば、近接判定などに用いる受信電波強度指標 (Received Signal Strength Indicator:RSSI) は、送受信機の距離に応じて変化することに加え、壁や地面などの周囲の物体による反射、遮蔽、回折による影響も受ける。この周囲の状況の微細な変化によって、RSSI は変動を起こす。そのため、アプリケーションに RSSI を用いる場合には、この変動を考慮したうえで設計する必要がある。また、システムの対象機以外の BLE 送信機やその他の電波を発する機器の影響を受けて電波干渉が発生し、通信が不通となることもある。

一般的なソフトウェア開発では、ソフトウェアが想定通りの動作を示すことを都度確認し、不具合があれば修正しながら開発を進める。しかし、BLE を用いたシステムで動作確認を行うには、無線電波を送受信する実機が必要となる。これには実機を用意する必要があり、加えて設置作業にも手間がかかる。送受信機の位置を変更するようなテストを行う場合、人や機械による機器の移動も必要となる。

これらの課題は、無線通信部分のテストに仮想的な環境を用いることで解決できる。機器の移動や周囲の環境を簡易に設定できるようなテストを無線送受信機を用いずにすべてコンピュータ上で実行できれば、テスト機材費用とその設置作業の工数を大きく削減できる。仮想的な環境を用いた無線通信システムのテスト手段のひとつに、シミュレーションを用いる方法が挙げられる。通信の模倣には、NS-3[84] のようなネットワークシミュレータが用いられる。ネットワークシミュレータは、ネットワークのトラフィックを模倣してスループットや遅延などのネットワーク特性を調べるのが主目的である。実環境向けのアプリケーションをそのまま稼働し、アプリケーションが生成するデータをフレーム単位で送受信してシステム全体の動作を確かめるようなテストに用いることはできない。もうひとつの手段は、無線エミュレーションを用いる方法である。IEEE802.11 のエミュレーションシステム NETorium[85] は、仮想ネットワークデバイスを用いて無線通信を有線ネットワーク上で模倣する。カプセル化した無線フレームの送受信をイーサネットを通じて行う。遅延やパケットロスなどの無線の特性は、パラメータとして注入する。電波干渉は、フレームごとに仮想ネットワークデバイスが衝突判定を行うことで模倣する。また、OS からは IEEE802.11 のデバイスと同様に扱うことができるため、実環境向けのアプリケーションをそのまま可動することができる。

NETorium のような無線エミュレーションを BLE を用いたシステムの検証に用いれば、先述の課題を解決できる。しかしながら、NETorium と同様の機能を実現するための BLE エミュレータは存在しない。さらに、他のプロトコルの無線エミュレータを改変して BLE エミュレータを開発することも困難である。BLE では、コントローラと呼ばれるフレーム送受信機が状態を保持し、他のデバイスとの接続確立の段階に応じて状態遷移する。状態に応じて、自律的にフレーム送信を行う場合もある。このようなふるまいをする BLE のコントローラの特徴は、IEEE802.11 や他のプロトコルの仮想ネットワークデバイスと異なるからである。そこで本研究では、BLE を用いたソフトウェアシステムのテストのための BLE エミュレータの設計と実装を行う。

ここで、ソフトウェアシステムテストのための BLE エミュレータの要件を整理する。まず、テスト対象のソフトウェアがそのまま動くことである。テスト用のソフトウェアと実環境用の

ソフトウェアを書き分けることは開発工数が増大し、違うソフトウェアを作成することで異なる挙動を示す可能性もあるため、同一のソフトウェアが稼働することが求められる。そして、BLE アプリケーションでは、近接判定などの用途に RSSI を用いる。RSSI はフレームごとに大きく変わるため、受信フレームごとにその状況に応じた RSSI が必要となる。そのため、BLE の通信をフレーム単位で行うことと、フレームごとに受信電波強度を計算することが求められる。最後に挙げるのは、電波干渉の模倣である。多数の機器が通信を行うシステムのテストでは機器の配置や設置台数に応じた接続可否の確認が重要となるため、テスト段階において電波干渉の確認が必要なためである。

これらを踏まえ、我々は BLE エミュレータ BluMoon を提案する。上記の要件を満たすため、

1. 実機向けソフトウェアが動作すること
2. フレーム単位で通信を行うこと
3. 受信電波強度を計算すること
4. 電波干渉を模倣できること

という条件のもとに BluMoon の設計を行う。BluMoon の開発が実現できれば、BLE を用いたソフトウェアシステムのテストのための環境をコンピュータ上に構築でき、開発の大きな手助けとなる。

我々は、BluMoon の原案について研究報告 [86] とワークショップ論文 [87] にて報告した。本論文は、既報の内容を整理してまとめたことに加え、電波干渉と受信電波強度を模倣するための設計と実装を追加として記述する。さらに、性能評価と実機との比較実験を行った結果についても追加として記述する。

6.2 Bluetooth Low Energy

Bluetooth は、2.4GHz 帯を使った近距離無線通信の規格である。BLE は Bluetooth Version 4.0 で採用された低消費電力のプロトコルである。BLE を含む Bluetooth の仕様は、Bluetooth SIG によって定められる。本章では、Bluetooth の階層構造と接続確立に焦点をおき、BLE エミュレーションの設計に関する内容について記述する。

BLE を含む Bluetooth は、アプリケーション、ホスト、コントローラという階層構造をとる (図 6.2)。Bluetooth を用いるアプリケーションは最上位層に位置し、多くの場合でホストの機能を利用する。電波を送受信して通信を行う機能はコントローラが担う。ホストとコントローラの間は、ホストコントローラインタフェース (Host-Controller Interface:HCI) を通じて命令や通知、データのやり取りが行われる。

ホストでは、用途に合わせてアプリケーション開発が容易となるよう、Generic Access Profile (GAP), Attribute Protocol (ATT), Generic Attribute Profile (GATT), Security Manager Protocol (SMP), Logical Link Control and Adaptation Protocol (L2CAP) 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。

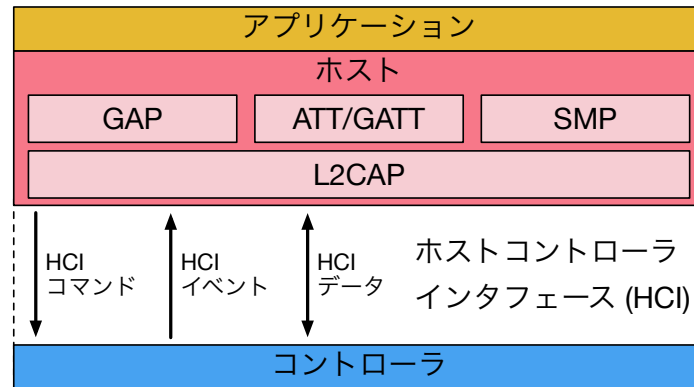


図 6.2. Bluetooth の階層構造.

コントローラは、電波を送受信して周囲のデバイスとの通信を行う。コントローラにはいくつか種類があり、BLE では、Low Energy (LE) というコントローラを用いる。他には、Basic Rate/Enhanced Data Rate (BR/EDR) というコントローラがあり、これは一般的には Classic Bluetooth とも呼ばれる。BR/EDR と LE の2つの規格が Bluetooth の主要な規格であり、市販の多くのコントローラは BR/EDR と LE の両方に対応したデュアルスタックで実装されている。以後、本論文で BLE コントローラまたは単にコントローラと呼ぶ場合、LE コントローラのことを指す。

HCI には、ホストからコントローラへ命令を送信する HCI コマンド、コントローラからホストへ応答や通知を送信する HCI イベント、データを双方向に送受信する HCI データの3種類の通信方式がある。これらは、HCI フレームという形式で送受信される。

6.2.1 階層構造

6.2.2 BLE コントローラの状態遷移

BLE では、機器はセントラルとペリフェラルという2つの役割に分かれる。セントラルとペリフェラルが接続を確立する順序を図 6.3 に示す。そして、BLE コントローラは接続の準備状況に応じて、Standby, Advertising, Scanning, Initiating, Connection という5つの状態 (図 6.4) のいずれかをとる。セントラルでは、接続確立までに Standby → Scanning → Initiating → Connection という状態遷移をたどる。ペリフェラルでは、接続確立までに Standby → Advertising → Connection という状態遷移をたどる。Standby は、コントローラを ON にした後の初期の状態である。Advertising は、アドバタイズパケットを定期的を送信する状態である。Advertising へは、LE Set Advertising Enable という HCI コマンドによって遷移する。Scanning は、周囲のアドバタイズパケットを待ち受けている状態である。アドバタイズパケットを受信した際には、ホストに LE Advertising Report という HCI イベントを送信する。Scanning へは、LE Set Scan Enable という HCI コマンドによって遷移する。Initiating は、セントラルが接続確立する準備ができ、ペリフェラルからのアドバタイズ

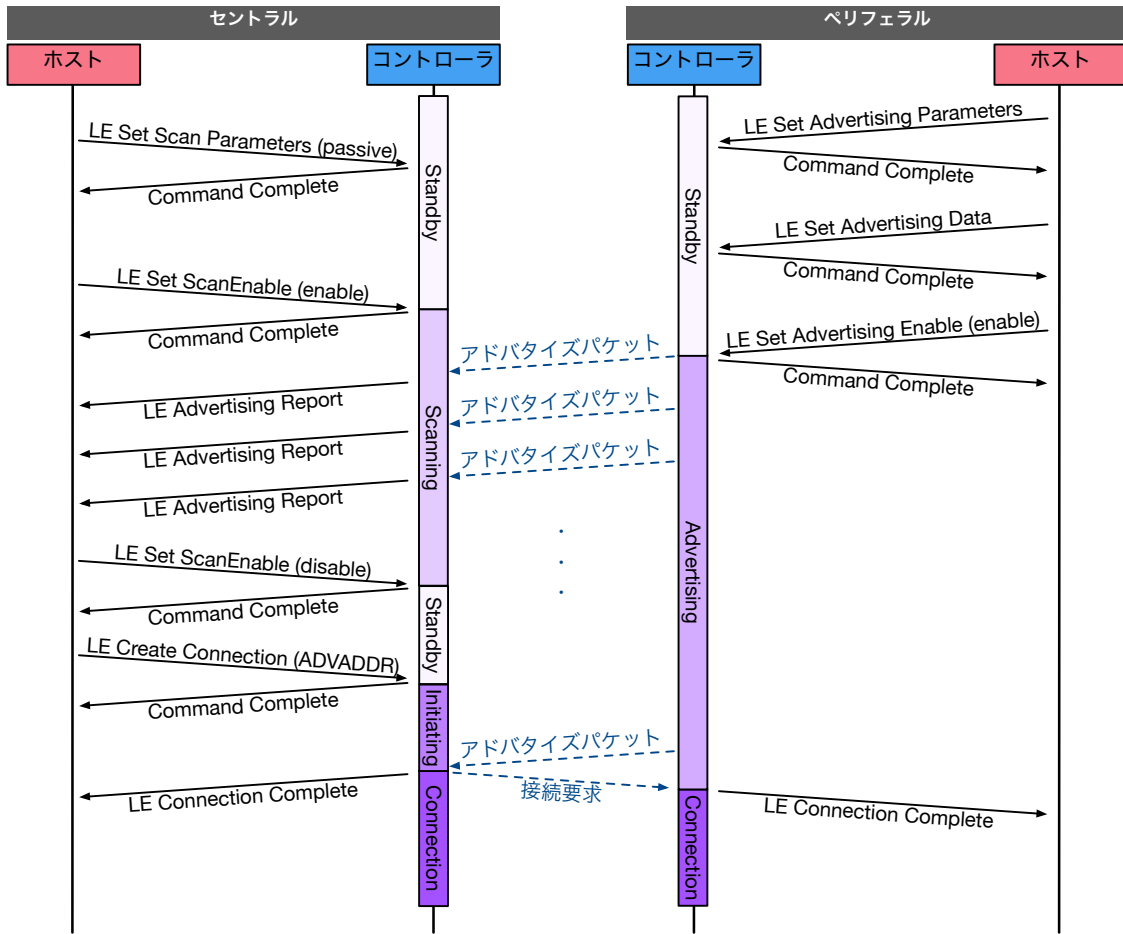


図 6.3. 接続確立までの HCI コマンドおよび HCI イベント, BLE フレームのシーケンス図.

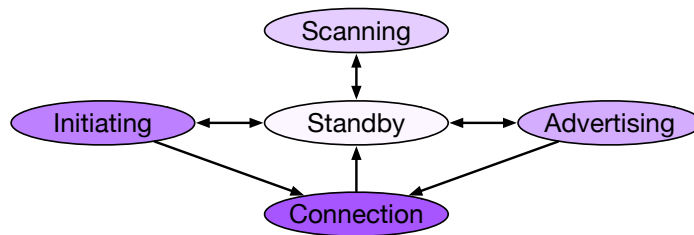


図 6.4. BLE コントローラの状態遷移図.

パケットを待ち受けている状態である。Initiating へは、LE Create Connection という HCI イベントによって遷移する。Connection は、セントラルとペリフェラルの間で接続が確立されている状態である。セントラルでは、Initiating の際にアドバタイズパケットを受信すると Connection に遷移する。このときセントラルは接続要求を送信する。ペリフェラルはこの接続要求を受信すると Connection に遷移する。Connection では、データを互いに送受信できる。Connection が切断されると、コントローラは Standby に戻る。

アドバタイズパケットや接続要求、データ送受信などのコントローラ間でやりとりされる



図 6.5. BLE フレームのデータ形式.

表 6.1. 本論文の BLE エミュレータの要件に対する既存のエミュレーションとシミュレーション手法の適合状況. 要件は (1) 実機向けソフトウェアが動作すること, (2) フレーム単位で通信を行うこと, (3) 受信電波強度を計算すること, (4) 電波干渉を模倣できること.

	(1)	(2)	(3)	(4)
ネットワークシミュレータ [84]	×	×	○	○
btvirt[90]	○	×	×	×
Android エミュレータ [91]	○	×	×	×
Wear OS エミュレータ [91]	○	△ ³	×	×
iOS シミュレータ ¹ [92]	△ ²	×	×	×
ペリフェラルシミュレータ [93][94]	○	×	×	×
BluMoon(提案手法)	○	○	○	○

¹ iOS バージョン 5 に対応するシミュレータに限定

² 実機向けと同一のソースコードをシミュレータ向けにビルド

³ Android API レベルでペアリングを模倣

データフレームを, 本論文では BLE フレームと呼ぶ. BLE フレームは図 6.5 に示す形式で構成される. なお, アドバタイズパケットは, 接続を確立せずに情報を送信する手段としてそれ単体でも活用される. アドバタイズパケットを測位と情報送信に活用することを目的とした, iBeacon[88] や Eddystone[89] などの規格もある.

