Exploiting Induced skin electric potential for body-area IoT system functions

Yan, Zhenyu

2020

Yan, Z. (2020). Exploiting Induced skin electric potential for body-area IoT system functions.
Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/137802>

<https://doi.org/10.32657/10356/137802>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0
International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 18:59:24 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

**EXPLOITING INDUCED SKIN ELECTRIC
POTENTIAL FOR BODY-AREA IOT SYSTEM
FUNCTIONS**

YAN ZHENYU

**SCHOOL OF COMPUTER SCIENCE AND
ENGINEERING**

2020

**EXPLOITING INDUCED SKIN ELECTRIC
POTENTIAL FOR BODY-AREA IOT SYSTEM
FUNCTIONS**

YAN ZHENYU

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2020

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

September 6th, 2019
.....

Date

.....

YAN Zhenyu

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

September 7th, 2019

.....

Date

.....

Asst. Prof. TAN Rui

Authorship Attribution Statement

This thesis contains material from three papers published in the following peer-reviewed journals and conferences in which I am listed as an author.

Chapter 3 contains material from a conference and its extended journal version:

- Zhenyu Yan, Yang Li, Rui Tan, and Jun Huang. Application-Layer Clock Synchronization for Wearables Using Skin Electric Potentials Induced by Powerline Radiation. In Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems (SenSys '17), Article 10, 14 pages. DOI: 10.1145/3131672.3131681.
- Zhenyu Yan, Rui Tan, Yang Li, and Jun Huang. Wearables Clock Synchronization Using Skin Electric Potentials. IEEE Transactions on Mobile Computing, 2018. DOI: 10.1109/TMC.2018.2884897. Accepted in December, 2018. In press.

The contributions of the co-authors are as follows:

- YAN Zhenyu co-designed the approach, performed laboratory work, analyzed the data, and prepared the paper drafts.
- TAN Rui provided the project direction, co-designed the approach, and revised the paper drafts.
- LI Yang implemented the phase locked loop algorithm and assisted experiments.
- HUANG Jun contributed to the design of the approach and revised the paper drafts.

Chapter 4 contains material from a conference paper:

- Zhenyu Yan, Qun Song, Rui Tan, Yang Li, and Adams Wai Kin Kong. Towards Touch-to-Access Device Authentication Using Induced Body Electric Potentials. In The 25th Annual International Conference on Mobile Computing and Networking (MobiCom '19). Article 23, 16 pages. DOI: 10.1145/3300061.3300118.

The contributions of the co-authors are as follows:

- YAN Zhenyu led the design and implementation of the approach, led all laboratory work, prepared the manuscript drafts.

- SONG Qun participated in the approach design discussion and laboratory work.
- TAN Rui provided the project direction, supervised the research, revised the drafts.
- LI Yang participated in discussions and provided the magnetic field experiment environment.
- Adams Wai Kin Kong provided information on biometrics-based authentication and revised the drafts.

September 6th, 2019

.....

Date

.....

YAN Zhenyu

Acknowledgements

Foremost, I would like to express my greatest gratitude to my advisor Prof. Rui Tan for the continuous guidance throughout my Ph.D. studies. His profound knowledge and engaging mentorship greatly inspired my research. As a researcher, Prof. Tan has been the role model demonstrating the diligence, critical thinking, and dedication to me. I feel so fortunate for being Prof. Tan's first Ph.D. student.

I would like to thank my friends Qun Song, Chaojie Gu, Linshan Jiang for cheering me up and helping me all the time. Thanks to Yutao Jiao, Jue Tian, Sreejaya Viswanathan, RongRong Wang, Yingbo Liu, Van Duc Le, Xiaozheng Gao for participating in the experiments. Thanks to Dr. Yang Li, Prof. Jun Huang, and Prof. Adams Wai Kin Kong for the guidance of the work shown in this thesis. Thanks to Dr. Xiaoxuan Lu for the sharing and discussions. I would like to thank my Thesis Advisory Committee members and Oral Defence Panel members, Prof. Jun Luo, Prof. Ping Wang, Prof. Yiling Mo, Prof. Jiamin Zheng, Prof. Mo Li, Prof. A S Madhukumar, for their comments. In addition, I really appreciate the comments given by the thesis examiners.

Last but not least, I would like to thank my parents for the constant support and unconditional love.

To my dear family

Abstract

Internet-of-Things (IoT) devices are increasingly deployed in indoor environments. In particular, there will be an IoT device proliferation in the household settings. For instance, it is estimated that by 2022, a typical family home could contain more than 500 smart devices. IoT devices are penetrating the industry environments to enable the realization of Industry 4.0. IoT devices are generally equipped with processing units, various sensors, communication modules, and software platforms to meet the application needs. A key characteristic of the IoT systems is the heterogeneity of hardware and software. Specifically, there are numerous IoT hardware platforms with diverse specifications and a number of embedded IoT operating systems (OSes) such as Android Things, Wear OS, Contiki, and Arduino. Such heterogeneity introduces significant challenges to the implementation of various basic system functions, such as clock synchronization and device authentication. Clock synchronization aims at keeping the clocks of the IoT devices under consideration having the same value all the time. However, the design of clock synchronization faces the trade-off between the synchronization accuracy and the universality over a variety of platforms. Specifically, high synchronization accuracy generally requires platform-dependent hardware-level network packet timestamping, which may not be available on any IoT platform. Thus, designing clock synchronization approaches to improve the Pareto frontier of accuracy versus universality is of great research interest. Device authentication verifies whether a device has a certain property (e.g., possession of a secret key). It can be used to manage the accesses to IoT devices. However, the existing device authentication approaches require cumbersome sensor hardware or user's manual input. Thus, they are devised for specific devices and are generally inapplicable to a range of heterogeneous IoT devices.

This thesis studies the clock synchronization and the device authentication problems for the IoT devices that can interact with the human bodies in the indoor environments. Such body-interacting devices include the wearable devices (e.g.,

smart watches, smart glasses, or medical sensors) and the indoor objects that the user can physically touch (e.g., light switches, remote controls for television, or voice assistants). This thesis designs and develops new clock synchronization and device authentication approach based on a new sensing modality called *induced skin electric potential* (iSEP) that provides desirable properties to solve the aforementioned challenges. Through extensive measurement studies on the electromagnetic radiation signals emitted from the electric power lines in indoor environments, we find that human body can act as an effective antenna and generate iSEP. The iSEP signal has three properties: First, it shows the same periodicity as the powerline voltages, which is 50 or 60 Hz. In particular, two iSEP signals have good synchrony across a large area, e.g., a city, due to the synchrony of the power grid voltages. Second, iSEP signals from the same location and the same body are similar, whereas those from different bodies are distinct. Third, sensing iSEP only requires a conductive wire connecting human body skin and the pin of an analog-to-digital converter (ADC). As ADC is widely available on most microcontrollers (MCUs), such a simple sensing mechanism can be universal for any IoT objects equipped with MCUs. Based on iSEP, we design TouchSync and TouchAuth systems for clock synchronization and device authentication for wearables and touchable IoT objects.

TouchSync achieves millisecond synchronization accuracy while preserving universality in that it uses standard system calls only, such as reading system clock, sampling sensors, and sending/receiving network messages. It is based on the synchrony of iSEP signals collected from the same human body or different human bodies. TouchSync integrates the iSEP signal into the universal principle of Network Time Protocol and solves an integer ambiguity problem by fusing the ambiguous results in multiple synchronization rounds to conclude an accurate clock offset between two synchronizing objects. With our shared code, TouchSync can be readily integrated into any wearable applications. Example applications include delivering audio stream synchronously for wireless earphones, motion analysis of multiple on-body sensors, and gesture detection for multiple wearable gaming controllers.

The design of TouchAuth is based on the electrostatics of iSEP generation and a resulting property, i.e., the iSEPs from two close locations and the same human body are similar. Extensive experiments verify the above property and show that

TouchAuth achieves high-profile receiver operating characteristics in implementing the touch-to-access authentication policy. Our experiments show that a range of possible interfering sources including appliances' electromagnetic emanations and noise injections into the power network do not affect the performance of TouchAuth. TouchAuth can be used in various applications with touchable IoT devices, e.g., family members wearing tokens can authenticate TV remote controller to access personalized programmes. In addition, the user with a wearable TouchAuth token can unlock the smart phone or authenticate a payment terminal by a simply touch.

The designs of TouchSync and TouchAuth exemplify an important cyber-physical approach that leverages on certain properties of the physical processes (i.e., ambient electric field and generation of iSEP) to improve performance, usability, and security of the basic IoT system functions. Applying such a cyber-physical approach to implement a broader scope of system functions for a wider range of IoT systems beyond wearable and touchable devices will be of great research interest in the future work.

Contents

Acknowledgements	ix
Abstract	xiii
List of Figures	xxi
List of Tables	xxiii
List of Acronyms	xxv
1 Introduction	1
1.1 Application-layer Clock Synchronization for Wearables Using iSEPs	2
1.2 Touch-to-Access Device Authentication Using iSEPs	6
1.3 Summay of Contributions	10
1.4 Thesis Structure	10
2 Preliminaries on Body Antenna Effect	11
2.1 Body Antenna Effect	11
2.2 Electric Field (EF) From Electrical Cabling	12
2.3 Interactions Among EF, Sensor, and Human Body	13
3 Wearables Clock Synchronization Using iSEPs	17
3.1 Related Work	17
3.2 Background	19
3.2.1 NTP Principle and Packet Timestamping	19
3.2.2 Performance of NTP over BLE Connection	21
3.3 Measurement Study	22
3.3.1 Measurement Setup	22
3.3.2 iSEP Sensing under Various Settings	23
3.3.2.1 Shared ground	24
3.3.2.2 Independent grounds	26
3.3.2.3 Summary	28
3.4 Design of TouchSync	29
3.4.1 TouchSync Workflow	29