6.3 関連研究

6.3.1 Bluetooth のシミュレータとエミュレータ

本節では, BLE に関するネットワークやアプリケーションのテストのために用いられる既存のシミュレータやエミュレータを挙げる. これらのエミュレーションやシミュレーションの, 1 章で挙げた要件 (1)~(4) への適合状況を表 6.1 にまとめた. また, これらが, BLE のアプリケーション, ホスト, コントローラ, 通信の各要素を模倣あるいは実機・実ソフトウェアのどちらで実行するのかを図 6.6 にまとめた. 本論文で提案する BluMoon では, 要件 (1)~(4) をすべて満たすよう, コントローラと通信をエミュレートする設計を行う.

BLE を始めとする通信の模倣には, ネットワークシミュレータが用いられる. 代表的なも

のには NS-3[84] がある。ネットワークシミュレータは、スループットや遅延を調べるためにネットワークのトラフィックを模倣するもので、システムテストという目的には合致しない。

btvirt は、Linux の Bluetooth スタック BlueZ [90] 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 の仮想コントローラとして振る舞い、UNIX ドメインソケット経由で HCI フレームの送受信を行い、HCI コマンドを受け付け、適切な HCI イベントを返す。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、btvirt 自体には外部と通信をする機能を有さないため、通信まで含めたシステムテストに用いることはできない。

モバイルアプリケーション開発では、PC 上の開発環境に付属したエミュレータやシミュレータを使うのが一般的となっている。Android では、開発環境 Android Studio[91] に付属するエミュレータや、その他にサードパーティ製のエミュレータが多数存在する。これらには Bluetooth を模倣する機能を持たず、Bluetooth の動作確認を基本的には実機で行う。ただし、腕時計型デバイス向けの Android ベースの OS である Wear OS[95] のエミュレータは、BLE のペアリングを模倣することができる。Wear OS 搭載デバイスは BLE のペリフェラルとなり、BLE のセントラルのスマートフォンとペアリングして使用する。そのため、Wear OS エミュレータには、セントラルの Android スマートフォン実機または Android エミュレータとペアリングする機能を持つ。実機の Android スマートフォンとペアリングする場合は、Wear OS エミュレータが稼働する PC と USB で接続し、実機側でポートフォワーディング設定を行う。iOS では開発環境 Xcode[96] に iOS シミュレータが付属し、実機向けと同一のソースコードをシミュレータ用にビルドしたバイナリを、MacOS 上のシミュレータで実行することができる。iOS シミュレータにも、BLE を模倣する機能が搭載されていた。Bluetooth を使用するアプリケーションを iOS シミュレータでテストする際に、シミュレータが稼働する Mac 本体の Bluetooth コントローラを使用することができる。周辺機器の実機を用意することで、iOS シミュレータでも BLE 通信を用いて動作確認することができる。ただし、この方法は iOS バージョン 5 に対応するシミュレータで終了しており、それ以降のバージョンでは実施できない [92]。これらは、特定のデバイスに特化したテスト環境を提供するものであり、BLE を用いたシステムのテストに汎用的に用いることはできない。

また、BLE のセントラルアプリケーションテストのために、ペリフェラル機器の動作を模擬するスマートフォンアプリケーションもある。本論文では、これらをペリフェラルシミュレータと呼ぶ、BLE Peripheral Simulator[93] 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する。同様に、LightBlue Explorer[94] も、様々なペリフェラルデバイスを仮想ペリフェラルとして起動することが出来る。これらは、スマートフォンの実機を用いて、仮想デバイスに成り代わるものである。実機を用いて物理的な電波を送受信するものであり、通信のエミュレーションを行うものではないため、我々が目的とするエミュレーションを用いたシステムテストに用いることはできない。

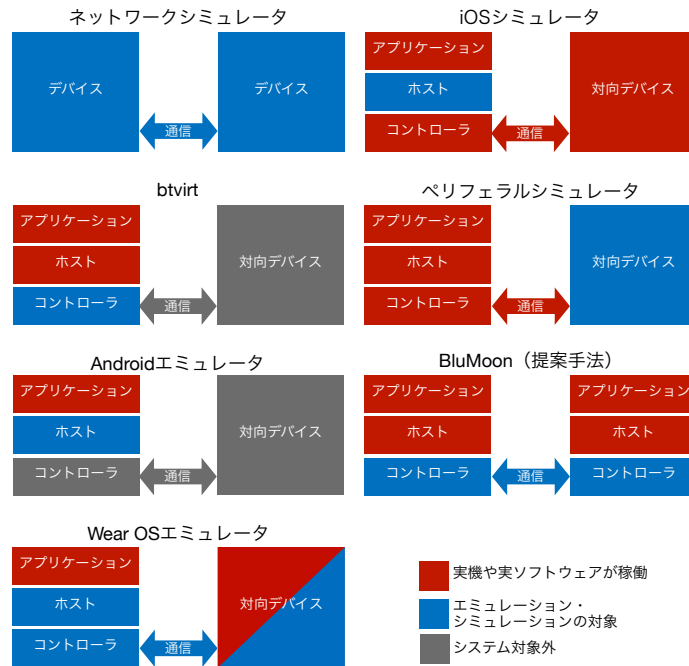


図 6.6. 既存のエミュレーション・シミュレーション手法の模倣ターゲットの比較.

6.3.2 Bluetooth のネットワーク仮想化手法

BluMoon では、フレーム単位での BLE 通信エミュレーションを行う。そのために、他のプロトコルの上に BLE フレームを載せて通信する仮想化を検討する。Bluetooth 通信の仮想化は、これまでも様々な手法が研究されてきた。

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 で規定されたデータをカプセル化して送受信する方法がいくつか提案されている。UbiPAN[97] では、互いの近くに設置されたゲートウェイがブリッジし、遠隔地にある機器同士と Bluetooth で通信を行うことができる。Tsuda *et al.*[98] 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、岡田と鈴木 [99] によって BLE に対応するよう拡張された。これらの手法は、カプセル化して転送された後に再び Bluetooth として無線フレームを送信する。そのため、RSSI などの無線特有の情報は含まれておらず、有線ネットワークでのエミュレーションに用いるには不十分である。

Bluetooth のエミュレーションをテストに用いる試みとして、BlueMonarch[100] がある。BlueMonarch は、Bluetooth の Object EXchange (OBEX) でのファイル送信を、ペアリングが不要な Service Discovery Protocol (SDP) 通信を用いて模擬するエミュレータである。コンテンツ配信システムなどの評価を行う。これは、Bluetooth 以外の通信プロトコルを利用するものではなく、Bluetooth の別プロトコルを利用して模擬を行うものである。よって、本論文で対象とする有線ネットワークでのエミュレーションとは異なる。

6.3.3 Bluetooth 以外の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[101] は、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[102] や、IEEE802.15.4 エミュレータ [103]、動的ネットワークエミュレーションツール DynamiQ[104] も提案される。QOMET と同じく IEEE 802.11 を対象としたエミュレータに Meteor[105] がある。大規模ソフトウェア無線ネットワークエミュレータ NETorium[85] では、上述の Meteor に加え、Asteroid という機構によりフレームの衝突を模倣する。仮想インタフェースとして mac80211_hwsim[106] を用いて通信を行う。

一方、ハードウェアによる無線エミュレータには、FPGA を用いた無線エミュレータ [107] や、Network Emulator II[108] のような商用機としての無線エミュレータがある。ハードウェアによる無線エミュレータは、無線特性を高精度で注入できるが、ハードウェアを要するため、多数のデバイスを用いたシステムのテスト環境を構築することが難しい。

6.4 BluMoon の設計方針

本章では、1 章で述べた BLE エミュレータの 4 つの要件を実現するための手法を提示し、設計方針を示す。

6.4.1 HCI を境界としたコントローラエミュレーション

BluMoon には、実機と同様のソフトウェアが動作するという要件がある。6.2 節で示したとおり、Bluetooth ではホストとコントローラは必ず HCI を経て相互接続される。ホストの下層に接続されるコントローラの挙動をエミュレートすれば、実機と同様のホストとアプリケーションを動かしてテストすることが出来る。コントローラの挙動をエミュレートすれば、コントローラやその先のリモートデバイスの挙動を自由に定義したり、実際にコントローラ同士で通信を行ったりすることができる。ホストからは、HCI を介して通常の Bluetooth 通信と同等に扱うことができる。BluMoon では、HCI より下層のコントローラ部分をエミュレートして置き換える。このエミュレートしたコントローラを BluMoon Controller (BM-CTL) と名付ける。

なお、その他の選択肢として、ホストより下層をエミュレートするという方法も存在する。しかし、ホストには多数のプロファイルが含まれているため、ホストの役割を模倣するようなエミュレータを制作する実装工数は大きい。本論文で提案する HCI を境界とする手法では

ホストに実際のソフトウェアをそのまま使うことができるため、実装工数を抑えることができ、バージョンアップへの追従などもより容易となる。

6.4.2 フレーム単位の通信の模倣

フレーム単位でのエミュレーションを実現するために、BM-CTL 同士で BLE フレームに相当するデータの送受信を行う。このデータを BluMoon Frame (BM-FRM) と名付ける。BM-FRM の形式を図 6.7 に示す。BM-FRM には、実際の BLE 形式に基づいた BLE フレームをそのまま格納する。加えて、後述する受信電波強度および電波干渉の模倣に用いる Channel, TxPower, Location の 3 種類のメタデータを付与する。

6.4.3 受信電波強度の模倣

BLE では、コントローラが受信したフレームごとに RSSI を算出し、その情報をホストに渡す。受信電波強度は距離に応じて減衰する特徴を有し、最も単純なモデルである自由空間での電波伝搬損失 $L(d)$ は

$$L(d) = -20 \log_{10} \frac{4\pi d}{\lambda} \quad (6.1)$$

で表すことができる。ここで λ は波長、 d は送信機と受信機の距離である。実環境での受信電波強度は、反射や回折などの様々な要素が影響するため、自由空間電波伝搬損失だけで模倣できるほど単純ではない。しかし、詳細なモデル化による算出は、条件設定の手間がかかったり、計算時間がかかりエミュレーションの実行に影響を及ぼす可能性がある。そこで、BluMoon では、もとの送信電力からこの自由空間電波伝搬損失を考慮し、さらに加算性白色ガウス雑音 (Additive White Gaussian Noise:AWGN) δ を加えて RSSI を算出する。

$$\text{RSSI} = \text{TxPower} - 20 \log_{10} \frac{4\pi d}{\lambda} + \delta \quad (6.2)$$

AWGN δ は、平均 0、標準偏差 σ の正規分布の確率密度関数

$$P(\delta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\delta^2}{2\sigma^2}\right) \quad (6.3)$$

のもとに乱択により生成する。

RSSI を計算するための TxPower は、BM-FRM のヘッダに含まれるものを用いる。距離 d は、送信機と受信機の位置を定めることで求まる。位置情報は、ノード外で集中的に生成し、一定時間毎に各 BM-CTL に向けて送信される。この位置情報生成機構を Location Generator (LC-GEN)、全ノードの位置情報を記述したデータを Location Information (LC-INF) と名付ける。送信機の位置は、各送信機が BM-FRM を送信する際にヘッダに含める。受信機の位置は、各受信機の BM-CTL が持ち、受信時に RSSI を計算する際に用いる。LC-GEN は、時系列に沿った位置情報を指定するシナリオ型や、各ノードがルールに基づいて移動するマルチエージェントシミュレータ型などを想定する。



図 6.7. BluMoon Frame (BM-FRM) のプロトコルフォーマット.

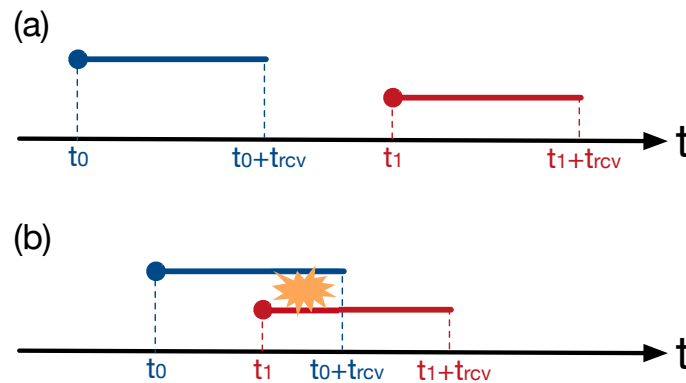


図 6.8. BluMoon における電波干渉の模倣手法の解説図. t_0 と t_1 が BM-FRM の到着時刻, t_{rcv} は BLE フレーム受信に要する時間である. (a) BM-FRM を受信した時刻から t_{rcv} が経過するまでに他の BM-FRM を受信しなければ, 電波干渉なしとみなしてその BM-FRM を受信する (b) 他の BM-FRM を受信した場合には電波干渉が発生したとみなしてその両方の BM-FRM を破棄する.

6.4.4 電波干渉の模倣

BLE は, 2.400GHz-2.480GHz の帯域を 0.002GHz ごとに分割し, 帯域の重ならない 40 チャンネルを構成する. 異なるノードから同じチャンネルにて BLE フレームが送信されると, 信号の復号化が不可能となり通信が成立しない. 本論文では, この現象を電波干渉と呼ぶ. BLE では, BLE 同士や他の 2.4GHz 帯通信との電波干渉を避けるための機構が用意されているが, Advertise では 37,38,39 の 3 つのチャンネルを固定で使用するため特に電波干渉が起こりやすい.

Bluetooth バージョン 4.2 では, フレーム送信のビットレートは 1Mbps と決められている. たとえば, 200[bit] のフレームを送信するには 200[μ s] かかる. 電波干渉は, このフレームの送信中に起こる. 一方, BluMoon ではイーサネットに接続されたネットワークでフレームを送信するため, フレーム送信に要する時間は実際の BLE での時間に比べると極めて短い. そこで, 実際の BLE でかかるであろう送信時間を見積もり, 擬似的な送信状態を生成して干渉を模倣する. この干渉の設計は, 無線ネットワークエミュレータ NETorium に含まれる Asteroid にて提案された.

具体的には, BM-FRM を受信した時刻から一定時間を本来 BLE フレーム受信に要する時間とみなし, その間に別の BM-FRM を受信した際には干渉が発生したと判定する. BLE でフレーム受信に要する時間を t_{rcv} は, フレームのビット長に比例する. BLE フレームのビッ

ト長, すなわち BM-FRM のボディのビット長を L とすると, t_{rcv} は

$$t_{rcv} = \frac{L}{1.0 \times 10^6 [\text{bit/s}]} \quad (6.4)$$

と見積もることができる.

この電波干渉の模倣手法を図 6.8 を用いて説明する. 図は, 2 つの BM-FRM を順次受信する際の時間の流れを示している. (a) が干渉なしの場合, (b) が干渉ありの場合である. それぞれの図で, 丸印が BM-FRM を受信した時刻である. それぞれ t_0, t_1 とした. BLE フレームの受信に要する時間 t_{rcv} は, 先に詳細を記述したとおりである. 青が 1 つ目のフレーム, 赤が 2 つ目のフレームについて示しており, これらは独立している. BM-FRM を受信した時刻から t_{rcv} が経過するまでに他の BM-FRM を受信しなければ, 電波干渉なしとみなしてその BM-FRM を受理する (図 6.8(a)). 一方, BM-FRM を受信した時刻から t_{rcv} が経過するまでに他の BM-FRM を受信した場合, 電波干渉が発生したとみなしてその両方の BM-FRM を破棄する (図 6.8(b)).

なお, 電波干渉を模倣するための他の手段として, 任意の確率でパケットロスを発生させる方法が考えられる. しかし, BLE のユースケースでは周囲の電波状況は時々刻々と変化する. 固定の確率でパケットロスが発生するよりも, 状況に応じて電波干渉が発生する方がソフトウェアテストのためのユースケースが広がる. パケットロスの発生確率を頻繁に更新することで状況変化に応じることはできるであろう. しかしその場合には, 周囲の電波環境を収集してパケットロス発生率を計算するための追加の仕組みが必要である. BluMoon では, フレーム単位でのエミュレーションを行っているため, 受信したフレームを用いれば, 追加の情報収集なしに電波干渉模倣を行うことができる. よって, 本提案手法を採用した.

6.5 BluMoon の設計と実装

本章では, 6.4 節の設計方針を実現するための BluMoon の具体的なソフトウェア設計および実装について記述する. 本研究では, BluMoon を Linux で動作するソフトウェアとして実装した.

6.5.1 概要

BluMoon は, コントローラの挙動を模倣する BluMoon Controller (BM-CTL) と, 各ノードの位置情報を生成し伝達する Location Generator (LC-GEN) によって構成される. BM-CTL は,

- BluMoon HCI Transceiver (BM-HCI)
- BluMoon Manager (BM-MGR)
- BluMoon Connector (BM-CNC)

という 3 つのモジュールに分けて設計を行った. これは, 機能ごとに分割することで設計

と実装を簡素化するためである。これにより、Bluetooth 規格のアップデートへの追従も容易になると考える。受信電波強度の計算のため必要となる各ノードの位置情報は Location Information (LC-GEN) として格納される。LC-INF は LC-GEN が生成し、各 BM-CTL に伝達する。これらの構成を図 6.9 にまとめた。

6.5.2 BluMoon HCI Transceiver

BluMoon HCI Transceiver (BM-HCI) は、ホストとの HCI フレーム送受信を担う。ホストから HCI データや HCI コマンドを受信した際には、BM-MGR にその内容を伝達する。また、BM-MGR から HCI イベント生成命令や HCI データ送信命令を受信した際には、ホストに HCI イベントや HCI データを送信する。

BM-HCI と Linux の Bluetooth スタックである BlueZ の構成とそれらの関係性を図 6.10 に示す。Linux では、BlueZ が Bluetooth スタックとしてホストの役割を担う。BlueZ では、コア機能を担う BlueZ Core に接続される形でいくつかのドライバが実装されている。ドライバは、実機と接続するための UART ドライバと USB ドライバに加えて、仮想コントローラと接続するための VHCI ドライバがある。デバイスの差異はドライバで吸収されるため、BlueZ Core から上層ではデバイスにかかわらず同様に扱うことができる。すなわち、VHCI ドライバに接続するように仮想コントローラを実装すれば、実機向けのソフトウェアをそのまま動作させることができる。VHCI ドライバは、UNIX ドメインソケット `/dev/vhci` を経由して HCI フレームの送受信を行う。BM-CTL が仮想コントローラとしてふるまうためには、VHCI ソケットを待ち受けるソケットサーバを立ち上げる必要がある。VHCI ドライバと通信する機能を、VHCI ソケットインタフェースとして BM-HCI 内に実装した。BlueZ と通信する仮想コントローラの参考実装としてオープンソースソフトウェア `btvirt` があり、VHCI ソケットインタフェースはこの `btvirt` を拡張して実装した。VHCI ソケットインタフェースが受け取った HCI フレームは、フォーマットを変換して BM-MGR に渡す。また、BM-MGR から HCI イベント生成命令や HCI データ送信命令を受信した際には、それをフォーマット変換して VHCI ソケットインタフェースを経由して BlueZ へ送信する。

6.5.3 BluMoon Manager

BluMoon Manager (BM-MGR) は、送受信に関する BluMoon コントローラ全体の管理を行う。BLE コントローラが持つ Standby, Advertising, Scanning, Initiating, Connection の 5 つの状態 (図 6.4) の管理、BM-FRM の送信制御、受信スロットの制御を行う。

BM-FRM を送信する際には、BLE フレームの形式 (図 6.5) にしたがってフレームを作成し、Channel, TxPower, Location のメタデータを付与して BM-CNC に渡す。Advertising の状態では、BM-MGR が内部にタイマーを持ち、一定間隔で送信を行う。

BLE コントローラは、待ち受け時は指定したチャンネルの BLE フレームを受信することができる。また、待機時の消費電力を削減するために、待ち受けとスリープを繰り返す。これら

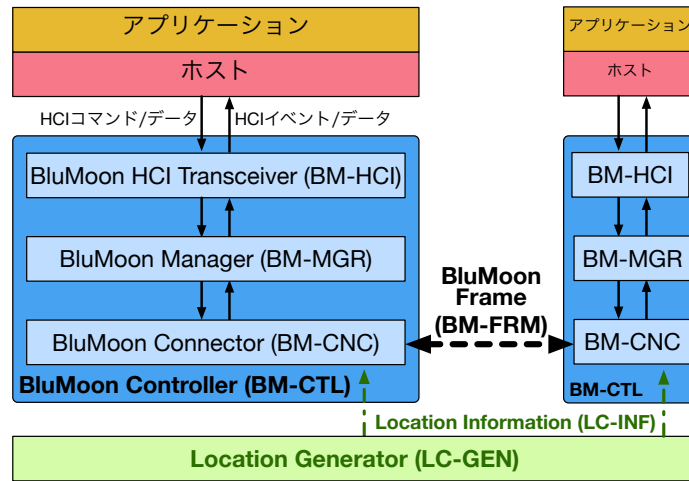


図 6.9. BluMoon の設計構成図.

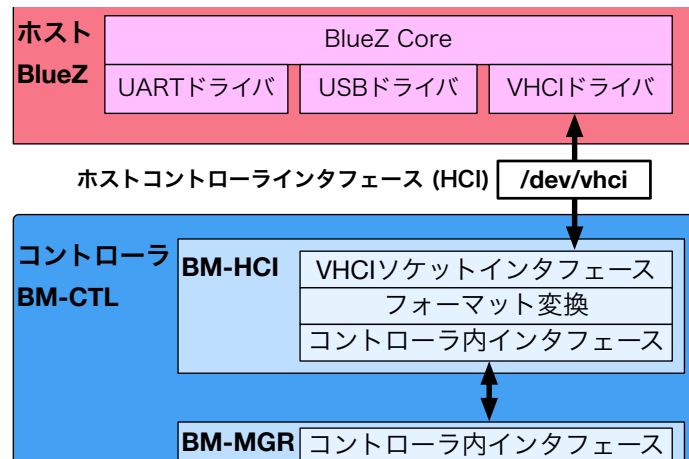


図 6.10. BM-HCI の構成とそのインターフェース. Linux の Bluetooth スタック BlueZ に含まれる VHCI と、HCI によって接続する. この HCI 接続には、UNIX ドメインソケット /dev/vhci が用いられる. BM-MGR とは、コントローラ内インターフェースで独自フォーマットのデータをやり取りする.

の切り替えも BM-MGR が担う. 受信した BM-FRM のメタデータに記述されたチャンネルと、BM-CTL の待ち受けチャンネルを照合し、一致した場合には BM-FRM を受理し、一致しない場合には破棄する.

6.5.4 BluMoon Connector

BluMoon Connector (MB-CNC) は、BM-FRM の実際の送受信を担う. BM-MGR から送信命令を受け取ると、BM-CNC で BM-FRM(図 6.7) を作成し、送信する. BM-FRM は、Generic Network Virtualization Encapsulation (Geneve) でカプセル化して UDP セグメントに格納し(図 6.11)、IP ネットワークにブロードキャストで送信する. BM-FRM を作成

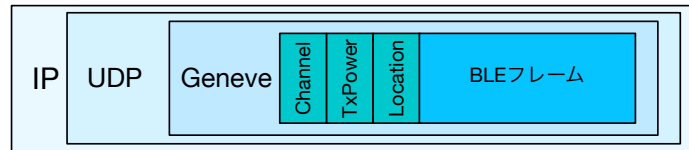


図 6.11. Geneve にカプセル化し UDP セグメントとして格納した BM-FRM.

```
{
  "1": {"x": 65, "y": 75},
  "2": {"x": 25, "y": 40},
  "3": {"x": 25, "y": 55},
  "4": {"x": 75, "y": 90}
}
```

図 6.12. Location Information (LC-INF) の記述例. この例では, ノード ID1~4 の 2 次元座標を指定している.

するための BLE フレームと Channel と TxPower は, 送信命令に含まれる. Location は, LC-GEN から送信される LC-INF から自身の ID の位置座標を保持しておく.

受信時は, 受信した BM-FRM を解析し, 電波干渉の判定と受信電波強度の計算を行う. これらの詳細は 6.4 節に記述したとおりである. 受信した BLE フレームは, RSSI を付与して BM-MGR に渡す.

6.5.5 Location Generator

Location Generator (LC-GEN) は, すべてのノードのノード ID と 2 次元座標を含んだデータを定期的にブロードキャストで送信する. このデータは Location Information (LC-INF) LC-INF は, すべてのノードのノード ID と 2 次元座標を含んだデータである. 図 6.12 のデータ形式で JSON で記述する. あらかじめ位置と時間の関係を定めたシナリオに沿う場合や, マルチエージェントシミュレータのようなシミュレーションで位置を決定する場合などが考えられる. 本論文では, シナリオに沿って LC-INF を送信する LC-GEN を Python スクリプトで実装した.

6.6 評価

6.6.1 機能評価

我々が実装した BluMoon が, 機能要件を満たしていることを確認する. Bluetooth の規格には非常に多くの HCI コマンド, HCI イベントがある. 本論文では, 表 6.2 に示す 6 つの

表 6.2. 対応した HCI コマンドおよび HCI イベント一覧

HCI コマンド・HCI イベント	種別
LE Set Advertising Parameter	
LE Set Advertising Data	
LE Set Advertising Enable	HCI コマンド
LE Set Scan Parameters	
LE Set Scan Enable	
LE Create Connection	
Command Complete	
LE Advertising Report	HCI イベント
LE Connection Complete	

```

$ 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 er
rors:0
        TX bytes:1016 acl:0 sco:0 commands:
42 errors:0

```

図 6.13. hciconfig コマンドの実行結果.

HCI コマンドと 3 つの HCI イベントに対応する実装を行った。これにより、6.2 節で示した LE コントローラの 5 つの状態 (図 6.4) にすべて対応することができ、最低限の要件を満たしたと言える。暗号化や通信パラメータ設定に関する実装は省略したが、これらは今後の追加実装によって対応することが可能である。

BlueZ に含まれる Linux 向けコマンドツール hciconfig の実行結果を図 6.13 に示す。hciconfig は、hci で接続されたコントローラの状態を確認することができるほか、起動と終了などの制御を行うことができる Bluetooth インタフェースの制御コマンドである。このように、エミュレートされた BM-CTL は、HCI を経由して通常の Bluetooth コントローラと同様に扱うことができる。

表 6.3. 計測を実施した環境

筐体	Dell PowerEdge R430
CPU	Intel Xeon E5-2683 v4 (2.1GHz,16core) × 2
メモリ	384GB
OS	Ubuntu 16.04 Server AMD64
Bluetooth スタック	BlueZ 5.43

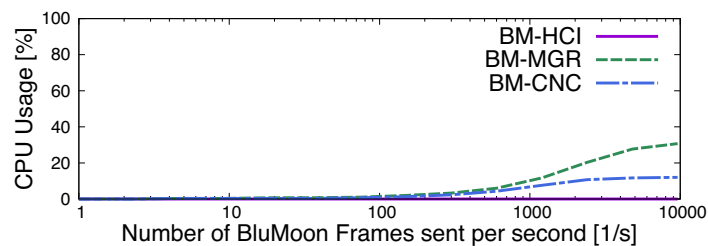


図 6.14. アドバタイズパケット送信時の CPU 使用率. 横軸は 1 秒あたりのアドバタイズ送信数. 縦軸は CPU 使用率. 値は BM-HCI, BM-MGR, BM-CNC のプロセスごとに算出される.

6.6.2 性能評価

BluMoon が実用可能な程度のリソース使用であることを確認するため、BluMoon 実行中の CPU 使用量を測定した。CPU 使用率は、アドバタイズ送信時とアドバタイズ受信時を測定した。計測には、ネットワークテストベッド StarBED[109] のグループ P のノードを用いた。ノードは、10 ギガビットイーサネットスイッチを経由して接続される。測定に使用した機材とソフトウェアの一覧は表 6.3 に示す。送信には HCI コマンドを、受信には `node.js` ライブラリ `noble`[110] を用いて作成した受信スクリプトを実行した。

アドバタイズ送信時の CPU 使用率測定結果を図 6.14 に示す。図は、単位時間あたりのアドバタイズ送信数に応じた CPU 使用率の変化を示す。BM-HCI, BM-MGR, BM-CNC はそれぞれ別プロセスとして実行しているため、プロセスごとの CPU 使用率を示している。アドバタイズ間隔を短くするほど、つまり、単位時間あたりの送信フレームを増やすほど、BM-MGR と BM-CNC の CPU 使用率は増える。一方、BM-HCI は、アドバタイズ開始時に HCI コマンド (LE Set Advertise Enable) を送信するのみで、アドバタイズ送信中には実行する処理がないため CPU を使用しない。Bluetooth バージョン 4.2 の仕様では、アドバタイズ間隔の最小値は 20 ミリ秒と定められている。つまり、1 秒あたりの送信数は最大でも 50 フレームである。この範囲では、CPU リソースをほとんど消費せずに送信を実行することが可能である。

アドバタイズ受信時の CPU 使用率測定結果を図 6.15 に示す。この図は、単位時間あたりのアドバタイズ受信数に応じた CPU 使用率の変化を示す。送信時とは異なり受信時には受信毎

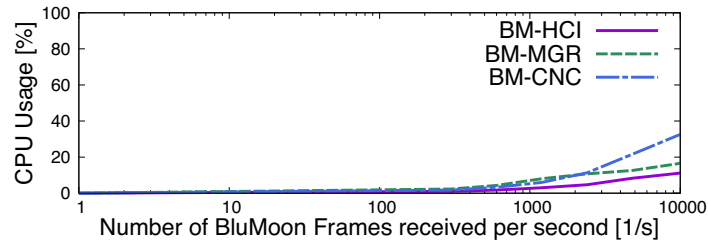


図 6.15. アドバタイズパケット受信時の CPU 使用率. グラフの見方は図 6.14 と同様.