3.4.2	iSEP Signal Processing	30
3.4.2.1	Band-pass filter (BPF) or mean removal filter (MRF)	30
3.4.2.2	Zero crossing detector (ZCD)	31
3.4.2.3	Phase-locked loop (PLL)	31
3.4.3	NTP Assisted with Dirac Combs	31
3.4.3.1	Protocol for a synchronization session	31
3.4.3.2	Clock offset analysis	33
3.4.4	Integer Ambiguity Solver (IAS)	35
3.4.4.1	An example of unique solution	36
3.4.4.2	Program for solving integer ambiguity	36
3.4.4.3	Convergence speed	38
3.4.4.4	Discussions	39
3.5	TouchSync with Internal Periodic Signal	40
3.5.1	Extended Design	40
3.5.1.1	Protocol for IPS-based TouchSync	41
3.5.1.2	Adaptive Period Mechanism (APM)	42
3.5.2	Numeric Experiments	42
3.5.2.1	Distribution of clock offset estimation errors	43
3.5.2.2	Convergence speed	44
3.5.3	Impact of Clock Skews	44
3.6	Implementation of TouchSync	45
3.7	Performance Evaluation	47
3.7.1	Signal Strength and Wearing Position	47
3.7.2	Impact of High-Power Appliances on TouchSync	48
3.7.3	Evaluation in Various Environments	50
3.7.3.1	Laboratory	50
3.7.3.2	Home	52
3.7.3.3	Office	53
3.7.3.4	Corridor	53
3.7.4	TouchSync-over-Internet	54
4	Touch-to-Access Device Authentication Using iSEPs	57
4.1	Background and Related Work	57
4.1.1	Device Authentication and Key Generation	57
4.1.2	Human Body Coupled Capacitive Sensing	58
4.1.3	Other Related Studies	59
4.2	System Overview and Research Objective	60
4.2.1	System Model	60
4.2.2	Threat Model	61
4.2.3	Approach Overview	61
4.2.4	Research Objective	62
4.3	Measurement Study	63
4.3.1	Measurement Setup	63

4.3.2	Measurement Results	64
4.3.2.1	Insensitivity to time-varying magnetic field (MF)	64
4.3.2.2	iSEPs on the same body	65
4.3.2.3	iSEPs on different bodies	68
4.4	Same-Body Contact Detection	70
4.4.1	Detection Algorithm	70
4.4.2	Assessment metrics	71
4.4.3	Mimicry Attack	73
4.5	Evaluation	73
4.5.1	Performance across Different Wearers	73
4.5.2	Various Indoor Environments	77
4.5.3	Various Possible Interfering Sources	77
4.5.4	Impact of Signal Length ℓ	79
4.5.5	TouchAuth Devices' Proximity	80
4.5.6	Mimicry Attack	80
4.5.7	Heterogeneous Devices	81
4.6	Limitations and Discussions	81
4.6.1	Proximity Requirement	82
4.6.2	iSEP Injection Attack	82
4.6.3	Other Interferences	83
4.6.4	Applicability to Outdoors	83
4.6.5	Applicability to Implantable Medical Devices (IMDs)	83
4.6.6	Comparison with Existing Approaches	84
5	Conclusion and Future Work	85
	List of Author's Publications	87
	References	89

List of Figures

2.1	The body antenna effect.	12
2.2	Mains hum measured by a sensor (without human body contact) placed close to a power cord supporting a 2 kW heater.	13
2.3	Impact of an uncharged conductor (e.g., human body) on an EF from a point charge. Arrows represent the directions of the EF; dotted curves represent equipotential lines.	14
3.1	NTP principle and packet timestamping.	20
3.2	Performance of NTP over a BLE connection.	21
3.3	Flora.	22
3.4	TouchSync Prototypes.	22
3.5	No human body contact.	24
3.6	iSEPs on a same wearer (shared ground).	24
3.7	iSEPs on different wearers (shared ground).	25
3.8	Absolute time displacement $ \epsilon $ between the EMR signals captured by the two Floras in various scenarios. Error bar represents (5%, 95%) confidence interval in one minute of data.	25
3.9	iSEPs on a same wearer (independent grounds).	26
3.10	iSEPs on different wearers (independent grounds).	27
3.11	Normalized range and time displacement $ \epsilon $ of iSEPs under different skin conditions and body orientations.	27
3.12	A synchronization process of TouchSync.	29
3.13	iSEP signal processing pipeline.	30
3.14	A synchronization session of TouchSync. The vertical arrows represent the Dirac comb impulses generated from the iSEP.	32
3.15	An example of solving the integer ambiguity. The transmissions of the auxiliary <code>reply2</code> messages are omitted in the illustration. . . .	36
3.16	Convergence speed of IAS.	39
3.17	Distribution of absolute clock offset estimation errors.	42
3.18	An example of IPS-based TouchSync. (unit: ms)	43
3.19	Convergence speed under various settings of IPS period.	43
3.20	Impact of signal strength.	48
3.21	Wearing position.	48
3.22	EMR signals near an electric oven and a microwave oven.	49
3.23	Errors introduced into time displacements of the EMR signals. . . .	49

3.24	Laboratory floor plan with test points marked.	52
3.25	Home floor plan with test points marked.	52
3.26	One-way delays over a ngrok tunnel.	54
3.27	Accuracy of TouchSync-over-Internet.	54
4.1	Left: A use scenario where the smart watch personalizes a remote control and the associated media system by a touch; Right: authentication process.	61
4.2	Z1.	63
4.3	Sensor placements.	63
4.4	Time-varying magnetic field generator.	64
4.5	iSEPs measured by two sensors in the same palm when the holder sits in a chair.	66
4.6	iSEPs measured by a sensor in the same palm when the wearer stands at different spots in the lab.	66
4.7	iSEPs at different locations of Person A.	67
4.8	APCC in different placements.	68
4.9	APCC under different clock offsets.	68
4.10	iSEPs measured by three sensors on two persons who sit steadily 1 m apart.	68
4.11	iSEPs measured by three sensors on two persons who sit 1 m apart and perform random hand movements.	69
4.12	Detection performance of APCC-TouchAuth when ℓ is 0.5 s, 1 s, and 2 s, respectively.	72
4.13	ROCs for 12 different wearers with the invalid authenticatee device. The x -axis and y -axis of each subfigure are α and β , respectively.	74
4.14	ROCs for 8 different wearers with the valid authenticatee device. The x -axis and y -axis of each subfigure are α and β , respectively.	75
4.15	Experiments in indoor environments.	76
4.16	ROCs in various environments.	76
4.17	ROCs with various nearby appliances.	77
4.18	ROCs with interference and skin moisture.	78
4.19	β versus ℓ	79
4.20	Sensor proximity.	79
4.21	β vs. ℓ under mimicry attack.	80
4.22	A Flora as the valid authenticatee.	80

List of Tables

3.1	Storage and compute overhead of TouchSync.	46
3.2	Signal strength and TouchSync accuracy.	51
4.1	Comparison with existing approaches.	84

List of Acronyms

ADC	Analog-to-digital Converter
APCC	Absolute Pearson Correlation Coefficient
APM	Adaptive Period Mechanism
BLE	Bluetooth Low Energy
BPF	Band-pass Filter
CSP	Clock Synchronization Protocol
DC	Direct Current
DTWD	Dynamic Time Warping Distance
ECG	Electrocardiography
EF	Electric Field
EMG	Electromyogram
EMI	Electromagnetic Interference
FAR	False Acceptance Rate
FRR	False Rejection Rate
FTSP	Flooding Time Synchronization Protocol
GPIO	General-purpose Input/Output
HCI	Host Controller Interface
HDP	Bluetooth Health Device Profile
IAS	Integer Ambiguity Solver
IBC	Intra-body Communication
iSEP	Induced Skin Electric Potential
IIR	Infinite Impulse Response
IMD	Implantable Medical Device
IMU	Inertial Measurement Unit
IoT	Internet-of-Things
IPS	Internal Periodic Signal

LI	Last Impulse
MAC	Media Access Control
MAE	Mean Absolute Error
MCU	Microcontroller
MF	Magnetic Field
MRF	Mean Removal Filter
NFC	Near-field Communication
NTP	Network Time Protocol
OS	Operating System
PI	Proportional Integral
PKI	Public Key Infrastructure
PLL	Phase-locked Loop
PPG	Photoplethysmogram
PTP	Precision Time Protocol
RBS	Reference-Broadcast Synchronization
RDS	Radio Data System
RMSE	Root Mean Square Error
ROC	Receiver Operating Characteristic
RPi	Raspberry Pi
RSSI	Received Signal Strength Indication
RTT	Round-trip Time
Rx	Reception
TAR	True Acceptance Rate
TDMA	Time-division Multiple Access
TLS	Transport Layer Security
TPSN	Timing-sync Protocol for Sensor Networks
Tx	Transmission
WSN	Wireless Sensor Network
ZC	Zero Crossing
ZCD	Zero Crossing Detector

Chapter 1

Introduction

Physical objects in our daily lives are increasingly equipped with sensing, computing, communication, and actuation capabilities. A huge number of such smart objects, together with the cloud computing back end, use the Internet as the communication media to form the Internet of Things (IoT), which will be the largest global cyber system in the coming decades. With the Internet as the communication infrastructure, the application layer will be a main area for IoT innovations. Various basic system functions in the application layer have been devised for general-purpose Internet hosts, such as time distribution, encryption services, and unified authentication like OpenID. However, the adaptation of these legacy system functions designed for resource-rich Internet hosts to the IoT objects need to be studied, because IoT objects may be short of the needed computing resources and human-machine interfaces. Moreover, different from the current Internet hosts that are mainly used for information exchange, IoT objects will be mainly tasked to sense and actuate for certain purposes. As such, new basic system functions will be needed for IoT.

Increased heterogeneity in both hardware and software platforms is a key characteristic of IoT. For example, unlike the current Internet hosts that use a limited number of *de facto* standard network interfaces (Ethernet, Wi-Fi, and cellular network), IoT objects additionally use a spectrum of network interfaces (including various 2.4GHz and sub-GHz technologies). The diverse network interfaces with distinct physical-layer and link-layer properties introduce challenges for implementing accurate network packet timestamping and clock synchronization as two basic

system functions. As another example, different from the standard password-based authentication on Internet hosts, diverse hardware has been employed for IoT objects for implementing new authentication approaches, such as various biometric scanners, face recognition cameras, or voice recognition microphones. Such heterogeneous authentication mechanisms make the envisaged unified authentication more challenging.

To address the heterogeneity-induced challenges, this thesis proposes to adopt a cyber-physical approach to leverage on the sensing capabilities of IoT objects and exploit certain properties of the physical ambient to develop low-cost, lightweight, and effective basic system functions for IoT objects. Specifically, this thesis studies two basic system functions of clock synchronization and device authentication for the wearable and touchable IoT objects found in indoor environments. It studies and exploits a new sensing modality of induced skin electric potential (iSEP) that is due to the electrostatic coupling between the human body and the electrical cabling of the building. Three properties of iSEP, i.e., periodicity, synchrony, and body-area similarity, enable lightweight, universal, and effective clock synchronization and device authentication for wearable and touchable IoT objects. The following two sections of this chapter introduce the detailed background of these two basic system functions and present our iSEP-based approaches.

1.1 Application-layer Clock Synchronization for Wearables Using iSEPs

The annual worldwide shipments of consumer wearables (e.g., smart watches, wristbands, eyewears, and clothing) grew by 17% in 2017 ([Gartner, 2017](#)). This rapid growth is expected to continue, projected to be 504 million units shipped in 2021 ([Gartner, 2017](#)). Along with the proliferation of consumer wearables, specialized domains such as clinical/home healthcare ([Chan, EstèVe, Fourniols, Escriba, & Campo, 2012](#)) and exercise/sport analysis ([Mokaya, Lucas, Noh, & Zhang, 2016](#)) are increasingly adopting smart wearable apparatuses. In the body-area networks formed by these wearables, a variety of system functions and applications depend on tight clock synchronization among the nodes. For instance, motion analysis ([Lorincz et al., 2009](#)) and muscle activity monitoring ([Mokaya, Lucas, Noh, &](#)

Zhang, 2015; Mokaya et al., 2016) require sensory data from multiple tightly synchronized nodes.

While current wearable systems adopt customized, proprietary clock synchronization approaches (Apple Inc., 2015), we envisage a wide spectrum of interoperable wearables that can synchronize with each other to enable more novel applications. In the envisaged scheme, an application developer can readily synchronize any two communicating wearables using high-level and standard system calls provided by their operating systems (OSes), such as reading system clock, transmitting and receiving network messages. However, the design of clock synchronization approaches faces a fundamental trade-off between the synchronization accuracy and the universality for heterogeneous platforms. This is because a high synchronization accuracy generally requires low-level timestamping for the synchronization packets, which may be unavailable on the hardware platforms or inaccessible to the application developer.

We illustrate this accuracy-universality trade-off using the Network Time Protocol (NTP) (*NTP: The Network Time Protocol*, 2017) and the Precision Time Protocol (PTP) (IEEE, 2008). NTP synchronizes a slave node and a master node by recording their clock values when a UDP synchronization packet is passed to and received from the sender's and receiver's OSes, respectively. Thus, NTP is universal in that it can be deployed to any host through UDP. However, as the application-layer timestamping cannot capture the details of the nondeterministic OS and network delays, NTP may yield significant synchronization errors up to hundreds of milliseconds (ms) in a highly asymmetric network. To solve this issue, PTP uses hardware-level timestamping provided by PTP-compatible Ethernet cards at the end hosts and all the switches on the network path to achieve microsecond (μ s) accuracy. However, the need of the special hardware inevitably negates its universality and restricts PTP's adoption to time-critical local-area networks only, e.g., those found in industrial systems.

In wireless networks, due to the more uncertain communication delays caused by media access control (MAC), NTP performs worse. Thus, similar to PTP, most existing clock synchronization approaches for wireless sensor networks (WSNs) (e.g., Reference-Broadcast Synchronization (RBS) (Elson, Girod, & Estrin, 2002), Timing-sync Protocol for Sensor Networks (TPSN) (Ganeriwala, Kumar, & Srivastava, 2003), and Flooding Time Synchronization Protocol (FTSP) (Maróti, Kusy,

Simon, & Lédeczi, 2004)) have resorted to MAC-layer timestamping provided by the nodes' radio chips to pursue synchronization accuracy. The study (Casas, Gracia, Marco, & Falcó, 2005) uses MAC-layer access of Bluetooth called the Host Controller Interface (HCI) to synchronize Bluetooth devices. The classic Bluetooth devices can use the Bluetooth Clock Synchronization Protocol (CSP) included in the Bluetooth Health Device Profile (HDP). However, Bluetooth Low Energy (BLE) does not support HDP and CSP. The need of the MAC-layer timestamping or specific radio links present a barrier for the wide adoption of these approaches to the broader IoT domain, where the IoT platforms use diverse radios and OSes, and in general they do not provide an interface for the MAC-layer timestamping.

In the first part of this thesis, we aim at developing a new clock synchronization approach for wearables that establishes a desirable accuracy-universality trade-off point between the two extremes represented by NTP and PTP to well address the momentum of IoT platform heterogenization. In particular, we stem from the sensor nature of wearables to explore ambient signals that can assist clock synchronization. Recent studies exploited external periodic signals such as powerline radiation (Rowe, Gupta, & Rajkumar, 2009) and Radio Data System (RDS) (L. Li et al., 2011) to calibrate the clocks of WSN nodes. However, these approaches need non-trivial extra hardware to capture the external signals. Moreover, they focus on *clock calibration* that ensures different clocks advance at the same speed, rather than synchronizing the clocks to have the same value. But they inspire us to inquire (i) the existence of a periodic and synchronous signal that can be sensed by different wearables without adding non-trivial hardware to preserve universality and (ii) how to exploit the signal to synchronize the wearables without using hardware-level packet timestamping.

For the first inquiry, we conduct extensive measurements to explore such signals. Our measurements based on Adafruit's Flora, an Arduino-based wearable platform, show that by simply sampling an onboard analog-to-digital converter (ADC), a Flora can capture powerline electromagnetic radiation that oscillates at the power grid frequency (e.g., 50 Hz). When the Flora's ADC has a physical contact with the wearer's skin, the sampled induced skin electric potential (iSEP) is a significantly amplified version of the powerline radiation, because the human body acts as an effective antenna. Although the iSEP's amplitude is dynamic due to the human body movements, its frequency is highly stable. Moreover, the iSEPs on the same

wearer and even different wearers in a same indoor environment exhibit desirable synchrony. The time displacement between the iSEPs at different positions of a still wearer is generally less than 1 ms. These results suggest that iSEP is a promising basis for synchronizing the wearables.

For the second inquiry, we integrate the periodic and synchronous iSEP signal into the universal principle of NTP to deal with the major source of NTP’s error, i.e., asymmetric communication delays. In the original NTP, the problem of estimating the offset between the slave’s and master’s clocks is a *real-domain* underdetermined problem that has infinitely many solutions. NTP chooses a solution by assuming symmetric communication delays, which does not hold in many scenarios, however. Assisted with the periodic iSEP, the problem reduces to an *integer-domain* underdetermined problem that has a finite number of solutions. However, it is challenging to infer which solution is correct. The integer ambiguity can be resolved by jointly considering multiple synchronization rounds with dynamic and asymmetric communication delays. Thus, the clock offset between a pair of wearables can be estimated with ms accuracy due to the ms synchrony between their iSEP signals.

Based on the above two key results, we design a novel clock synchronization approach for wearables, which we call *TouchSync*. It runs at the application layer in that the needed iSEP sampling, the network message exchange and timestamping can be implemented using standard wearable OS calls. Thus, by introducing a rather simple skin contact, we can readily achieve ms synchronization accuracy across heterogeneous wearable platforms, without resorting to the hardware-level packet timestamping that is extremely difficult to be standardized. To simplify the adoption of TouchSync by application developers, we design and release a `touchsync.h` header file (Yan, Li, Tan, & Huang, 2017b) that implements TouchSync’s signal processing algorithms and the integer ambiguity solver. With this header, the implementations of TouchSync in Arduino and TinyOS need about 50 and 150 lines of code, respectively, which manage sensor sampling, synchronization message exchange, and application-layer timestamping only. All these tasks can be easily implemented by Arduino and TinyOS developers. We then conduct extensive experiments in various indoor environments to show the pervasive availability of the iSEP signals. On the same wearer, TouchSync mostly achieves sub-ms accuracy and the largest error is 2.9 ms. We conduct a TouchSync-over-Internet

proof-of-concept experiment that yields errors below 7 ms between two wearers 10 km apart.

The ADC-skin contact needed by TouchSync can be easily integrated into the wearable designs with near-zero cost. Our experiments show that, in the absence of the contact, TouchSync can still work with graceful performance degradation. iSEP is a new and valuable sensing modality for integration consideration, since it provides accurate timing and is indicative of other information about the wearer (e.g., body orientation, movements, and indoor location) as suggested by our measurements in Chapter 3.

1.2 Touch-to-Access Device Authentication Using iSEPs

The indoor environments are increasingly populated with smart objects. It is estimated that by 2022, a typical family home could contain more than 500 smart devices (Gartner, 2014). Managing the access with many objects, including accessing the information on them or granting them to access certain information, becomes challenging. Typing password is tedious and infeasible for the objects without a keyboard or touchscreen. Biometrics-based user authentication suffers various shortcomings. Fingerprint scanning requires a well positioned finger press. Moreover, due to cost factor, small objects unlikely have fingerprint scanners. Face recognition solutions require face positioning and are costly (Mayo, 2018). Voice recognition-based access can be disturbing in certain environments, e.g., an open-plan office with colleagues and a bedroom with sleeping buddies. Moreover, defining a separate voice passphrase for each smart object to avoid incorrect invoking may result in too many passphrases.

In the second part of this thesis, we aim to develop a low-cost and convenient *touch-to-access* scheme that can be easily implemented on smart objects found in indoor environments. Specifically, a simple touch on an object allows an authorized user to access the object. This scheme will not require non-trivial interferences for user interactions, e.g., touchscreen. Different from integrating user identification (e.g., fingerprint scanning) into the objects, we resort to a *device authentication* approach that offloads the user’s identity to a personal *wearable token* device (e.g.,

a smart watch or a bracelet) and uses the token to access a touched object that has been previously paired with the token. This touch-to-access device authentication approach can greatly improve the user's convenience and experience in interacting with the smart objects. For instance, in a home with multiple residents, when a user wearing his token turns on a TV set using a smart remote control, the control obtains the user identity from the token and instructs the TV set to list the user's favorite channels. The user can touch other smart objects to personalize them. For example, the user can touch a music player for the favorite music. Another example is that the user can switch on a light that automatically tunes to the user's favorite color temperature or hue.

If the user can protect the personal wearable token well, the touch-to-access device authentication can be used in more access-critical scenarios. For example, a touch on a smartphone or tablet unlocks the device's screen automatically, allows in-app purchases, or passes the parental controls. Beyond the above use scenarios for improved convenience in access control, the touch-to-access scheme can enhance the security of various systems. For instance, it can be used with fingerprint scanning to form a two-factor authentication against fake fingerprints. A wireless reader can access a worn medical sensor only if the reader has a physical contact with the wearer's skin. The contact enforces the wearer's awareness regarding the access and prevents remote wireless attacks with stolen credentials ([Halperin et al., 2008](#)). Thus, the touch-to-access scheme is more secure than the existing hardware token approaches such as Duo ([Duo, 2018](#)).