に HCI イベントを生成してホストへ送信する. そのため, BM-HCI, BM-MGR, BM-CNC のすべてのプロセスで受信数に応じて CPU 使用率が上昇する. アドバタイズ受信数に仕様上の上限はない. しかし, 周囲に存在するアドバタイズ送信機には現実的には限りがある. Bluetooth では最大出力によって Class が定められており, 最大出力の最も高い Class1 で到達距離の目安は 100m 程度である. 100m 四方の空間に 1m おきにアドバタイズ送信機が配置される場合には 10000 個程度のアドバタイズ送信機が存在する. これらのアドバタイズ送信機が仕様上の初期既定値 1.28 秒ごとにアドバタイズを送信した場合, 1 秒あたりのアドバタイズ受信数は 7812.5 である. このアドバタイズ受信数の範囲では, 現実的な CPU リソースの消費で受信が可能である.

6.6.3 受信電波強度

BluMoon の設計が要件 (3) を実現したことを示すため, 距離に応じた受信電波強度の計測を行った. 送信機と受信機を設置してアドバタイズパケットの送受信を行い, 送受信機の間隔を変えながら RSSI を記録した. 送受信機の間隔は 0.1~30m とし, 0.1~1m では 0.1m ごと, 1~10m では 1m ごと, 10~30m では 10m ごとに計測した. 各位置での計測は 10 秒間行った. 計測は実環境と BluMoon の両方で行い, 実環境は屋内と屋外の 2 ケースを計測した. 実環境での送受信機には, Raspberry Pi 3 Model B を用いた (図 6.16(a)). 屋外の計測は見通しの良い広場 (図 6.16(b)) で行い, 屋内の計測はガラス製の壁が設置された屋内の廊下 (図 6.16(c)) で実施した.

計測結果を図 6.17 に示す. この図は, 実測値に加えてその対数近似を示す. 実環境と同様に BluMoon のエミュレーション結果においても, 距離に応じて減衰するという電波の特徴が現れている. そして, BluMoon の値は, 屋内での計測と屋外での計測の概ね中間をとっている. これは, BluMoon での計測値が実環境と比べて非現実的ではないことを示す.

なお, ここでの結果は, BluMoon にて (6.2) および (6.3) の式を適応した場合の比較にすぎない. BluMoon では, RSSI の計算を行うことを特徴のひとつとしたが, その計算モデルは本論文で提案したものに限るわけではなく, 必要に応じて置き換えることができる.

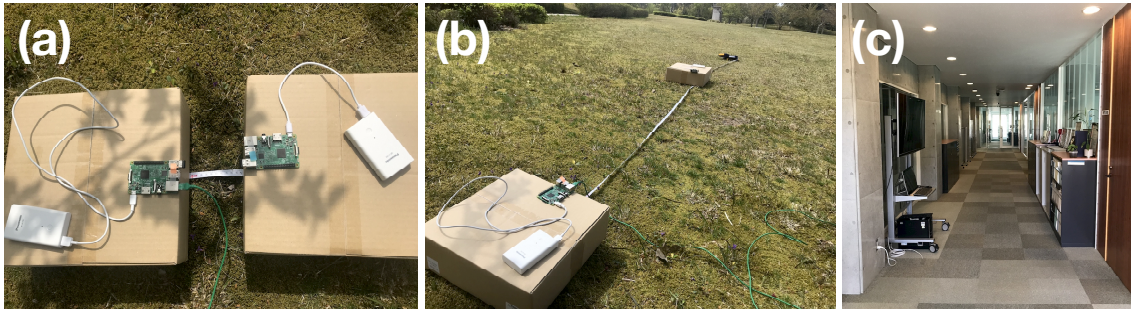


図 6.16. 受信電波強度計測を実施した実環境. (a) 送信機と受信機の配置. (b) 屋外の計測環境. (c) 屋内の計測環境.

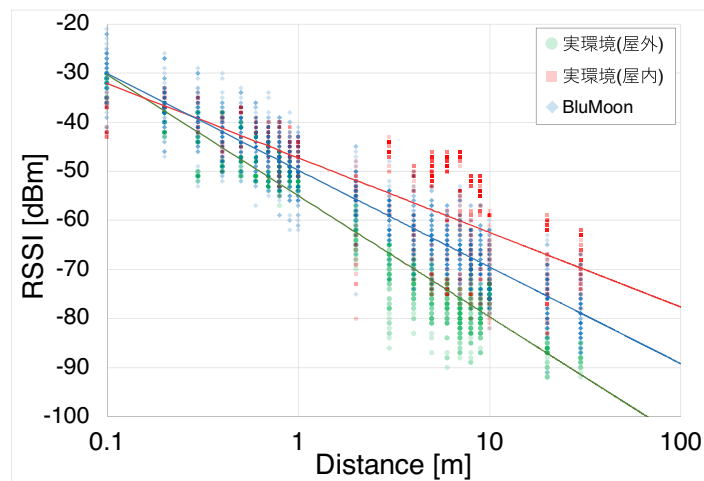


図 6.17. 距離に応じた受信電波強度の計測結果. 横軸の距離は対数軸である. 各点は実測値, 直線は実測値の対数近似を示す.

6.6.4 電波干渉

BluMoon の設計が要件 (4) を実現したことを示すため, 電波干渉源を設置して受信率の計測を行った. 受信率の計測のために, 送信機から送信されるアダバイズパケットの受信数を数えた. 送信機のアダバイズの間隔は Bluetooth の規格で定められる範囲のうち最短の 20msec とした. 理想的には 1 秒あたり 50 個のアダバイズパケットを受信することができる. これを分母として, 何個受信するかを受信率として算出した. また, 送信機以外に, 干渉源として最大 4 台の干渉機を設置した. これらの干渉機は, 送信機と同じく 20msec 間隔でアダバイズパケットを送信し続ける. 実環境での計測は他の 2.4GHz 帯電波が概ね遮蔽されたスチール製倉庫の中で行い, 送受信機および干渉機には Raspberry Pi 3 Model B を用いた (図 6.18).

計測は実環境と BluMoon の両方で行い, さらに理論値とも比較した. 理論値は, 次の通り導出した. あるパケット長のアダバイズパケットを受信する時, 1 台の干渉機のアダバタイ

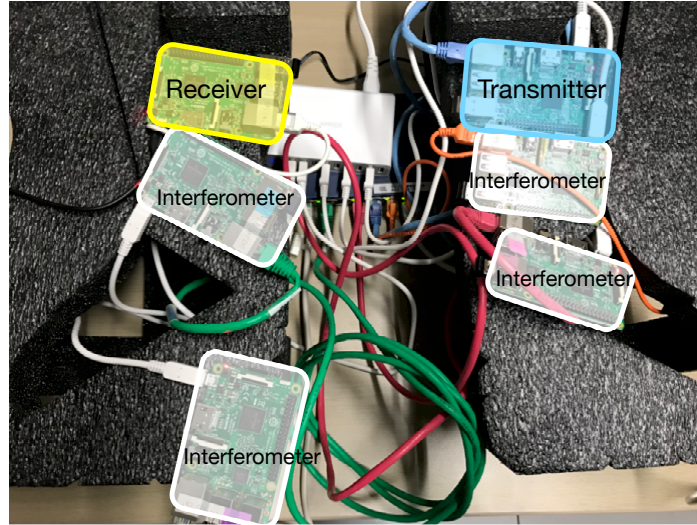
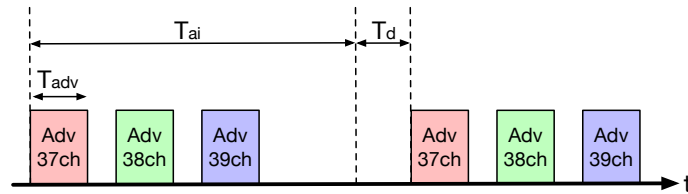


図 6.18. 受信率計測を実施した実環境.

図 6.19. アドバタイズパケット送信における T_{adv} , T_{ai} および T_d の関係.

ズパケットと衝突する確率は

$$P_0 = \frac{2T_{adv}}{T_{ai} + T_d} \quad (6.5)$$

となる. ここで T_{adv} はアドバタイズパケットの送信に要する時間であり, 数式 (6.4) より導出できる. T_{ai} は Advertise Interval と呼ばれるパラメータで, あるアドバタイズパケットを送信してから次のアドバタイズパケットを送信するまでの間隔である. T_d は Advertise Delay と呼ばれるパラメータで, 同じ間隔のアドバタイズパケット同士が継続して衝突することを回避するため挿入されるランダムな待ち時間である. この値はパケット送信ごとに異なるが, 衝突確率の理論値導出に用いるため平均値を T_d とする. これらの関係性は図 6.19 に示した. また, 送信機と干渉機で T_{adv} , T_{ai} および T_d は同じとしている. 干渉機が N 台ある場合に少なくとも 1 つの干渉機のアドバタイズパケットと衝突する確率 P_N は, どの干渉機のアドバタイズパケットとも衝突しないことの余事象の確率となるため

$$P_N = 1 - (1 - P_0)^N \quad (6.6)$$

$$= 1 - \left(1 - \frac{2T_{adv}}{T_{ai} + T_d}\right)^N \quad (6.7)$$

と導出できる.

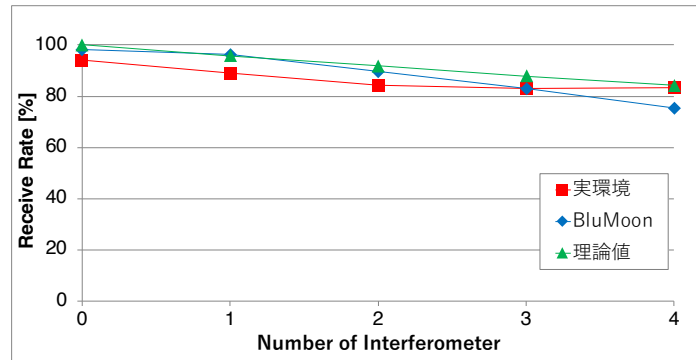


図 6.20. 干渉機の台数に応じた受信率の計測結果.

計測結果を図 6.20 に示す．理論値と実機と同様に，BluMoon でも干渉機の台数が増えるごとに受信率が低下する傾向が現れる．BluMoon と理論値との相関関数は 0.9799 であり，提案した手法によって電波干渉が模倣されていることを示している．なお，実環境と理論値との相関関数は 0.9241 であり，BluMoon の方が理論値に近い計測結果となった．前節でも記述したとおり，実環境での計測では様々な要因によって電波環境が変化し，それによって干渉にもばらつきが出る．実環境を想定した時，BluMoon のエミュレーションにこのばらつきをどのように取り入れるべきかは，今後の検討課題とする．

6.7 ケーススタディ

BluMoon は，BLE アプリケーションのテストを行うことを想定している．そのため，エミュレーション結果が実環境と大きく相違がないことが，有用性につながる．そこで，送受信機の実機を物理空間に設置した実環境での計測と，同じ条件で BluMoon によるエミュレーションを実行し，その結果を比較する．

6.7.1 実験概要

実験は，BLE アドバタイズを用いた近接判定アプリケーションを想定して行う．場所固定で設置したアドバタイズ送信機の周囲を，アドバタイズ受信機が移動し，送信機と受信機の距離に応じた近接の程度を判定する．近接の程度は，Apple 社が提唱する BLE ビーコンの標準規格 iBeacon を参考にして Immediate ($\sim 1\text{m}$)，Near ($1\sim 3\text{m}$)，Far ($3\text{m}\sim$) の 3 段階とし，これらは RSSI から自由空間電波伝搬損失をもとに距離を推定した．近接判定アプリケーションは，受信機側で実行した．アプリケーションは，node.js ライブラリ noble を用いて作成した．このアプリケーションを，実環境とエミュレーション環境で同じものを実行し，結果を比較した．

実環境および BluMoon での実験設定を図 6.21 に示す． $2.9\text{m} \times 5.5\text{m}$ の矩形の空間の内角に，4 台のアドバタイズ送信機を設置する．これらをビーコン 1~4 と呼ぶ．受信機は，それらを囲む軌道を周回移動し，スタート地点から 3 周する．実環境における実験は，情報通信

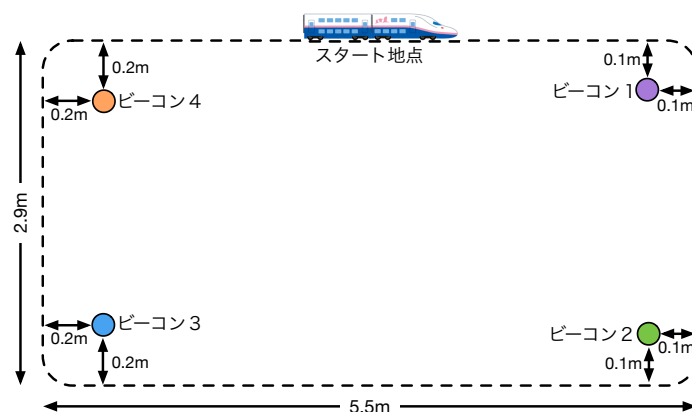


図 6.21. 実験の設定.