The essence of the touch-to-access scheme is the detection of whether the wearable token and the smart object in question have physical contact with the same user's body. Existing studies tackle this same-body contact detection problem by intra-body communication (IBC) ([Zimmerman, 1995](#); [Matsushita, Tajima, Ayatsuka, & Rekimoto, 2000](#); [Park et al., 2006](#); [Baldus, Corroy, Fazzi, Klabunde, & Schenk, 2009](#); [Vu et al., 2012](#); [Holz & Knaust, 2015](#); [Roeschlin, Martinovic, & Rasmussen, 2018](#)) and physiological sensing such as electrocardiography (ECG) ([Poon, Zhang, & Bao, 2006](#); [F. Xu, Qin, Tan, Wang, & Li, 2011](#); [Venkatasubramanian, Banerjee, & Gupta, 2010](#); [Hu et al., 2013](#); [Rostami, Juels, & Koushanfar, 2013](#)), photoplethysmogram (PPG) ([Poon et al., 2006](#); [Venkatasubramanian et al., 2010](#); [Hu et al., 2013](#)), and electromyogram (EMG) ([L. Yang, Wang, & Zhang, 2016](#)). IBC requires either non-trivial customized transceivers ([Zimmerman, 1995](#); [Matsushita](#)

et al., 2000; Park et al., 2006; Baldus et al., 2009; Roeschlin et al., 2018) or a touchscreen as the receiver (Vu et al., 2012; Holz & Knaust, 2015), resulting in increased cost or reduced applicable scope. The physiological sensing approaches are based on a *body-area property*, i.e., the physiological signals captured from the same human body have similar values or features, whereas those collected from different human bodies are distinct. However, the physiological sensors are often bulky due to the required physical distances among a sensor’s electrodes (Barill, 2003; L. Yang et al., 2016). Furthermore, they often need careful placement and may perform poorly in daily life settings (Chang, Hu, Anderson, Fu, & Huang, 2012).

Different from IBC and physiological sensing, in this chapter, we investigate the feasibility and effectiveness of using *induced skin electric potential* (iSEP) due to the *body antenna effect* for touch-to-access device authentication. As a non-physiological phenomenon, the body antenna effect refers to the alteration of the intensity of the mains hum captured by an ADC when the ADC has a physical contact with a human body. The mains hum induced by the building’s electrical cabling is ubiquitous. In addition, ADC is a basic electronic component that is widely available on microcontrollers. Recent studies have exploited the body antenna effect for key stroke detection (Elfekey & Bastawrous, 2013), touch sensing (Cohn, Morris, Patel, & Tan, 2011), motion detection (Cohn, Gupta, et al., 2012), gesture recognition (Cohn, Morris, Patel, & Tan, 2012). These studies leverage several characteristics of iSEP, such as signal intensity alteration (Elfekey & Bastawrous, 2013), or feed iSEP signals to machine learning algorithms for motion and gesture recognition (Cohn et al., 2011; Cohn, Gupta, et al., 2012; Cohn, Morris, et al., 2012). To use iSEP for device authentication, its body-area property and the underlying physical mechanism need to be well understood. To the best of our knowledge, these issues have not been studied. This thesis investigates in detail the physical mechanism of the iSEP’s generation and its body-area property. The iSEP measurement by an ADC is the difference between the electric potentials of the ADC pin and the ground¹ of the sensor, respectively. From electrostatics, a human body, which can be viewed as an uncharged conductor, will alter its nearby electric field (EF) emitted from the electrical cabling of the building due to electrostatic induction. As a result, the iSEP measurement by a sensor will be affected by the presence of a nearby human body. In particular, we make the following two

¹Throughout this chapter, “ground” refers to the floating ground of a device.

hypotheses based on the above understanding. First, the iSEP signals measured by two sensors that are on the same human body and close to each other will be similar. This is because 1) the two sensors' ADC pins will have the same potential due to their connections to the equipotential human body, and 2) their grounds will most likely have similar potentials as they are close to each other in the EF. Second, the iSEP signals collected from different human bodies will be different. This is because different human bodies will most likely have different potentials and thus affect nearby EFs differently since they build up different surface charge distributions in the electrostatic induction.

Our extensive measurement results are consistent with the above two hypotheses. Based on the results, we design a prototype system called *TouchAuth* that performs touch-to-access device authentication based on iSEP signals. We implement the same-body contact detection algorithm based on two similarity metrics, i.e., absolute Pearson correlation coefficient (APCC) and root mean square error (RMSE). Extensive experiments show that the APCC-based TouchAuth achieves true acceptance rates of 94.2% and 98.9% subject to a false acceptance rate (FAR) upper bound of 2% when one and five seconds of iSEP signal is recorded, respectively. In contrast, ECG/PPG approaches (Poon et al., 2006; Venkatasubramanian et al., 2010; Hu et al., 2013) need to record the signal(s) for tens of seconds to achieve comparable detection accuracy (Section 4.1). Our experiments show that various possible interfering sources including appliances' electromagnetic emanations and noise injections into power networks do not affect TouchAuth.

In summary, TouchAuth is a low-cost, lightweight, and convenient approach for the authorized users to access smart objects in indoor environments. To implement TouchAuth, the smart object's and the wearable token's microcontroller ADCs are to be wired to their conductive exteriors. Compared with the near-field communication (NFC) approach that enforces a proximity requirement on device authentication, the touch requirement of TouchAuth is more intuitive and clearer. Moreover, compared with the ADCs that are widely available on microcontrollers, the NFC chips are more costly and need to be integrated into the smart objects to read the wearable tags.

1.3 Summay of Contributions

The contributions of this thesis are summarized as follows:

- It explains the generation mechanism of iSEP.
- It extensively measures iSEP signals to investigate various properties of iSEPs including periodicity, phase, and body-area similarity.
- It design a clock synchronization approach called TouchSync based on iSEP for wearables. TouchSync runs at the application layer that only need standard OS calls such as sampling sensors and transmitting/receiving network messages. By solving an integer ambiguity problem, TouchSync achieves synchronization errors of below 3 ms and 7 ms on the same wearer and between two wearers 10 km apart, respectively.
- It designs an iSEP-based device authentication approach called TouchAuth. TouchAuth uses iSEPs to detect whether two devices are in proximity on the same human body. TouchAuth achieves comparable authentication accuracy as existing physiological sensing approaches, but with much shorter sensing times. TouchAuth offers a low-cost, lightweight, and convenient way for the authorized users to access smart IoT objects found in indoor environments.

1.4 Thesis Structure

Chapter 2 presents the technical preliminaries of body antenna effect, the electric field (EF) of electrical cabling, and the interactions among EF, sensor, and human body. Chapter 3 presents TouchSync, an iSEP-based clock synchronization approach for wearables. Chapter 4 presents TouchAuth, a device authentication approach for wearables using iSEP. Chapter 5 concludes this thesis and discusses future work.

Chapter 2

Preliminaries on Body Antenna Effect

In this chapter, we present a set of measurement results to illustrate the body antenna effect. Moreover, we describe the underlying physical mechanism of body antenna effect, i.e., the electric field (EF) from the electric cabling, and the generation of skin electric potential induced by EF.

2.1 Body Antenna Effect

First, we illustrate the body antenna effect. The two curves in Fig. 2.1 are the measurement traces of a mote-class sensor placed at a fixed position, with an ADC pin floating in the air or pinched by a person, respectively. Without body contact, the sensor captures the mains hum with weak amplitude and a frequency of about 50 Hz (i.e., the nominal grid frequency in our region). With body contact, the signal has greater amplitude and exhibits more clearly the frequency of 50 Hz. The above result shows that the human body affects the reception of mains hum. Several recent studies ([Cohn, Morris, et al., 2012](#); [Elfekey & Bastawrous, 2013](#)) exploited this human body antenna effect for various applications. However, they do not provide an in-depth explanation of the effect. This chapter explains this effect in detail, which guides our experiments and the designs of TouchSync and TouchAuth.

FIGURE 2.1: The body antenna effect.

2.2 Electric Field (EF) From Electrical Cabling

A line of charge emits an EF, whereas a current through the line generates a magnetic field. Thus, a charged wire carrying alternating current (ac) generates both time-varying electric and magnetic fields. However, since the magnetic fields generated from the two close-lying current-carrying wires within a single power cable tend to cancel each other, the overall magnetic field around a power cable is normally very weak (Vistnes, 2001).¹ At the nominal frequency of the ac power grid, i.e., 50 or 60 Hz, the power cable's EF is an extremely low frequency (ELF) radiation with a wavelength of thousands of miles. At such a wavelength scale, we do not need to consider the magnetic field excited by the time-varying EF. Thus, EF is the main emanation from a power cable.

Modern buildings often have complex electrical cabling. For example, permanent power cables run above ceilings, below floors, and on walls. There are also power extension cords installed by residents. As the EF from a cable is a vector field with intensity attenuating with the distance from the cable, the combined EF caused by all the cables in a building is a vector field with a complex intensity distribution over the space. In normal homes with 220 V power supply, the intensity of the combined EF is often between 3 V/m and 30 V/m (Vistnes, 2001). The EF is the superposition of the EFs emitted by surrounding electrified power cables and appliances. Due to the spatial distribution of the power cables and appliances, the gradients of the EF at different locations are generally different.

¹The low-intensity net magnetic field due to a small mismatch between the two wires' currents, which is caused by the vagabond currents effect, is sensible using special devices such as hall effect sensors and tank circuits tuned to the power grid frequency.

FIGURE 2.2: Mains hum measured by a sensor (without human body contact) placed close to a power cord supporting a 2 kW heater.

Note that if an appliance is powered off, the intensity of the EF from the power cable supporting the appliance remains unchanged. This is due to the fact that most switches only break the connection in one wire, while the wires still have the same service voltage as when the appliance is powered on. We conduct an experiment to verify this. Fig. 2.2a shows the mains hum measured by a sensor that has a conductor wire connected to an ADC pin to improve EF sensing and is placed close to a power cord supporting a 2 kW heater. From the figure, the operating status of the heater does not affect the measurements. Fig. 2.2b shows the sensor's measurements when the heater remains off and the power cord is connected to or disconnected from the wall outlet. We can see that the intensity of the sensor readings is weaker when the power cord is diselectrified. The remaining intensity is caused by the EFs from other electrified power cables in the building. The above results suggest that (i) the ambient field is an EF caused by the ac voltages, (ii) the ac current changes caused by appliances' operating status changes have little impact on the ambient field.

2.3 Interactions Among EF, Sensor, and Human Body

First, we discuss the situation without a human body. Our discussions below concern a time instant only. As a typical sensor's ADC has high input impedance

FIGURE 2.3: Impact of an uncharged conductor (e.g., human body) on an EF from a point charge. Arrows represent the directions of the EF; dotted curves represent equipotential lines.

(hundreds of $k\Omega$ up to a few $M\Omega$), the ADC pin and the ground of the sensor can be considered insulated for simplicity of discussion. The ADC and ground have different potentials in the building's ambient EF due to their physical distance. The potential difference is the measurement of the sensor. For instance, in an EF with an intensity up to 30 V/m , if the equivalent distance between the ADC pin and the ground is 1 cm , the measurement can be up to 0.3 V . This is consistent with our results in Figs. 2.1 and 2.2.

Now, we discuss the situation with a human body. A body can be viewed as a conductor due to its low impedance (a few $k\Omega$ (Reilly, 1998)). From electrostatics, an uncharged conductor in an EF builds up a surface charge distribution to reach an electrostatic equilibrium, where the EF inside the conductor is zero and the conductor's surface is an equipotential surface (Purcell & Morin, 2013). The surface charge distribution will generate an EF. As a result, the EF combining that from the original source (i.e., electrical cabling) and the electrostatically induced conductor (i.e., the human body) is different from the EF in the absence of the conductor. In other words, the human body affects its nearby EF. The change of field intensity results in the change of potential difference between the sensor's ADC and ground, i.e., the sensor's measurement.

We use an electrostatics example as shown in Fig. 2.3 to illustrate the human body's impact on ambient EF. When an uncharged conductor is in the field from a point charge, negative/positive surface charges will be built up. As a result, the EF intensity, which is characterized by the density of the equipotential lines, will change in the space close to the electrostatically induced conductor. For instance, in Fig. 2.3, the EF between the charge and the conductor is intensified. The reading of

a sensor in this area will increase if its ADC and ground are arranged in the direction of the field. In practice, the indoor EF will be much more complex than the one shown in Fig. 2.3. Nevertheless, the example provides a basic understanding of the body antenna effect. Noted that the iSEP sensor runs at 3.7 volts and captures the electric potentials from human skin passively, there is no current go through the user.

Chapter 3

Wearables Clock Synchronization Using iSEPs

In this chapter, we present TouchSync, an application-layer clock synchronization approach for wearables¹. Section 3.1 reviews related work. Section 3.2 introduces background. Section 3.3 presents a measurement study. Section 3.4 designs TouchSync. Section 3.5 extends the design to address the situations where the iSEPs are too weak. Section 3.6 and Section 3.7 implement and evaluate TouchSync, respectively.

3.1 Related Work

Highly stable time sources are ill-suited for wearables. Chip-scale atomic clocks are too expensive (\$1,500 per unit (Wire, 2011)). GPS receivers are power-consuming and do not work in indoor environments. The studies (Ferrari, Zimmerling, Thiele, & Saukh, 2011) and (Lenzen, Sommer, & Wattenhofer, 2009) propose synchronization protocols for IEEE 802.15.4 networks. The studies (Terraneo, Leva, Seva, Maggio, & Papadopoulos, 2015) and (Lim, Maag, & Thiele, 2016) improve the synchronization accuracy by compensating the propagation delays. However, these studies (Ferrari et al., 2011; Lenzen et al., 2009; Terraneo et al., 2015; Lim et al.,

¹This chapter is partially published on (Yan, Li, Tan, & Huang, 2017a) and (Yan, Tan, Li, & Huang, 2019).

2016) need the MAC-layer access, which is usually not provided by the heterogeneous wearable platforms. Recent studies exploited external signals available in indoor environments to synchronize or calibrate the clocks of distributed nodes. In (Chen, Wang, Chang, & Terzis, 2011), an AM radio receiver is designed to decode the global time information broadcast by timekeeping radio stations. In (L. Li et al., 2011), a mote peripheral is designed to capture the periodic RDS signals of FM radios for clock calibration. In (Rowe et al., 2009), a mote peripheral called *syntonistor* can receive the periodic electromagnetic radiation from powerlines to calibrate wireless sensors' clocks, where some clock synchronization approach is still needed for the initial synchronization. In (Viswanathan, Tan, & Yau, 2016; Rabadi, Tan, Yau, & Viswanathan, 2017), voltage sensors plugged in to wall power outlets are used to secure clock synchronization against malicious network delays. In particular, the voltage cycle length fluctuations are exploited to implement a data-based clock synchronization approach (Viswanathan et al., 2016). In (Y. Li, Tan, & Yau, 2017), such fluctuations extracted from the powerline radiation are used as *natural timestamps*. However, the error of the natural timestamps can be up to hundreds of ms. Moreover, the clock synchronization based on the natural timestamps needs to transmit a considerable amount of cycle length data and a compute-intensive matching process to decode the fluctuations to time information (Y. Li et al., 2017). Thus, the natural timestamping approach is ill-suited for tight clock synchronization among resource-constrained wearables. In (Lazik, Rajagopal, Sinopoli, & Rowe, 2015), a smartphone captures ultrasonic beacons from pre-deployed synchronized beacon nodes to synchronize its own clock. All the above approaches (Chen et al., 2011; L. Li et al., 2011; Rowe et al., 2009; Viswanathan et al., 2016; Rabadi et al., 2017; Y. Li et al., 2017; Lazik et al., 2015) need non-trivial customized hardware and infrastructures, which reduce their universality.

Two recent studies leverage built-in sensing modalities to capture external periodic signals for clock calibration. In (Hao, Zhou, Xing, Mutka, & Chen, 2011), a 802.15.4 radio is used to capture the Wi-Fi beacons to calibrate motes' clocks. Although this approach requires no peripherals, it uses the received signal strength indication (RSSI) register of the radio chip, which makes it hardware specific and nonuniversal for IoT platforms that use diverse radios. In (Z. Li et al., 2012), motes use light sensors to capture the fluorescent light that flickers at a frequency twice of the power grid frequency to calibrate their clocks. Although light sensors are widely available, the required fluorescent lighting may not be available in natural lighting

environment. In contrast, the powerline radiation that induces the iSEP signal used by our approach pervades civil infrastructures.

The studies (L. Li et al., 2011; Rowe et al., 2009; Hao et al., 2011; Z. Li et al., 2012) mentioned above, including the two (Rowe et al., 2009; Z. Li et al., 2012) that are power grid related, focus on *clock calibration* that involves no message exchanges among nodes. Though continuous clock calibration maintains the nodes synchronized once they are initially synchronized, the initial synchronization and the resynchronizations needed upon clock calibration faults are not addressed in these studies. Thus, these studies and ours are complementary, in that the principle of TouchSync can be used for the initial synchronization of the systems adopting these clock calibration approaches (L. Li et al., 2011; Rowe et al., 2009; Hao et al., 2011; Z. Li et al., 2012).

In (He, Duan, Cheng, Shi, & Cai, 2014), a distributed clock synchronization approach based on consensus is proposed. Note that TouchSync addresses the clock synchronization between two nodes only. The noise estimation algorithm in (He et al., 2014) can be integrated with TouchSync to improve performance. TouchSync can be applied to both centralized and distributed clock synchronization systems.

3.2 Background

3.2.1 NTP Principle and Packet Timestamping

Many clock synchronization approaches adopt the principle of NTP, which is illustrated in Fig. 3.1a. A *synchronization session* consists of the transmissions of a request packet and a reply packet. The t_1 and t_4 are the slave's clock values when the request and reply packets are transmitted and received by the slave node, respectively. The t_2 and t_3 are the master's clock values when the request and reply packets are received and transmitted by the master node, respectively. Thus, the round-trip time (RTT) is $RTT = (t_4 - t_1) - (t_3 - t_2)$. Based on a *symmetric link assumption* that assumes identical times for transmitting the two packets, the offset between the slave's and the master's clocks, denoted by δ_{NTP} , is estimated as $\delta_{NTP} = t_4 - (t_3 + \frac{RTT}{2})$. Then, this offset is used to adjust the slave's clock to achieve clock synchronization. Under the above principle, non-identical times for

FIGURE 3.1: NTP principle and packet timestamping.

transmitting the two packets will result in an error in estimating the clock offset. The estimation error is half of the difference between the times for transmitting the two packets.

We use Fig. 3.1b and the terminology in (Maróti et al., 2004) to explain how the timestamps (e.g., t_3 and t_4) are obtained in NTP and existing WSN clock synchronization approaches. The *send time* and the *receive time* are the times used by the OS to pass a packet between the synchronization program and the MAC layer at the sender and receiver, respectively. They depend on OS overhead. The *access time* is the time for the sender's MAC layer to wait for a prescribed time slot in time-division multiple access (TDMA) or a clear channel in carrier-sense multiple access with collision avoidance (CSMA/CA). It often bears the highest uncertainty and can be up to 500 ms (Maróti et al., 2004). The transmission (Tx) and reception (Rx) times are the physical layer processing delays at the sender and receiver, respectively. The *propagation time* equals the distance between the two nodes divided by the speed of light, which is generally below $1 \mu\text{s}$.

As illustrated in Fig. 3.1b, NTP timestamps the packet when the packet is passed to or received from the OS. Thus, the packet transmission time used by NTP is subjected to the uncertain OS overhead and MAC. Therefore, as measured in Section 3.2.2, NTP over a Bluetooth connection can yield nearly 200 ms clock offset estimation errors. To remove these uncertainties, FTSP uses MAC-layer access to obtain the times when the beginning of the packet is transmitted/received by the radio chip. As the propagation time is generally below $1 \mu\text{s}$, FTSP simply estimates the clock offset as the difference between the two hardware-level timestamps. Thus, the two-way scheme in Fig. 3.1a becomes non-essential for FTSP.

FIGURE 3.2: Performance of NTP over a BLE connection.