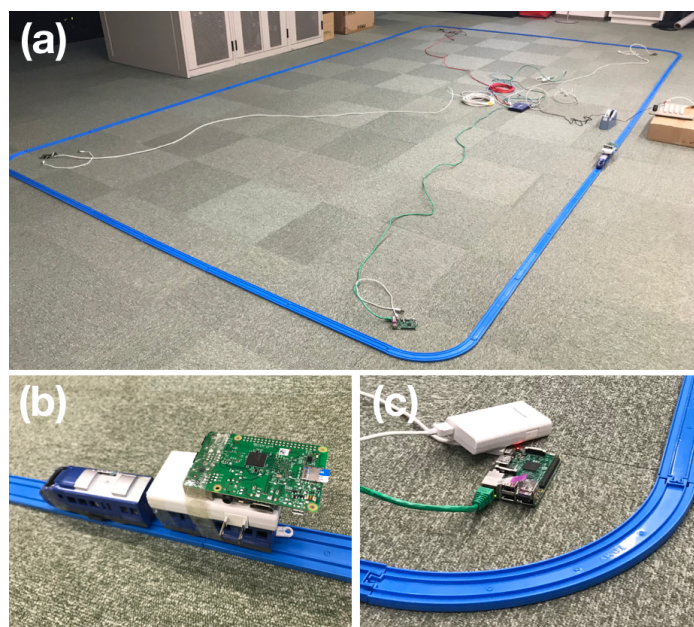


図 6.22. 実験を実施した実環境. (a) 全体像. (b) 受信機. (c) 送信機.

研究機構北陸 StarBED 技術センター内のサーバ室にて実施した。コンクリート製の壁に囲まれ、周囲には金属製ラックが設置されるため反射が発生しうる環境である。アダプタイズ送信機と受信機には Raspberry Pi 3 Model B を用いた。受信機はレールに沿って走る電池駆動の電車型玩具に固定し、等速で移動するように設置した (図 6.22)。BluMoon でも、これと同じ配置と移動を行うシナリオを実行した。なお、BluMoon で用いるパラメータは、送信電力を $-20[\text{dBm}]$ 、AWGN の確率密度関数 (数式 (6.3)) の標準偏差 σ を 5 とした。これらの値は、実環境で予備実験を実施して RSSI を計測してから決定した。

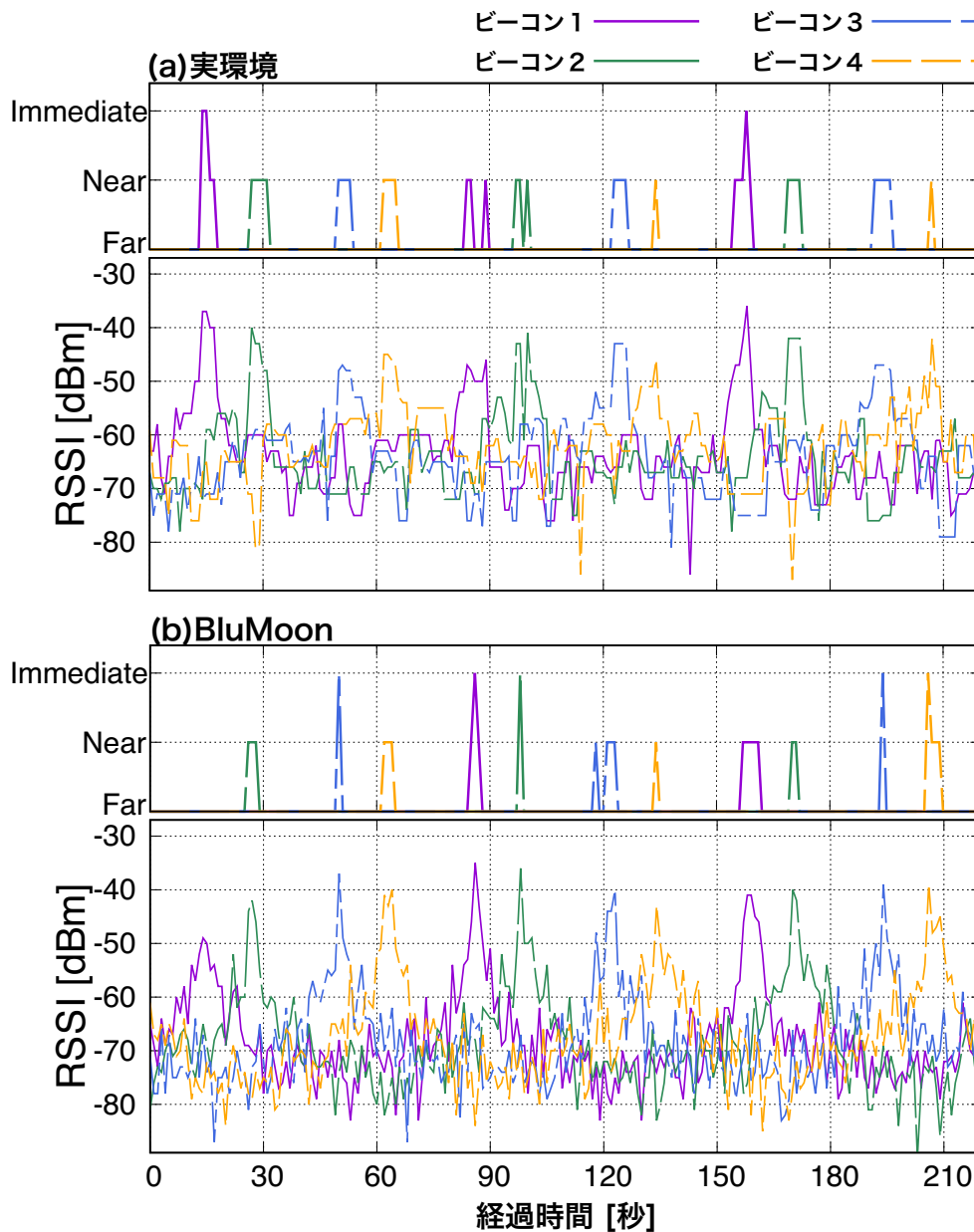


図 6.23. BLE ビーコンを用いた近接判定実験の結果. 近接判定と RSSI 実測データを上下に並べている. (a) 実環境. (b) BluMoon.

6.7.2 結果

実環境と BluMoon にて近接判定アプリケーションを実行した結果を図 6.23 に示す. 図は近接判定結果と RSSI の実測値を示し, 実環境と BluMoon の両方の結果を示している. 実験では 4 台のビーコンが設置された環境を 3 周するため, Near と Immediate を区別しなければ, 計 12 回の近接イベントが発生する. 実環境では 12 回すべての近接を検知した. 一方, BluMoon では 11 回の近接を検知した. 実環境を用意せずともエミュレーションでアプリ

ケーションの動作テストが可能である。

なお、1度の近接検知失敗は、AWGNによりRSSIが弱まったためである。一般的に、エミュレータを用いた実験では、試行によらず同一の結果が得られる。これは同一条件でテストが実行できるという利点であるが、単純なモデルを用いる場合には、実環境で起こっているような電波強度の高頻度のゆらぎは再現することができない。そのため、本論文のRSSI計算にはAWGNを追加している。実験におけるエミュレーションと実環境の結果の差異は、このランダム性を用いた結果である。しかしながら、BluMoonで用いるRSSIの計算モデルはこれに限定するものではなく、必要に応じたモデルを使用することによってより要求に合ったエミュレーションを実現することができる。

6.8 おわりに

本論文では、ソフトウェアシステムテストのためのBLEエミュレータBluMoonを提案し、その設計、実装および評価を行った。BluMoonはソフトウェアシステムのテストに使用することを想定したエミュレータであり、

1. 実機向けソフトウェアが動作すること
2. フレーム単位で通信を行うこと
3. 受信電波強度を計算すること
4. 電波干渉を模倣できること

という要件を挙げた。(1)の要件に対し、HCIを境界としてコントローラをエミュレーションで置き換える設計を提案した。これにより、HCIより上位層では実機向けと同じソフトウェアが動作する。(2)の要件を満たすため、BM-FRMというエミュレーションのための形式を提案した。BM-FRMには、エミュレーションに必要なメタデータと実際のデータ形式に沿ったBLEフレームが格納される。BM-FRMはIPネットワークを通じてUDPで送信される。(3)の要件に対しては、送信端末と受信端末の2点間距離から電波減衰を計算することによってRSSIを模倣する設計を提案した。RSSIは、受信端末側のエミュレータによってフレーム受信毎に計算される。エミュレータ外部に設置されるLC-GENが各端末の位置情報を生成し各エミュレータへ伝達する。(4)の要件に対しては、同一チャンネルのフレームが同時に到着した場合に干渉とみなしてフレームを破棄する設計を導入した。これらの設計をもとに実装を行い、いくつかの計測実験を通じて有用性の評価を行った。機能評価では、HCI経由で実機のBLEコントローラと同様に扱えることを確認した。性能評価では、リソース使用量による評価を行い、BluMoonが現実的なリソース消費の範囲で実行可能であることを示した。また、実機との比較実験により、近接判定アプリケーションのテストというユースケースにおいてBluMoonが活用できることを示した。

本論文では、受信電波強度の模倣に自由空間での電波伝搬損失という単純なモデルを用いた。そのため、BluMoonで模倣している状況は周囲に何も無い開放空間である。しかしこれは、最も単純な例として採用したに過ぎない。実際の空間では、地面や壁、什器などが存在す

るため、電波はこれらに反射して複数の経路で到達するため強度が変わる [111]. 空間中の物体の位置や素材を考慮したより詳細なモデルへの置き換えることで、より実環境に近いエミュレーション環境の構築が可能となる. 地面の反射波を考慮した二波反射モデルや、反射するものの位置から経路を計算するレイトレーシングの手法を使うことで、より正確なエミュレーションを実現することができる. また、本論文では、実環境とエミュレーションの比較として近接判定アプリケーションというユースケースを用いたが、これはユースケースの一例に過ぎない. BluMoon では、実環境と同じデータの BLE フレームの送受信を行うため、センサデータを収集したり、スタンプラリーを実施したりするような様々な種類の BLE アプリケーションをテストすることができる.

ソフトウェア開発におけるテストの手法や論理は、ソフトウェア工学という分野の一体系として様々な知見が積み上げられてきた. しかしながら、BLE アプリケーションのように物理空間の影響を受けるシステムのテストに関しては、議論や情報整理はまだ発展途上である. 我々の別の研究では、BluMoon でのエミュレーション結果を用いて電波環境を物理空間の箱の中で再現し、携帯端末の実機を用いたテストに活用することも行っている [112]. 我々が本論文で示したエミュレーションによるテストの手法をきっかけとして、今後増え続けるであろう周囲の環境を考慮したシステムのテストの知見が蓄積していくことを大いに期待する.

第7章

エミュレーションとその物理化によるテストプラットフォーム: AOBAKO

7.1 はじめに

7.1.1 CPHI としての位置づけ

本節では、ここまでの CPHI に関する議論を元に、CPHI を考慮したシステムの開発環境について提案する。BLE ビーコンを用いたアプリケーションの検証システム AOBAKO の設計と実装について述べる。AOBAKO には、前節で述べた BluMoon を用いた。

7.1.2 背景

ユビキタスコンピューティングにおいてコンテキストウェアはもっとも重要な要素のひとつである [113]。モバイルアプリケーションではすでにコンテキストは活用されており、コンテキストを注入するモバイルアプリケーションテスト platform も研究されている [114][115]。モバイルアプリケーションにおける重要なコンテキストのひとつに、位置情報がある。位置情報の最も主要な測定方法であるグローバル・ポジショニング・システム (Global Positioning System:GPS) は屋内で使用できないため、モバイルアプリケーションが屋内位置を活用するためには他の方法が用いられる。そのひとつが BLE ビーコンを用いた屋内測位である。BLE ビーコンは建物の形状や人の流れ等の要因によって電波の到達範囲が変わるため、事前調査が欠かせない。しかし、実機と人力を用いた現地調査実施のコストは大きい。さらに、人手による現地調査では再現性がない。そのため、コンテキストウェアネスの検証環境が必要となる。

本章では、コンテキストウェアアプリケーション検証のためのシステム AOBAKO を提案する。AOBAKO では、BLE 屋内測位を使ったモバイルアプリケーションのテストをターゲットとする。BLE ビーコン情報の生成には、無線ネットワークエミュレータを活用し、ビーコン送受信機の正確なトラフィックが仮想空間の中で再現される。このエミュレーションには、前章に記述した BluMoon を用いる。エミュレーションの結果を、物理空間に転送する。



図 7.1. AOBACO の全体像

物理空間の端末は、エミュレーション空間にある端末が受信するものと同じビーコンを受信できる。AOBAKO は、コンテキストウェアアプリケーションの検証を手助けするテストベッドの一形態である。このシステムにより、いままで人手に頼り再現性のなかったモバイルアプリケーションにおけるコンテキストの検証が可能となる。

7.2 AOBACO

AOBAKO は、BLE エミュレーションにもとづいて物理空間で電波を発信する仮想ビーコンシステムである。AOBAKO は、BluMoon, AOBACO DESK, AOBACO SCOPE, AOBACO BOX の 4 要素で構成される (図 7.1, 7.2) 対象アプリケーションが稼働するモバイル端末を箱に入れるだけで、任意のビーコン環境でテストが実行できる。本研究では、我々は送信ビーコンのフォーマットを iBeacon[88] のフォーマットとして実装した。

7.2.1 BluMoon: 無線ネットワークエミュレータ

BluMoon[87] は、BLE のフレーム送受信を正確に模擬する無線ネットワークエミュレータである。BluMoon のエミュレーションでは、UDP データグラムでのフレーム送受信を行い、BLE 通信を有線ネットワーク上で模倣する。無線通信の特徴である電波干渉、同一チャネルにおけるフレーム衝突を考慮して計算される。受信電波強度は、送受信機それぞれの位置から計算される。本研究では、BluMoon はネットワークテストベッド StarBED[109] で実行した。

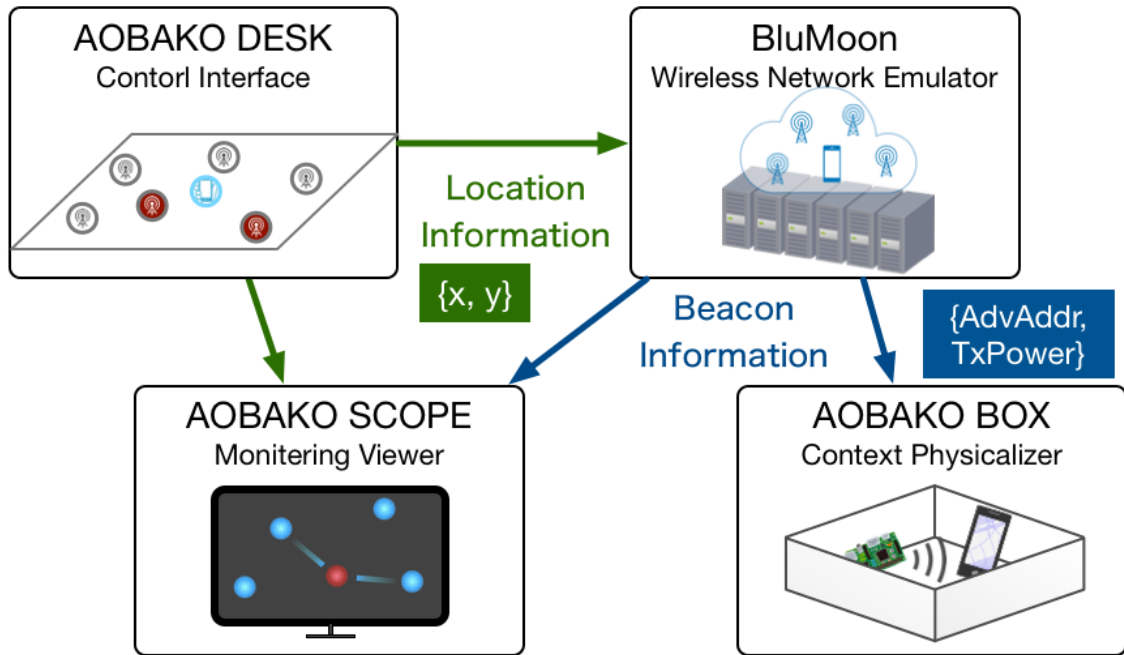


図 7.2. AOBAKO のシステム構成

7.2.2 AOBAKO DESK: 操作インタフェース

AOBAKO DESK(図 7.3) は、ビーコンの送受信機の位置を指定する操作インタフェースである。ユーザは、ビーコン送信機と受信機が描かれた駒を任意の場所に配置する。駒の上にはウェブカメラを設置し、撮影した画像を解析して駒の位置を算出する。計算された位置は Message Queueing Telemetry Transport(MQTT) にて AOBAKO SCOPE と BluMoon に送信される。

7.2.3 AOBAKO SCOPE: 監視ビューワ

AOBAKO SCOPE(図 7.4) は、仮想ビーコン送受信機の位置と仮想ビーコンのフレーム送信を表示する監視ビューワである。青い塊がビーコン送信機、赤い塊がビーコン受信機を示す。エミュレーションでのビーコン受信に合わせて、送信機から受信機に弾が発射される。

7.2.4 AOBAKO BOX: コンテキストの物理化

The AOBAKO BOX, which is a metal box installed in the physical space (Figure 7.5), transfers BLE beacons from virtual to physical space. AOBAKO BOX は、エミュレーションによって導出された BLE ビーコンを、物理空間に設置された金属箱(図 7.5)に転移する。本研究ではこれを「コンテキストの物理化」と名付けた。金属箱と電波遮蔽フィルムは、外部からの 2.4GHz 帯の電波を遮断する。物理的なビーコン発信機として 4 つの Raspberry

Pi が箱の中に設置される (図 7.7).

7.2.5 メッセージ

AOBAKO では、2 種類のメッセージが交換される。Location Information と Beacon Information である。Location Information は、すべてのビーコンコントランシーバの 2 次元座標を含み、一定時間おきに AOBAKO DESK から全ての仮想端末に送信される。Beacon Information は、beacon の物理化のために用いられるパラメータである。これらのメッセージは JavaScript Object Notation (JSON) で記述され、MQTT で送信される。

7.3 ビーコンの物理化

AOBAKO BOX は、BluMoon が生成した Beacon Information をもとに、ビーコンの物理化を行う。ビーコンの物理化をするために、2 つの考慮ポイントがある。それは、送信機多重化と距離偽装である。

7.3.1 送信機多重化

Note that we could only install four physical-beacon transmitters in AOBAKO BOX due to space limitations. 本研究で制作した AOBAKO BOX には、空間上の制約により物理ビーコン送信機を 4 台しか設置できない。しかし、AOBAKO DESK に設置し BluMoon でエミュレートされるビーコンはそれよりも多い。そのため、4 台の物理ビーコン送信機で 4 台よりも多いビーコン送信機がある状況を模倣しないとイケない。iBeacon では、送信元の識別子として Advertise Address(AdvAddr) が使われる。本研究では、この AdvAddr の値をフレーム送信毎に変更することで、多くのビーコンがある状況を模倣する。

7.3.2 距離偽装

AOBAKO BOX 内に置いた物理端末は、AOBAKO DESK で指定した位置に仮想的に配置される。この物理端末を任意の場所に擬似的に移動させるためには、受信するビーコンの距離見積もりを偽装する。iBeacon では、送受信機間の距離 d は

$$d = 10^{(P-R)/20}. \quad (7.1)$$

で表される。ここで R は受信電波強度 (received signal strength indication:RSSI), P は送信機から 1m 離れた地点での RSSI の値である。iBeacon では P は TxPower と表される。AOBAKO BOX 内での物理送信機と物理端末の距離 d_B は

$$d_B = 10^{(P_B-R_B)/20}, \quad (7.2)$$

と算出される。ここで添字 B は AOBAKO BOX を示す。しかしながら、物理送信機は箱の中に固定されているため、 R_B を変えることはできない。そこで、ビーコンを距離 d_B' から受

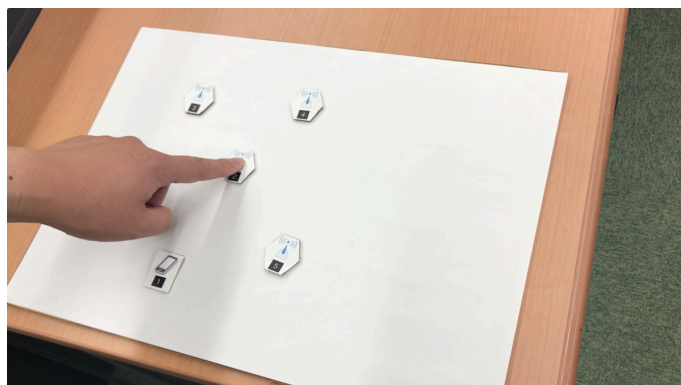


図 7.3. AOBAKO DESK.



図 7.4. AOBAKO SCOPE.



図 7.5. AOBAKO BOX.

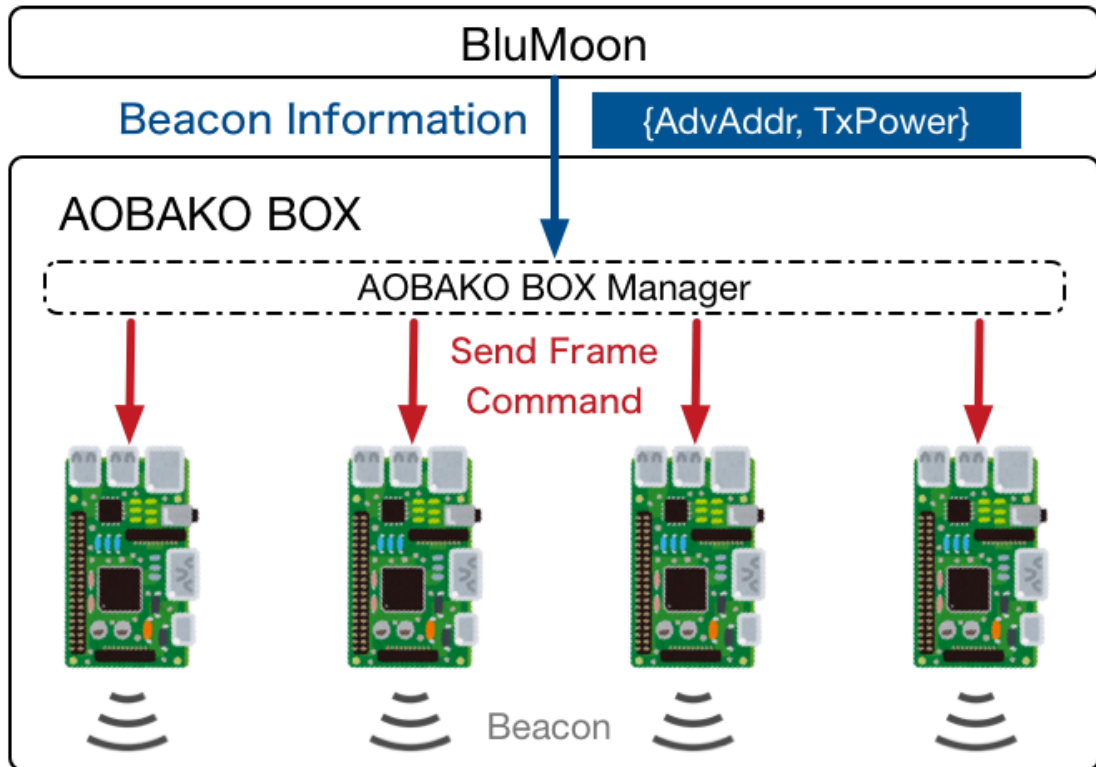


図 7.6. AOBako BOX のシステム構成.

信したように偽装するため、 P_B の値

$$d_B' = 10^{(P_B' - R_B)/20}. \quad (7.3)$$

となるように調整する。ここで d_B' は $d_V = 10^{(P_V - R_V)/20}$ と等しい。つまり、仮想空間での距離に偽装するための P_B' は

$$P_B' = P_V + R_B - R_V. \quad (7.4)$$

と算出できる。

7.3.3 物理ビーコン送信の管理

AOBAKO BOX での物理ビーコン送信は AOBako BOX Manager(図 7.6) で制御される。AOBAKO BOX Manager の実態は Mac で稼働する Python スクリプトである。AOBAKO BOX Manager は、Beacon Information を受け取ったら、その AdvAddr と TxPower をパラメータとしたフレーム送信を Raspberry Pi に指示する。この指示は、4 台の Raspberry Pi にラウンドロビン方式で行う。命令は SSH 経由でのコマンド実行により行う。



図 7.7. AOBAKO BOX の内部構造.

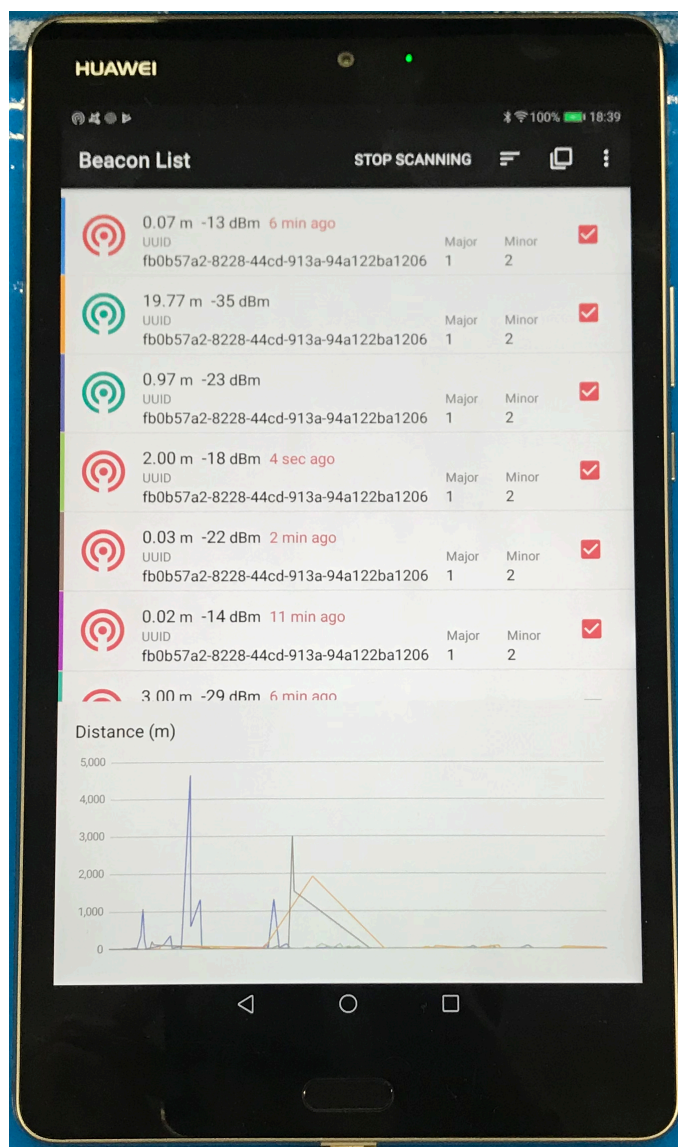


図 7.8. 物理化されたビーコンの受信.