3.2.2 Performance of NTP over BLE Connection

As our objective is to devise a new clock synchronization approach that uses application-layer timestamping as NTP does, this section measures the performance of NTP to provide a baseline. We implement the NTP principle described in Section 3.2.1 on Flora (Adafruit, 2018), a wearable platform. Our setup includes a Flora node and a Raspberry Pi (RPi) 3 Model B+ single-board computer (Raspberry Pi 3 Model B, 2017) that perform the NTP slave and master, respectively. They are connected via Bluetooth Low Energy (BLE). More details of the Flora setup can be found in Section 3.3.1 of the chapter. Fig. 3.2a shows the distributions of the one-way application-layer communication delays over 110 NTP sessions. The slave-to-master delays are mostly within $[40, 50]$ ms, with a median of 42 ms and a maximum of 376 ms (not shown in the figure). As specified by the BLE standard, the master device pulls data from a slave periodically. The period, called *connection interval*, is determined by the master. The slave needs to wait for a pull request to transmit a packet to the master. In the RPi’s BLE driver (BlueZ), the connection interval is set to 67.5 ms by default. As the arrival time of a packet from the slave’s OS is uniformly distributed over the connection interval, the expected access time is $67.5/2 = 33.75$ ms. This is consistent with our measured median delay of 42 ms, which is about 8 ms longer because of other delays (e.g., send and receive times). The exceptionally long delays (e.g., 376 ms) observed in our measurements could be caused by transient wireless interference and OS delays. For the master-to-slave link, the delays are mostly within $[0, 10]$ ms, with a median of 8 ms and a maximum of 153 ms. A BLE slave can skip a number

FIGURE 3.3: Flora.

FIGURE 3.4: TouchSync Prototypes.

of pull requests, which is specified by the *slave latency* parameter, and sleep to save energy. Under BlueZ’s default setting of zero for slave latency, the slave keeps awake and listening, yielding short master-to-slave delays.

The asymmetric slave-to-master and master-to-slave delays cause significant errors in the NTP’s clock offset estimation. At the end of each synchronization session, the RPi computes this error as $\delta_{NTP} - \delta_{GT}$, where δ_{NTP} and δ_{GT} are NTP’s estimate and the ground-truth offset, respectively. Fig. 3.2b shows the distribution of the errors. We observe that 28% of the errors are larger than 25 ms. The largest error in the 110 NTP sessions is 183 ms. Such an error profile does not well meet the ms accuracy requirements of many applications (Dinescu, Mazza, Kujanski, Gaza, & Sagan, 2015; Lorincz et al., 2009; Mokaya et al., 2016). Though it is possible to calibrate the average error to zero by using prior information (e.g., the settings of the connection interval and slave latency), the calibration is tedious, nonuniversal, and incapable of reducing noise variance.

3.3 Measurement Study

In this section, we conduct measurements to gain insights for guiding the design of TouchSync.

3.3.1 Measurement Setup

Our measurement study uses two Flora nodes and a RPi 3 Model B single-board computer. The Flora is an Arduino-based wearable platform. Each Flora node, as

shown in Fig. 3.3, consists of a main board with an ATmega32u4 MCU (8 MHz, 2.5 KB RAM), a BLE 4.1 module, and a 150mAh lithium-ion polymer battery. The RPi has a built-in BLE 4.1 module and runs Ubuntu MATE 16.04 with BlueZ 5.37 as the BLE driver. Each node is powered by an independent battery and has no electrical connection to any ground or grounded appliance. We use Adafruit’s nRF51 Arduino library and BluefruitLE Python library on the Floras and the RPi, respectively, to send and receive data over BLE in the UART mode. The Floras and RPi operate as BLE peripheral (slave) and central (master), respectively. To obtain the ground truth clocks in each experiment, we synchronize the Floras with the RPi as follows. At the beginning of the experiment, we wire a general-purpose input/output (GPIO) pin of the RPi with a digital input pin of each Flora. Then, the RPi issues a rising edge through the GPIO pin and records its clock value t_{master} . Upon detecting the rising edge, a Flora records its clock value t_{slave} and sends it to the RPi. The RPi computes the ground-truth offset between the Flora’s and the RPi’s clocks as $\delta_{GT} = t_{slave} - t_{master}$. Then, we remove the wiring and conduct experiments.

3.3.2 iSEP Sensing under Various Settings

In this set of measurements, we explore i) whether a human body is an effective antenna for receiving the powerline radiation and ii) whether the iSEP signals induced by the radiation on the same wearer or different wearers are synchronous. The Flora’s microcontroller (MCU) has a 10-bit ADC that supports a sampling rate of up to 15 kHz. To facilitate experiments, we have made two Flora-based prototypes as shown in Fig. 3.4. We place the Flora into a 3D-printed insulating wristband and use a thin stainless steel conductive thread to create a connection between Flora’s ADC pin and the wearer’s skin. The Flora samples the ADC at 333 Hz continuously for two minutes and streams the timestamped raw data to the RPi for offline analysis. The sampling rate of 333 Hz is sufficient to capture the powerline radiation or iSEP with a frequency of 50 Hz in our region. All samples are normalized using the reference voltage of the ADC. As only the ADC pin is connected to the researcher, the grounding of the Flora may affect the sampling result. To understand the impact of grounding, we conduct two sets of comparative experiments, where the two Floras have shared and independent grounds, respectively. During the experiment, neither Flora nor human body has contact

FIGURE 3.5: No human body contact.

FIGURE 3.6: iSEPs on a same wearer (shared ground).

with grounded appliance. In each experiment set, there are two scenarios: still and moving. For the moving scenario, the researcher keeps changing the body orientation, movement, and location. The experiments are conducted in a computer science laboratory with various appliances such as lights, computers, and printers.

3.3.2.1 Shared ground

We wire the ground pins of the two Floras, such that they have a shared ground. We conduct three tests.

First, Fig. 3.5a shows the signals captured by the two Floras when they have no physical contact with any human body. The signals have small fluctuations with a normalized peak-to-peak amplitude of 0.024. The signals fluctuate at a frequency of 50 Hz. This suggests that the Floras can pick up the powerline radiation. However, the signals are weak.

Second, a researcher touches the ADC pins of the two Floras with his two hands, respectively. Figs. 3.6a and 3.6b show the signals captured by the two nodes during

FIGURE 3.7: iSEPs on different wearers (shared ground).

FIGURE 3.8: Absolute time displacement $|\epsilon|$ between the EMR signals captured by the two Floras in various scenarios. Error bar represents (5%, 95%) confidence interval in one minute of data.

the same time duration, when the researcher stands still and walks, respectively. Under the two scenarios (still and moving), the two nodes yield salient and almost identical signals. The peak-to-peak amplitudes in the two figures are around 0.4 and 0.8, which are 17 and 33 times larger than that of the signal shown in Fig. 3.5a. This suggests that the human body can effectively receive the powerline radiation.

Third, two researchers touch the ADC pins of the two Floras separately. Figs. 3.7a and 3.7b show the signals captured by the two nodes when the two researchers stand still and walk, respectively. The two nodes yield salient signals with different amplitudes. We note that several factors may affect the reception of powerline radiation, e.g., human body size, position and facing of the body in the electromagnetic field generated by the powerlines.

We evaluate the synchrony between the signals captured by the two Floras shown in Figs. 3.6 and 3.7. We condition the signals by first applying a band-pass filter (BPF) to remove the direct current (DC) component that may fluctuate as seen in Fig. 3.6b and then detect the zero crossings (ZCs) of the filtered signals. More

FIGURE 3.9: iSEPs on a same wearer (independent grounds).

details of the BPF and ZC detection are presented in Section 3.4.2. We use the *time displacement* between the two signals' ZCs as the metric to evaluate their synchrony. Specifically, the time displacement, denoted by ϵ , is given by $\epsilon = t_A^{ZC} - t_B^{ZC}$, where t_A^{ZC} and t_B^{ZC} represent the ground-truth times of Node A's ZC and the corresponding ZC at Node B, respectively. Fig. 3.8a shows the error bars for $|\epsilon|$, which correspond to the scenarios in Figs. 3.6a, 3.6b, 3.7a, and 3.7b, respectively. In Fig. 3.8a, “same” and “diff” mean the same wearer and different wearers, respectively; “still” and “move” mean standing still and walking, respectively. On the same wearer, the iSEPs captured by the two Floras are highly synchronous, with an average $|\epsilon|$ of $0.9 \mu\text{s}$. On different wearers, the $|\epsilon|$ increases to about 1 ms. When the two wearers move, the average $|\epsilon|$ is 0.35 ms smaller than that when they stand still. Note that as shown shortly in Fig. 3.11b, the body orientation affects the time displacement. In Fig. 3.8a, the two wearers stand still facing certain orientations. The time displacement under such body orientation setting is consistently large. Thus, the average time displacement when they move is smaller. However, consistent with intuition, the human body movements increase the variance of $|\epsilon|$, since they create more signal dynamics as seen in Fig. 3.6b.

3.3.2.2 Independent grounds

Then, we remove the connection between the two Floras' ground pins, such that they have independent grounds. This setting is consistent with real scenarios, where the wearables are generally not wired. We conduct three tests.

Fig. 3.5b shows the two Floras' signals when they have no physical contact with any human body. The signals have small oscillations with a frequency of 50 Hz.

FIGURE 3.10: iSEPs on different wearers (independent grounds).

FIGURE 3.11: Normalized range and time displacement $|\epsilon|$ of iSEPs under different skin conditions and body orientations.

Figs. 3.9a and 3.9b show the signals of the two Floras worn on two wrists of a researcher when he stands still and walks, respectively. Compared with the results in Fig. 3.6a based on a shared ground, the two signals in Fig. 3.9a have an offset in their values. This offset is the difference between the electric potentials at the two Floras' grounds. Fig. 3.9b shows the signals over two seconds that contain about 100 iSEP cycles to better illustrate the changing signal envelopes over time due to the human body movements. Compared with Fig. 3.6b, the two signals in Fig. 3.9b have different signal envelopes. This is because the electric potentials at the two Floras' grounds, which are induced by the powerline radiation, are not fully correlated in the presence of human body movements.

Figs. 3.10a and 3.10b show the signals of the two Floras worn by two researchers each when they stand still and walk, respectively. Salient EMR signals can be observed. Moreover, the human movements cause significant fluctuations of DC lines and the signal amplitudes, as seen in Fig. 3.10b.

We also evaluate the synchrony between the two Floras' signals. Fig. 3.8b shows the time displacement's error bars that correspond to the scenarios in Figs. 3.9a,

3.9b, 3.10a, and 3.10b. The average $|\epsilon|$ is below 2 ms. Compared with the results in Fig. 3.8a that are based on a shared ground, the time displacements increase. This is because of the additional uncertainty introduced by the independent floating grounds of the two Floras. Nevertheless, on the same wearer, the average $|\epsilon|$ is about 0.5 ms only. The body movements increase the 95%-percentile of $|\epsilon|$ to 1.5 ms. In Section 3.4.2, we use a phase-locked loop to reduce the variations of ϵ .