7.3.4 結果

AOBAKO BOX に置かれた携帯端末は，物理化された仮想ビーコンを受信する．図 7.8 は，さまざまな距離からの多数のビーコンを受信している様子を示す．ビーコンの受信には，Google Play で公開されている Android アプリ [116] を用いた．

7.4 まとめ

本研究では，モバイルなどのコンテキストウェアアプリケーションのテスト環境の一例として AOBAKO を提案した．AOBAKO では，生成したコンテキストを物理化することが鍵となる．本研究では，AOBAKO のシステム実装を行い，提案するコンセプトが実現可能であることを示した．本研究で提案する AOBAKO のシステムがコンテキストウェアアプリケーションのソフトウェアテストにおいて大いに役立つことを期待する．

第 8 章

結論

8.1 議論

本論文では、cyber と human のインタラクションにおいて physical に着目して分解することで整理を行うことを試みた。機器に搭載されるセンサが増加の一途をたどる中、今後物理量の取得は複雑かつ重要な要素となる。CPHI の概念は、機器やサービスを設計する上で役立つようになると考える。

また、第 3 章の SleepTyping では、キーボードの押下を用いて、睡眠時の姿勢を取得した。ここでは、もともと文字入力を目的としたキー押下の圧力の物理量とセンサを用いて、異なる情報を伝達可能であることを示した。物理量に着目することで、物理センサの目的外使用の可能性を示す。

本論文では、物理現象を模倣するシミュレーションおよびエミュレーションを利用することでテストプラットフォームを構築することを提案した。この提案が実現可能であることを示すため、AOBAKO というテストプラットフォームシステムを構築した。AOBAKO では、電波強度の計算モデルを用いて物理を模倣した。このモデルは非常に単純なものではあるが、モジュラブルで入れ替え可能な設計として構築しているため、シミュレーションモデルをより高精細なものに置き換えることによって要件に合わせた物理空間の模倣が可能である。物理モデルを可換にする設計は、CPHI 開発支援プラットフォームの設計の普遍的な考え方として役立つと考える。

8.2 おわりに

本論文では、CPHI という新たな概念を提唱した。これは、サイバー情報と物理と人間を統合的に扱う、新しい概念である。本論文では、CPHI のパターンとして

- Cyberization from Human (人間からのサイバー化)
- Realization to Human (人間への意識化)
- Human-Physical Merging (人間と物理の情報融合)
- Physical Augmentation (物理的拡張)

の4つを提案した。第1部では、CPHIのインタフェースに関する事例を示した。3章では、SleepTypingというCyberization from Humanの事例を示した。4章では、PICALAというRealization to Humanの事例を示した。5章では、Augmented TypingというPhysical Augmentationの事例を示した。第2部では、CPHIのシステム開発をサポートする研究事例を示した。6章では、BluMoonというBLEエミュレータの事例を示した。これには、干渉と減衰という物理における問題の模倣を取り入れている。7章では、AOBAKOという人間と物理とコンピュータの構成要素を統合的に検証できるプラットフォームの事例を示した。これらは、既存のHCIの概念からみると、物理の概念が新たに追加された。既存のCPSの概念からみると、人間の概念が新たに追加された。

インターネットが登場してから今日まで、人間はPCやスマートフォンといった情報端末を通じてインターネットにアクセスしてきた。しかし、これはまだインターネットを活用した社会の初期段階のプロトタイプに過ぎない。物理世界にあるデバイスの情報やそのデバイスが取得する情報を取り込んだり、その情報を社会システムの前提として組み込んだり、処理した結果を物理世界にフィードバックすることで、インターネットの真のポテンシャルが発揮される。Internet of Things (IoT) という潮流とともに、人間の意思や行動を介さずに自律的にインターネットに接続する機器が増えてきた。Mark Weiser がユビキタスコンピューティングとして提唱した、21世紀のコンピュータの実装は、まさにいま始まったところである。CPHIの概念は、そのような複雑性をなす世界で、システムを考案し設計し実装するための道標になるであろう。

謝辞

本研究を行い本論文を執筆するにあたり、多くの方にご指導やご助言を頂きました。

主指導教員の丹 康雄教授，副指導教員のリム 勇仁准教授には，ゼミでの議論を中心として，博士課程を通じて多くのことを学びました。篠田 陽一教授には，研究ミーティングを通じて多くのことを学びました。長きに渡る指導に感謝します。本論文の審査員をお引き受け頂き，論文を改善するための多くの助言を頂いた Razvan Beuran 特任准教授，金沢工業大学 中沢実教授に感謝します。副テーマ研究の指導教員をお引き受け頂き，副テーマ研究を進める上で多くの助言を頂いた明治大学 中村 聡史教授に感謝します。

研究に関する議論を日々行い研究を支援して頂いた Marios Sioutis 研究員，牧野 義樹研究員をはじめとする丹・リム研究室の皆様，情報通信研究機構北陸 StarBED 技術センターの宮地 利幸センター長，三輪 信介主任研究員，井上 朋哉研究員、明石 邦夫研究員をはじめとする研究員，研究技術員の皆様，株式会社工科堂 岡田 崇代表取締役，大阪大学 高野 祐輝特任准教授に感謝します。

本研究のテーマの着想を得るにあたり，ニコニコ学会βや ABPro 等の大学外の発表の場とそこでの議論が大きな助けになりました。産業技術総合研究所 江渡 浩一郎主任研究員およびニコニコ学会β関係者の皆様，明治大学 宮下 芳明教授および宮下研究室の皆様，津田塾大学 栗原 一貴教授に感謝します。

本研究は，博士後期課程の社会人コースとして，業務と並行して行いました。これを支持してくださった株式会社東芝 研究開発センターおよび設計開発センター，クウジット株式会社，明治大学 総合数理学部 先端メディアサイエンス学科，情報通信研究機構の同僚の皆様に感謝します。

参考文献

- [1] John Von Neumann. *First Draft of a Report on the EDVAC*. 1945.
- [2] Ivan E Sutherland. Sketchpad a man-machine graphical communication system. *Simulation*, Vol. 2, No. 5, pp. R-3, 1964.
- [3] William K English, Douglas C Engelbart, and Melvyn L Berman. Display-selection techniques for text manipulation. *IEEE Transactions on Human Factors in Electronics*, No. 1, pp. 5-15, 1967.
- [4] Douglas C Engelbart. Workstation history and the augmented knowledge workshop." . In *Proceedings of the ACM Conference on the History of Personal Workstations*, pp. 73-83, 1986.
- [5] Alan Kay and Adele Goldberg. Personal dynamic media. *multimedia-from wagner to virtual reality*, pp. 167-178, 1977.
- [6] Apple. macOS. <https://www.apple.com/macOS/>.
- [7] Microsoft. Windows. <https://www.microsoft.com/windows/>.
- [8] Robert W Scheifler and Jim Gettys. The X window system. *ACM Transactions on Graphics (TOG)*, Vol. 5, No. 2, pp. 79-109, 1986.
- [9] Apple. iPhone. <https://www.apple.com/iphone/>.
- [10] Apple. iPad. <https://www.apple.com/ipad/>.
- [11] Microsoft. Kinect. <https://developer.microsoft.com/windows/kinect/>.
- [12] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, Vol. 19, No. 2, pp. 4-10, 2012.
- [13] Sony Interactive Entertainment. PlayStation VR. <https://www.jp.playstation.com/psvr/>.
- [14] Alan C Kay. *A personal computer for children of all ages*. Xerox Palo Alto Research Center Palo Alto, CA, 1972.
- [15] David Howes. *Empire of the Senses*. Berg Publishers Oxford, 2005.
- [16] RD Portugal and Benar Fux Svaiter. Weber-Fechner law and the optimality of the logarithmic scale. *Minds and Machines*, Vol. 21, No. 1, pp. 73-81, 2011.
- [17] James J Gibson. The theory of affordances. *Hilldale, USA*, Vol. 1, p. 2, 1977.
- [18] Donald A Norman, et al. *The psychology of everyday things*, Vol. 5. Basic books

- New York, 1988.
- [19] Edward A Lee. Cyber physical systems: Design challenges. In *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, pp. 363–369. IEEE, 2008.
- [20] Leadership Under Challenge: Information Technology R&D in a Competitive World - An Assessment of the Federal Networking and Information Technology R&D Program. President’ s Council of Advisors on Science and Technology (PCAST), 2007.
- [21] 奥村洋. CPS 研究の世界的潮流と日本の現状. 研究 技術 計画, Vol. 32, No. 3, pp. 251–265, 2017.
- [22] Norbert Wiener. *Cybernetics or Control and Communication in the Animal and the Machine*, Vol. 25. MIT press, 1965.
- [23] Mark Weiser. The computer for the 21st century. *Scientific american*, Vol. 265, No. 3, pp. 94–105, 1991.
- [24] Bill N Schilit and Marvin M Theimer. Disseminating active mop infonncition to mobile hosts. *IEEE network*, 1994.
- [25] Anind K Dey. Understanding and using context. *Personal and ubiquitous computing*, Vol. 5, No. 1, pp. 4–7, 2001.
- [26] Michael Knappmeyer, Saad Liaquat Kiani, Eike Steffen Reetz, Nigel Baker, and Ralf Tonjes. Survey of context provisioning middleware. *IEEE Communications Surveys & Tutorials*, Vol. 15, No. 3, pp. 1492–1519, 2013.
- [27] Hiroshi Ishii, Dávid Lakatos, Leonardo Bonanni, and Jean-Baptiste Labrune. Radical atoms: beyond tangible bits, toward transformable materials. *interactions*, Vol. 19, No. 1, pp. 38–51, 2012.
- [28] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. inFORM: dynamic physical affordances and constraints through shape and object actuation. In *Uist*, Vol. 13, pp. 417–426, 2013.
- [29] Hiroshi Ishii, Daniel Leithinger, Sean Follmer, Amit Zoran, Philipp Schoessler, and Jared Counts. Transform: Embodiment of radical atoms at milano design week. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 687–694. ACM, 2015.
- [30] Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pp. 234–241. ACM, 1997.
- [31] Gunar Schirner, Deniz Erdogmus, Kaushik Chowdhury, and Taskin Padir. The future of human-in-the-loop cyber-physical systems. *Computer*, No. 1, pp. 36–45, 2013.
- [32] Sulayman K Sowe, Eric Simmon, Koji Zettsu, Frederic de Vault, and Irena Bojanova. Cyber-physical-human systems: Putting people in the loop. *IT professional*, Vol. 18,

- No. 1, pp. 10–13, 2016.
- [33] Marios Sioutis, Yuto Lim, and Yasuo Tan. Achieving optimal illumination conditions using local search. In *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*, pp. 168–172. IEEE, 2015.
- [34] 西田佳史, 武田正資, 森武俊, 溝口博, 佐藤知正. 圧力センサによる睡眠中に呼吸・体位の無侵襲・無拘束な計測. *日本ロボット学会誌*, Vol. 16, No. 5, pp. 705–711, 1998.
- [35] 亀山研一, 鈴木琢治, 行谷まち子. 快眠のための睡眠判定と睡眠モニタシステム. *東芝レビュー*, Vol. 61, No. 10, pp. 41–44, 2006.
- [36] 城和男, 萩原啓. 赤外線モーションセンサを用いた睡眠時体動の評価. *生体医工学*, Vol. 47, No. 1, pp. 7–14, 2009.
- [37] 岸本圭史, 赤堀顕光, 小栗宏次. 3軸加速度センサを用いた睡眠時における姿勢の推定. *電子情報通信学会技術研究報告. MBE, ME とバイオサイバネティクス*, Vol. 105, No. 456, pp. 45–48, 2005.
- [38] 庵前修, 前川卓也ほか. Wi-Fi 電波強度の変化を利用した無拘束な睡眠時体動判定手法の提案. *研究報告ユビキタスコンピューティングシステム (UBI)*, Vol. 2015, No. 7, pp. 1–8, 2015.
- [39] タニタ. 睡眠計スリースキャン sl-504. http://www.tanita.co.jp/product/g/_TSL504WH/.
- [40] オムロン. 睡眠計 hsl-101. <http://www.wellnesslink.jp/p/about/app07.html>.
- [41] オムロン. ねむり時間計. <http://www.healthcare.omron.co.jp/sp/hsl002c/>.
- [42] Fitbit. Flex. <http://www.fitbit.com/jp/flex/>.
- [43] 三浦淳, 湯浅友典, 千葉武雄, 相津佳永, 佐々木春喜. 擬似夜明け療法による生体リズム位相調節の試み. *室蘭工業大学紀要*, Vol. 63, pp. 117–121, 2014.
- [44] 西田健志. 学会イベントにおけるコミュニケーション促進の継続的实践: コミュニティの一員としてのシステム開発 (小特集 学会イベント支援). *情報処理*, Vol. 56, No. 5, pp. 458–464, 2015.
- [45] 栗原一貴. 放送化の時代のプレゼンテーション支援システム (小特集学会イベント支援). *情報処理*, Vol. 56, No. 5, pp. 465–471, 2015.
- [46] 綾塚祐二, 河口信夫. 参加者が作る会議支援システム: Wiss challenge (<特集>インタラクティブソフトウェア). *コンピュータソフトウェア*, Vol. 23, No. 4, pp. 76–81, oct 2006.
- [47] Jun Rekimoto, Yuji Ayatsuka, Hitoraka Uoi, and Toshifumi Arai. Adding another communication channel to reality: An experience with a chat-augmented conference. In *CHI 98 Conference Summary on Human Factors in Computing Systems*, pp. 271–272. ACM, 1998.
- [48] Joseph F McCarthy, et al. Digital backchannels in shared physical spaces: experiences at an academic conference. In *CHI'05 extended abstracts on Human factors in computing systems*, pp. 1641–1644. ACM, 2005.

- [49] Drew Harry, Joshua Green, and Judith Donath. Backchan. nl: integrating backchannels in physical space. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 1361–1370. ACM, 2009.
- [50] 西田健志, 栗原一貴, 後藤真孝. On-air forum: リアルタイムコンテンツ視聴中のコミュニケーション支援システムの設計とその実証実験. *コンピュータ ソフトウェア*, Vol. 28, No. 2, pp. 183–192, 2011.
- [51] C. Ross, M. Terras, C. Warwick, and A. Welsh. Enabled backchannel: conference twitter use by digital humanists. *Journal of Documentation*, Vol. 67, No. 2, pp. 214–237, 2011.
- [52] ニコニコ動画. <http://www.nicovideo.jp/>.
- [53] 江渡浩一郎. ユーザー参加型の価値を追究する新しい学会 ニコニコ学会βの試み. *情報管理*, Vol. 55, No. 7, pp. 489–501, 2012.
- [54] 井上良太, 白松俊, 大園忠親, 新谷虎松ほか. 発表中の資料へのフィードバックに基づくインタラクティブプレゼンテーションシステムの実現. *情報処理学会論文誌*, Vol. 56, No. 10, pp. 2011–2021, 2015.
- [55] Kazutaka Kurihara, Toshio Mochizuki, Hiroki Oura, Mio Tsubakimoto, Toshihisa Nishimori, Jun Nakahara, Yuhei Yamauchi, and Shin-ichi Watanabe. Borderless canvas: Development of a multi-display discussion software for knowledge-emergent presentations. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Vol. 2009, pp. 3676–3688, 2009.
- [56] 加藤由訓, 苗村健. ラジへえ : 声の効果音を用いた感想共有メディア (<特集>アート&エンタテインメント3). *日本バーチャルリアリティ学会論文誌*, Vol. 18, No. 3, pp. 345–356, sep 2013.
- [57] PFU. Happy hacking keyboard. <http://www.pfu.fujitsu.com/hhkeyboard/>.
- [58] Kinesis. Ergonomic keyboards and accessories. <https://www.kinesis-ergo.com/>.
- [59] Optimus. Optimus maximus keyboard. <http://www.artlebedev.com/everything/optimus/>, 2008.
- [60] Florian Block, Hans Gellersen, and Nicolas Villar. Touch-display keyboards: transforming keyboards into interactive surfaces. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems (CHI 2010)*, pp. 1145–1154, 2010.
- [61] Apple. Macbook pro - apple. <https://www.apple.com/macbook-pro/>, 2016.
- [62] 岩井俊雄. 映像装置としてのピアノ, 1995.
- [63] faithroom. ピアノにプロジェクションマッピング — ラピュタ・アレンジ / projection mapping on piano "castle in the sky". <https://www.youtube.com/watch?v=vNcrJ97zdFo>, 2014.
- [64] H. Ishii, C. Wisneski, J. Orbanes, B. Chun, and J. Paradiso. PingPongPlus: design of an athletic-tangible interface for computer-supported cooperative. *Proceedings*