We further evaluate the impact of skin condition and body orientation on the iSEPs when the two Floras are worn by two hands of the same wearer. Fig. 3.11a shows the normalized iSEP signal range and their time displacement on the same still person, when the skin is dry or wet. The iSEP signal is amplified when the sensor has direct contact with the human body. The skin moisture condition does not affect the amplitude and synchrony ($|\epsilon|$) of iSEP much. We have also tested iSEP sensing using two conductive threads for a Flora. Results show that the signal reception is not affected. However, using multiple conductive threads reduces the chance that the wearable loses direct skin contact. Fig. 3.11b shows the time displacements when the wearer is in different orientations. We can see that the body orientation affects the time displacement. However, the average time displacement $|\epsilon|$ in all tested orientations is below 4ms.

3.3.2.3 Summary

From the above measurements, we obtain the following three key observations. First, the human body can act as an antenna that effectively improves the powerline radiation reception. Second, during the human body movements, the iSEP amplitude changes. However, the synchrony between the two iSEP signals captured by the two nodes on the same wearer or different wearers is still acceptably preserved. In Section 3.4.2, we condition the iSEP signals to improve the synchrony. Third, the floating ground of a node introduces additional uncertainty, because the powerline radiation can generate a varying electric potential at the ground pin. However, the floating ground does not substantially degrade the synchrony between the two nodes' iSEP signals. All experiments in the rest of this chapter are conducted under the floating ground setting. The above three observations suggest that the iSEP induced by powerline radiation is a good periodic signal that can be exploited for synchronizing wearables.

FIGURE 3.12: A synchronization process of TouchSync.

3.4 Design of TouchSync

This section presents the design of TouchSync. Section 3.4.1 overviews the workflow. Section 3.4.2 presents the signal processing algorithms to generate stable, periodic, and synchronous impulses trains (i.e., Dirac combs) from the iSEP signals. Section 3.4.3 presents a synchronization protocol assisted with the Dirac combs. Section 3.4.4 solves the integer ambiguity problem to complete synchronization.

3.4.1 TouchSync Workflow

TouchSync synchronizes the clock of a slave to that of a master. This chapter focuses on the synchronization between a slave-master pair, which is the basis for synchronizing a network of nodes. A *synchronization process*, as illustrated in Fig. 3.12, is performed periodically or in an on-demand fashion. For instance, the wearer(s) may push some buttons on two wearables to start a synchronization process. The period of the synchronization can be determined by the needed clock accuracy and the clock drift rate. During the synchronization process, both the slave and the master continuously sample the iSEP signals and store the time-stamped samples into their local buffers. Note that the iSEP signal strength may become too weak to support accurate clock synchronization in certain scenarios. The TouchSync node will switch to a mode that generates an internal periodic signal to drive the clock synchronization, which will be presented in Section 3.5. At the beginning of the synchronization process, the slave node sends a message to the master to signal the start of the sensor sampling. As shown in Fig. 3.12, a

FIGURE 3.13: iSEP signal processing pipeline.

synchronization process has multiple *synchronization sessions*. In each session, the slave and the master exchange three messages: **request**, **reply1**, and **reply2**. The **request** and **reply1** are used to measure the communication delays. After transmitting the **reply1**, the master retrieves a segment of iSEP signal from its buffer to process and transmits the processing results using the **reply2** to the slave. Upon receiving the **reply1**, the slave retrieves a segment of iSEP signal from its buffer to process. Upon receiving the **reply2**, the slave tries to solve the integer ambiguity problem to estimate the offset between the slave's and the master's clocks. If the ambiguity cannot be solved, another synchronization session is initiated; otherwise, the two nodes stop sampling iSEPs and the slave uses the estimated offset to adjust its clock and complete the synchronization process.

3.4.2 iSEP Signal Processing

This section presents TouchSync's signal processing pipeline as illustrated in Fig. 3.12 that aims to generate a highly stable, periodic, and synchronous Dirac comb from an iSEP signal with fluctuating DC component and jitters as shown in Fig. 3.9 and Fig. 3.10. We apply a signal processing pipeline illustrated in Fig. 3.13. It consists of three steps that are described in what follows.

3.4.2.1 Band-pass filter (BPF) or mean removal filter (MRF)

We apply a 5th-order/6-tap infinite impulse response (IIR) BPF with steep boundaries of a (45 Hz, 55 Hz) passband to remove the fluctuating DC component and high-frequency noises of the iSEP signal. For too resource-limited wearables, a MRF that subtracts the running average from the original signal can be used instead of the BPF for much lower compute and storage complexities. Its effect is similar to high-pass filtering.

3.4.2.2 Zero crossing detector (ZCD)

It detects the ZCs, i.e., the time instants when the filtered iSEP signal changes from negative to positive. It computes a linear interpolation point between the negative and the consequent positive iSEP samples as the ZC to mitigate the impact of low time resolution due to a low iSEP sampling rate.

3.4.2.3 Phase-locked loop (PLL)

We apply a software PLL to deal with the ZC jitters and miss detection caused by significant dynamics of the iSEP signal. The PLL generates an impulse train using a loop and uses an active proportional integral (PI) controller to tune the interval between two consecutive impulses according to the time differences between the past impulses and the input ZCs. The controller skips the time differences larger than 25 ms to deal with ZC miss detection.

3.4.3 NTP Assisted with Dirac Combs

TouchSync uses the synchronous Dirac combs at the slave and the master to achieve clock synchronization through multiple synchronization sessions. This section presents the protocol for a single synchronization session.

3.4.3.1 Protocol for a synchronization session

A synchronization session of TouchSync is illustrated in Fig. 3.14. We explain it from the following two aspects.

Message exchange and timestamping: The session consists of the transmissions of three messages: `request`, `reply1`, and `reply2`. The `request` and `reply1` messages are similar to the two UDP packets used by NTP. Their transmission and reception timestamps, i.e., t_1 , t_2 , t_3 , and t_4 , are obtained upon the corresponding messages are passed/received to/from the OS. The master transmits the auxiliary `reply2` message to convey the results of its signal processing, which is detailed below.

FIGURE 3.14: A synchronization session of TouchSync. The vertical arrows represent the Dirac comb impulses generated from the iSEP.

Signal processing and clock offset estimation: After the master has transmitted the **reply1** message, the master (i) retrieves from its signal buffer an iSEP signal segment that covers the time period from t_2 to t_3 with some safeguard ranges before t_2 and after t_3 , (ii) feeds the signal processing pipeline in Section 3.4.2 with the retrieved iSEP signal segment to produce a Dirac comb as illustrated in Fig. 3.14, and (iii) identifies the last impulses (LIs) in its Dirac comb that are right before the time instants t_2 and t_3 , respectively. The LIs are illustrated by thick red arrows in Fig. 3.14. Then, the master computes the elapsed times from t_2 's LI to t_2 and t_3 's LI to t_3 , which are denoted by ϕ_2 and ϕ_3 , respectively. The ϕ_2 and ϕ_3 are the *phases* of the t_2 and t_3 with respect to the Dirac comb. After that, the master transmits the **reply2** message that contains t_2 , t_3 , ϕ_2 , and ϕ_3 to the slave. After receiving the **reply1** message, the slave retrieves an iSEP signal segment that covers the time period from t_1 to t_4 with some safeguard ranges, executes the signal processing pipeline, identifies the LIs right before t_1 and t_4 , and computes the phases ϕ_1 and ϕ_4 , as illustrated in Fig. 3.14. After receiving the **reply2** message, based on $\{t_1, t_2, t_3, t_4\}$ and $\{\phi_1, \phi_2, \phi_3, \phi_4\}$, the slave uses the approach in Section 3.4.3.2 to analyze the offset between the slave's and master's clocks.

Now, we discuss several design considerations for the protocol described above. TouchSync uses the **request** and **reply1** messages to measure the clock offset, while the **reply2** is an auxiliary message to convey the timestamps t_2 , t_3 and the measurements ϕ_2 , ϕ_3 . With this auxiliary message, we can decouple the task of timestamping the reception of **request** and the transmission of **reply1** from the signal processing task of generating the Dirac comb and computing ϕ_2 and ϕ_3 . On

many platforms (e.g., Wear OS and watchOS), continuously sampled sensor data is passed to the application block by block. With the decoupling, the master can compute ϕ_2 and ϕ_3 after the `reply1` is transmitted and the needed iSEP data blocks become available.

3.4.3.2 Clock offset analysis

We now analyze the offset between the slave's and the master's clocks based on $\{t_1, t_2, t_3, t_4\}$ and $\{\phi_1, \phi_2, \phi_3, \phi_4\}$. Denote by T the period of the Dirac comb. In our region served by a 50 Hz grid, the nominal value for T is 20 ms. To capture the small deviation from the nominal value, T can be easily computed as the average interval between consecutive impulses of the Dirac comb. We define the *rounded phase differences* θ_q and θ_p (which correspond to the `request` and `reply1` messages, respectively) as

$$\theta_q = \begin{cases} \phi_2 - \phi_1, & \text{if } \phi_2 - \phi_1 \geq 0; \\ \phi_2 - \phi_1 + T, & \text{otherwise.} \end{cases} \quad (3.1)$$

$$\theta_p = \begin{cases} \phi_4 - \phi_3, & \text{if } \phi_4 - \phi_3 \geq 0; \\ \phi_4 - \phi_3 + T, & \text{otherwise.} \end{cases} \quad (3.2)$$

As ϕ_k is the elapsed time from t_k 's LI to t_k , we have $0 \leq \phi_k < T$, for $k \in [1, 4]$. From Eqs. (3.1) and (3.2), we can verify that $0 \leq \theta_q < T$ and $0 \leq \theta_p < T$. From our measurements in Section 3.2.2, the times for transmitting the `request` and `reply1` messages can be longer than T . Thus, we use i to denote the non-negative integer number of the Dirac comb's periods elapsed from the time of sending `request` to the time of receiving it at the master, and j to denote the non-negative integer number of the Dirac comb's periods elapsed from the time of sending `reply1` to the time of receiving it at the slave.

We denote τ_q and τ_p the actual times for transmitting the `request` and the `reply1` messages, respectively. Thus,

$$\tau_q = \theta_q + i \cdot T - \epsilon, \quad \tau_p = \theta_p + j \cdot T + \epsilon, \quad (3.3)$$

where ϵ is the time displacement between the slave's and master's Dirac combs. Here, we assume a constant ϵ to simplify the discussion. Therefore, the RTT

computed by $\text{RTT} = (t_4 - t_1) - (t_3 - t_2)$ must satisfy

$$\text{RTT} = \tau_q + \tau_p = \theta_q + \theta_p + (i + j) \cdot T. \quad (3.4)$$

In Eq. (3.4), RTT , θ_q , and θ_p are measured in the synchronization session illustrated in Fig. 3.14; i and j are unknown non-negative integers. If the i and j can be determined, the estimated offset between the slave's and the master's clocks, denoted by δ , can be computed by either one of the following formulas:

$$\delta = t_1 - (t_2 - \tau_q) = t_1 - t_2 + \theta_q + i \cdot T - \epsilon, \quad (3.5)$$

$$\delta = t_4 - (t_3 + \tau_p) = t_4 - t_3 - \theta_p - j \cdot T - \epsilon. \quad (3.6)$$

It can be easily verified that the above two formulas give the same result. The analysis in the rest of this chapter chooses to use Eq. (3.6). The ϵ is generally unknown. If we ignore it in Eq. (3.6) to compute δ , it becomes part of the clock offset estimation error.