- of the SIGCHI conference on Human factors in computing systems: the CHI is the limit, pp. 394–401, 1999.
- [65] Jun Kato, Takashi Nakashima, Hideki Takeoka, Kazunori Ogasawara, Kazuma Mura, Toshinari Shimokawa, and Masaaki Sugimoto. Openpool: Community-based prototyping of digitally-augmented billiard table. In *GCCE '13: Proceedings of the 2nd IEEE Global Conference on Consumer Electronics*, pp. 175–176, 2013.
- [66] Philippe Dubost. Magic keyboard. <https://vimeo.com/45154003>, 2012.
- [67] hecomi. Visual io - concept of mixed reality-based interface - - youtube. <https://www.youtube.com/watch?v=HC1WDhaDzO8>, 2016.
- [68] Yoshinari Takegawa, E. Tayanagi, M. Tsubakimoto, and M. Hirata. Evaluation of a piano learning support system focusing on the learning process. In *Proceeding of World Conference on Educational Media and Technology (EdMedia2013)*, pp. 2306–2314, 2013.
- [69] Katja Rogers, Amrei Rohlig, Matthias Weing, Jan Gugenheimer, Bastian Konings, Melina Klepsch, Florian Schaub, Enrico Rukzio, Tina Seufert, and Michael Weber. P.i.a.n.o.: Faster piano learning with interactive projection. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS 2014)*, pp. 149–158, 2014.
- [70] 土井麻由佳, 宮下芳明. プロジェクションマッピングによる箏演奏学習支援システム. 情報処理学会研究報告ヒューマンコンピュータインタラクション (HCI), 第 2017 巻, pp. 1–8, 2017.
- [71] Koji Tsukada, Keita Watanabe, Daisuke Akatsuka, , and Maho Oki. Fabnavi: Support system to assemble physical objects using visual instructions. In *Fab10*, 2014.
- [72] Yoh Akiyama and Homei Miyashita. Projectron mapping: The exercise and extension of augmented workspaces for learning electronic modeling through projection mapping. In *Proceedings of the adjunct publication of the 27th annual ACM symposium on User interface software and technology*, pp. 57–58. ACM, 2014.
- [73] Hironori Yoshida, Takeo Igarashi, Yusuke Obuchi, Yosuke Takami, Jun Sato, Mika Araki, Masaaki Miki, Kosuke Nagata, Kazuhide Sakai, and Syunsuke Igarashi. Architecture-Scale Human-Assisted Additive Manufacturing. *Acm Transactions on Graphics*, Vol. 34, No. 4, pp. 1–8, 2015.
- [74] 橋本菜摘, 椎尾一郎. Studi/o : プロジェクションマッピングによるトイブロックの組み立て・記録支援. 情報処理学会論文誌, Vol. 57, No. 12, pp. 2577–2588, 2016.
- [75] 若園祐作, 暦本純一. Claymore: 粘土を用いた造形の支援. インタラクティブシステムとソフトウェアに関するワークショップ 2014 (WISS2014), pp. 147–152, 2013.
- [76] Microsoft. Hololens. <https://www.microsoft.com/hololens>, 2016.
- [77] 川崎智博, 井手口健. 動画映像から受ける印象の因子分析と映像再生速度の各因子に

- 与える影響 (研究速報). 電子情報通信学会論文誌. A, 基礎・境界, Vol. 85, No. 9, pp. 1022–1025, sep 2002.
- [78] yumu19. cyberkeyboard 反応まとめ - togetter まとめ. <https://togetter.com/li/946782>, 2016.
- [79] Apple. 3d touch - ios - apple developer. <https://developer.apple.com/ios/3d-touch/>, 2015.
- [80] 辻本進, 曾我真人, 瀧寛和. 拡張現実感を用いたタイピングスキル学習支援環境の構築. 電子情報通信学会技術研究報告. ET, 教育工学, 第 111 巻, pp. 149–153, 2012.
- [81] 田村拓也, 曾我真人, 瀧寛和. 運指の誤りの診断とアドバイスを与えるタイピングスキル学習支援環境. 電子情報通信学会技術研究報告. ET, 教育工学, 第 113 巻, pp. 29–34, 2013.
- [82] Takaaki Imamura, Takayuki NAGAI, and Hiroshi NAKANO. A simple eye-gaze detecting function for touch-typing training system. In *Global Engineering Education Conference (EDUCON), 2012 IEEE*, pp. 1–5, 2012.
- [83] Bluetooth. <https://www.bluetooth.com/>.
- [84] Thomas R Henderson, Mathieu Lacage, George F Riley, Craig Dowell, and Joseph Kopena. Network simulations with the ns-3 simulator. *SIGCOMM demonstration*, Vol. 14, No. 14, p. 527, 2008.
- [85] Kunio Akashi, Tomoya Inoue, Shingo Yasuda, Yuuki Takano, and Yoichi Shinoda. NETorium: high-fidelity scalable wireless network emulator. In *Proceedings of the 12th Asian Internet Engineering Conference*, pp. 25–32. ACM, 2016.
- [86] 湯村翼, 明石邦夫, 井上朋哉. 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. 情報処理学会研究報告ユビキタスコンピューティングシステム (UBI), Vol. 2017, No. 12, pp. 1–7, 2017.
- [87] Tsubasa Yumura, Kunio Akashi, and Tomoya Inoue. Blumoon: Bluetooth low energy emulation system with software-implemented controller. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 860–865. IEEE, 2018.
- [88] Apple. Getting started with ibeacon version 1.0, 2014.
- [89] Google. Eddystone format. <https://developers.google.com/beacons/eddytone/>, 2016.
- [90] BlueZ. <http://www.bluez.org/>.
- [91] Google. Android Studio. <https://developer.android.com/studio/>, 2013.
- [92] Apple. Technical note tn2295: Testing core bluetooth applications in the ios simulator. https://developer.apple.com/library/archive/technotes/_index.html, 2012.
- [93] BLE Peripheral Simulator. <https://github.com/WebBluetoothCG/ble-test-peripheral-android/>.

- [94] LightBlue Explorer. <https://itunes.apple.com/jp/app/lightblue-explorer/id557428110>.
- [95] Google. Wear OS by Google. <https://wearos.google.com/>, 2014.
- [96] Apple. Xcode, 2003.
- [97] Jérémie Albert, Tegawendé F Bissyandé, Yérom-David Bromberg, Serge Chaumette, and Laurent Réveillere. Ubipan: A bluetooth extended personal area network. In *Complex, Intelligent and Software Intensive Systems (CISIS), 2010 International Conference on*, pp. 774–778. IEEE, 2010.
- [98] Kazuma Tsuda, Hidekazu Suzuki, Kensaku Asahi, and Akira Watanabe. Proposal for a seamless connection method for remotely located bluetooth devices. In *2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*, pp. 78–79, 2014.
- [99] 岡田真実, 鈴木秀和. Bluetooth over DTLS による IoT デバイスの遠隔制御システム. *情報処理学会論文誌コンシューマ・デバイス & システム (CDS)*, Vol. 8, No. 2, pp. 34–42, 2018.
- [100] T.J. Smith, Stefan Saroiu, and Alec Wolman. 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.
- [101] Razvan Beuran, Junya Nakata, Takashi Okada, Lan Tien Nguyen, Yasuo Tan, and Yoichi Shinoda. A multi-purpose wireless network emulator: QOMET. *Proceedings - International Conference on Advanced Information Networking and Applications, AINA*, pp. 223–228, 2008.
- [102] Razvan Beuran, Lan Tien Nguyen, Toshiyuki Miyachi, Junya Nakata, Ken-ichi Chinen, Yasuo Tan, and Yoichi Shinoda. Qomb: A wireless network emulation testbed. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pp. 1–6, 2009.
- [103] Razvan Beuran, Junya Nakata, Yasuo Tan, and Yoichi Shinoda. Emulation Testbed for IEEE 802.15.4 Networked Systems. *IEICE transactions on communications*, Vol. 95, No. 9, pp. 2892–2905, 2012.
- [104] Razvan Beuran, Shingo Yasuda, Tomoya Inoue, Yuuki Takano, Toshiyuki Miyachi, and Yoichi Shinoda. DynamiQ: A Tool for Dynamic Emulation of Networks. *EAI Endorsed Trans. Self-Adaptive Systems*, Vol. 1, No. 2, p. e1, 2015.
- [105] 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.
- [106] Jouni Mailnen. mac8011_hwsim. https://wireless.wiki.kernel.org/en/users/Drivers/mac80211_hwsim, 2008.
- [107] 真野浩, 猿渡俊介. 無線システムエミュレータの実装と評価. *情報処理学会論文誌*,

- Vol. 55, No. 5, pp. 1541–1554, 2014.
- [108] Ixia. Network Emulator II. <https://www.ixiacom.com/products/network-emulator-ii/>.
- [109] Toshiyuki Miyachi, Takeshi Nakagawa, Ken-ichi Chinen, Shinsuke Miwa, and Yoichi Shinoda. StarBED and SpringOS architectures and their performance. In *International Conference on Testbeds and Research Infrastructures*, pp. 43–58. Springer, 2011.
- [110] Sandeep Mistry. noble. <https://github.com/sandeepmistry/noble>, 2015.
- [111] Jakub Neburka, Zdenek Tlamsa, Vlastimil Benes, Ladislav Polak, Ondrej Kaller, Libor Bolecek, Jiri Sebesta, and Tomas Kratochvil. Study of the performance of rssi based bluetooth smart indoor positioning. In *Radioelektronika (RADIOELEKTRONIKA), 2016 26th International Conference*, pp. 121–125. IEEE, 2016.
- [112] Tsubasa Yumura, Masatoshi Enomoto, Kunio Akashi, Futoshi Hirose, Tomoya Inoue, Satoshi Uda, Toshiyuki Miyachi, Yasuo Tan, and Yoichi Shinoda. Aobako: A testbed for context-aware applications with physicalizing virtual beacons. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pp. 476–479. ACM, 2018.
- [113] Anind K. Dey. Understanding and using context. *Personal Ubiquitous Computing*, Vol. 5, No. 1, pp. 4–7, 2001.
- [114] Chu Luo, Miikka Kuutila, Simon Klakegg, Denzil Ferreira, Huber Flores, Jorge Goncalves, Mika Mäntylä, and Vassilis Kostakos. Testaware: A laboratory-oriented testing tool for mobile context-aware applications. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, Vol. 1, No. 3, pp. 80:1–80:29, 2017.
- [115] Manoj R. Rege, Vlado Handziski, and Adam Wolisz. Realistic context generation for mobile app testing and performance evaluation. In *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 297–308, March 2017.
- [116] contexagon. Beacon analyzer, 2017. <https://play.google.com/store/apps/details?id=com.contexagon.beaconanalyzer>.

本研究に関する発表

論文誌

- 湯村 翼, 明石 邦夫, 井上 朋哉, 丹 康雄, BluMoon: ソフトウェアシステムテストのための Bluetooth Low Energy エミュレータ (投稿準備中)

国際会議 (査読付き)

- Tsubasa Yumura, Kunio Akashi, Tomoya Inoue, BluMoon: Bluetooth Low Energy Emulation System with Software-Implemented Controller, IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2018, pp. 860-865
- Tsubasa Yumura, Masatoshi Enomoto, Kunio Akashi, Futoshi Hirose, Tomoya Inoue, Satoshi Uda, Toshiyuki Miyachi, Yasuo Tan, Yoichi Shinoda, AOBako: A Testbed for Context-Aware Applications with Physicalizing Virtual Beacons, ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018, pp. 476-479
- Tsubasa Yumura, Yuto Lim, Yasuo Tan, PICALA: An Interactive Presentation System to Share Reaction of Audiences with Light Color, Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems, 2019 (to appear)
- Tsubasa Yumura, Satoshi Nakamura, Augmented Typing: Augmentation of Keyboard Typing Experience by Adding Visual and Sound Effects, Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems, 2019 (to appear) 【Best Poster/Demo Award 受賞】

国内会議 (査読付き)

- 湯村 翼, リム 勇仁, 丹 康雄, PICALA: 聴講者の感情を照明色で表現するインタラクティブなプレゼンテーション, NICOGRAPH 2016 予稿集, 2016, pp. 57-64

国内会議 (査読なし)

- 湯村 翼, リム 勇仁, 丹 康雄, SleepTyping: 配置や数を自由に変更できる睡眠時姿勢センサの開発, 情報処理学会研究報告ユビキタスコンピューティングシステム (UBI), 2015, pp. 1-6
- 湯村 翼, リム 勇仁, 丹 康雄, PICALA: プレゼンテーションにおける照明色による聴講者の感情共有システム, エンタテインメントコンピューティングシンポジウム 2015 論文集, 2015, pp. 18-24
- 石井 優衣, 栗原 一貴, 湯村 翼, 日常の動作を複数のゲームの操作に再利用する「ながらゲーム」の提案, インタラクション 2017 論文集, 2017, pp. 325-327
- 湯村 翼, 中村 聡史, Augmented Typing: 映像と音の演出付与による キーボードタイピング体験の拡張, 情報処理学会研究報告エンタテインメントコンピューティング (EC), 2017, pp. 1-9
- 湯村 翼, 明石 邦夫, 井上 朋哉, 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, 情報処理学会研究報告ユビキタスコンピューティングシステム (UBI), 2017, pp. 1-7
- 石井 優衣, 栗原 一貴, 湯村 翼, 睡眠時間の副次利用によるエンタテインメント価値創出, 第 25 回インタラクティブシステムとソフトウェアに関するワークショップ (WISS2017) 論文集, 2017, pp 1-2

報道等

- 日本テレビ SENSORS, 2015 年 6 月 14 日 (Augmented Typing に関して)
- フジテレビ とくダネ!, 2016 年 3 月 7 日 (Augmented Typing に関して)
- TBS サンデー・ジャポン, 2016 年 3 月 13 日 (Augmented Typing に関して)
- 日本テレビ NEXT クリエイターズ, 2018 年 3 月 15 日 (Augmented Typing に関して)
- INTERNET Watch, 「ダムカード」の次は「基準点カード」がアツい! “G 空間” コンテストで注目のアツい入賞作品たち, 2018 年 11 月 29 日 (AOBAKO に関して)
- 日刊工業新聞 2019 年 6 月 18 日 (AOBAKO に関して)