Eq. (3.4) is an *integer-domain* underdetermined problem. Clearly, from Eq. (3.4), both i and j belong to the range $\left[0, \frac{\text{RTT} - \theta_q - \theta_p}{T}\right]$. Thus, Eq. (3.4) has a finite number of solutions for i and j . Note that, under the original NTP principle, we have a *real-domain* underdetermined problem of $\text{RTT} = \tau_q + \tau_p$ that has infinitely many solutions. NTP chooses a solution by assuming $\tau_q = \tau_p$, which does not hold in general. Thus, by introducing the Dirac combs, the *ambiguity* in determining τ_q , τ_p , and δ is substantially reduced from infinitely many possibilities to finite possibilities. Though we still have ambiguity in the integer domain, our analysis and extensive numeric results in Section 3.4.4 show that the ambiguity can be solved.

Note that, in (Rabadi et al., 2017), the periodic and synchronous power grid voltage signals collected directly from power outlets are used to synchronize two nodes that have high-speed wired network connections. The approach in (Rabadi et al., 2017) uses the elapsed times from LIs (i.e., ϕ_1 , ϕ_2 , ϕ_3 , ϕ_4) to deal with asymmetric communication delays and improve synchronization accuracy. However, due to the high-speed connectivity, it only considers the case where both i and j are zero. In contrast, with wireless connectivity, i and j are random and often non-zero due to the access time. Estimating i and j is challenging and it is the subject of Section 3.4.4.

3.4.4 Integer Ambiguity Solver (IAS)

Before we present the approach to solving integer ambiguity, we make the following two assumptions for simplicity of exposition. First, we assume that the ground-truth clock offset δ_{GT} is a constant during a synchronization process. From our performance evaluation in Section 3.7, TouchSync generally takes less than one second to achieve synchronization. Typical crystal oscillators found in MCUs and personal computers have drift rates of 30 to 50 ppm (Hao et al., 2011). Thus, the maximum drift of the offset between two clocks during one second is $50 \text{ ppm} \times 1 \text{ s} \times 2 = 0.1 \text{ ms}$. This drift is smaller than the ms-level time displacement ϵ between two iSEP signals, which dominates the synchronization error of TouchSync. Second, we assume $\epsilon = 0$. In Section 3.4.4.4, we will discuss how to deal with non-zero and time-varying ϵ .

We let $i_{\min}$ and $i_{\max}$ denote the minimum and maximum possible values for i ; $j_{\min}$ and $j_{\max}$ denote the minimum and maximum possible values for j . For instance, from our one-way message transmission time measurements (Section 3.2.2), the BLE's slave-to-master transmission times are always greater than 30 ms. Thus, we may set $i_{\min} = 1$, since in our region T is 20 ms. When we have no prior knowledge about the ranges for i and j , we may simply set $i_{\min} = j_{\min} = 0$ and $i_{\max} = j_{\max} = \frac{\text{RTT} - \theta_q - \theta_p}{T}$. Section 3.4.4.4 will discuss how the use of the prior knowledge impacts on the integer ambiguity solving.

TouchSync performs multiple synchronization sessions to solve the integer ambiguity problem. In this section, we use $x[k]$ to denote a quantity x in the k^{th} synchronization session. For instance, $\text{RTT}[k]$ denotes the measured RTT in the k^{th} session. From Eqs. (3.4) and (3.6), for the k^{th} synchronization session, we have

$$\begin{cases} \text{RTT}[k] = \theta_q[k] + \theta_p[k] + (i[k] + j[k]) \cdot T; \\ \delta = t_4[k] - t_3[k] - \theta_p[k] - j[k] \cdot T; \\ i_{\min} \leq i[k] \leq i_{\max}, \quad j_{\min} \leq j[k] \leq j_{\max}. \end{cases} \quad (3.7)$$

If TouchSync performs K synchronization sessions, we have an underdetermined system of $2K$ equations with $(2K + 1)$ unknown variables (i.e., δ and $\{i[k], j[k] | k \in [1, K]\}$). In the integer domain, such an underdetermined system can have a unique solution.

FIGURE 3.15: An example of solving the integer ambiguity. The transmissions of the auxiliary `reply2` messages are omitted in the illustration.

3.4.4.1 An example of unique solution

We use an example in Fig. 3.15 to illustrate. The unit for time is ms, which is omitted in the following discussion for conciseness. In this example, $T = 20$, $i_{\min} = j_{\min} = 1$, $i_{\max} = j_{\max} = 4$, and the ground-truth clock offset $\delta_{GT} = 105$. Two synchronization sessions are performed. The timestamps and the actual message transmission delays are shown in Fig. 3.15. The ground-truth values for i and j in the two synchronization sessions are: $i[1] = 2$, $j[1] = 1$, $i[2] = 1$, and $j[2] = 2$. The RTTs can be computed as $RTT[1] = 75$ and $RTT[2] = 78$. With any synchronous Dirac combs, from Eqs. (3.1) and (3.2), the rounded phase differences computed by the two nodes must be $\theta_q[1] = 10$, $\theta_p[1] = 5$, $\theta_q[2] = 7$, and $\theta_p[2] = 11$. For the first synchronization session, Eq. (3.7) has two possible solutions only:

$$\{i[1] = 1, j[1] = 2, \delta = 85\}, \{i[1] = 2, j[1] = 1, \delta = 105\}. \quad (3.8)$$

For the second synchronization session, Eq. (3.7) has two possible solutions only as well:

$$\{i[2] = 1, j[2] = 2, \delta = 105\}, \{i[2] = 2, j[2] = 1, \delta = 125\}. \quad (3.9)$$

From Eqs. (3.8) and (3.9), $\delta = 105$ is the only common solution. Thus, we conclude that δ must be 105.

3.4.4.2 Program for solving integer ambiguity

From the above example, due to the diversity of the ground-truth values of i and j in multiple synchronization sessions, the intersection of the δ solution spaces of these synchronization sessions can be a single value. Thus, the integer ambiguity problem is solved. On the contrary, if the ground-truth i and j do not change

Algorithm 1 Slave's pseudocode for a synchronization process

```

1: Global variables:  $t_1, t_4, \delta$ 's solution space  $\Delta = \emptyset$ , session index  $k = 0$ 
2:
3: command start_sync_session() do
4:    $k = k + 1$ 
5:    $t_1 = \text{read\_system\_clock}()$ 
6:   send message request = { } to master
7: end command
8:
9: event reply1 received from master do
10:   $t_4 = \text{read\_system\_clock}()$ 
11:  if iSEP signal strength is good:
12:    wait till iSEP data covering  $t_1$  and  $t_4$  are available
13:    run iSEP signal processing pipeline in Section 3.4.2
14:  else:
15:    generate internal periodic signal2
16:  endif
17:  compute  $\phi_1$  and  $\phi_4$ 
18: end event
19:
20: event reply2 received from master do
21:  compute  $\theta_q$  and  $\theta_p$  using Eqs. (3.1) and (3.2).
22:   $\text{RTT} = (t_4 - t_1) - (\text{reply2}.t_3 - \text{reply2}.t_2)$ 
23:  solve Eq. (3.7),  $\Delta'$  denotes the set of all possible solutions for  $\delta$ 
24:  if  $k == 1$ :
25:     $\Delta = \Delta'$ 
26:  else:
27:     $\Delta = \Delta \cap \Delta'$ 
28:  endif
29:  if  $\Delta$  has only one element  $\delta$ :
30:    use  $\delta$  to adjust clock
31:  else:
32:    start_sync_session() // start a new synchronization session
33:  endif
34: end event

```

over multiple synchronization sessions, the ambiguity remains. With application-layer timestamping, the message transmission times are highly dynamic due to the uncertain OS overhead and MAC. Such uncertainties and dynamics, which are undesirable in the original theme of NTP, interestingly, become desirable for solving the integer ambiguity in TouchSync.

From the above key observation, TouchSync performs the synchronization session illustrated in Fig. 3.14 *repeatedly* until the intersection among the δ solution spaces

Algorithm 2 Master's pseudocode for a synchronization process

```

1: event request received from slave do
2:    $t_2 = \text{read\_system\_clock}()$ 
3:   ... // execute other compute tasks
4:    $t_3 = \text{read\_system\_clock}()$ 
5:   send message reply1 = { } to slave
6:   if iSEP signal strength is good:
7:     wait till iSEP data covering  $t_1$  and  $t_4$  are available
8:     run iSEP signal processing pipeline in Section 3.4.2
9:   else:
10:    generate internal periodic signal2
11:   endif
12:   compute  $\phi_2$  and  $\phi_3$ 
13:   send message reply2 =  $\{t_2, t_3, \phi_2, \phi_3\}$  to slave
14: end event

```

of all the synchronization sessions converges to a single value. The pseudocode of the algorithms running at the slave and the master can be found in Algorithm 1 and 2.

3.4.4.3 Convergence speed

We run a set of numeric experiments to understand the convergence speed of the IAS. We use the number of synchronization sessions until convergence to characterize the convergence speed, which is denoted by K in the rest of this chapter. We fix $i_{\min}$ and $j_{\min}$ to be zero. For a certain setting $\langle i_{\max}, j_{\max} \rangle$, we conduct 100,000 synchronization processes to assess the distribution of K . For each synchronization session of a synchronization process, we randomly and uniformly generate the ground-truth i and j , as well as θ_q and θ_p within their respective ranges, i.e., $i \in [0, i_{\max}]$, $j \in [0, j_{\max}]$, and $\theta_q, \theta_p \in [0, T)$. Then, we simulate the integer ambiguity solving program presented in Section 3.4.4.2 to measure the K for each synchronization process. In practice, the i , j , θ_q and θ_p may not follow the uniform distributions. But the numeric results here help us understand the convergence speed. In Section 3.7.3, we will evaluate the convergence speed in real-world settings.

²Line 15 deals with a situation where the iSEP signal is too weak. The detailed explanation can be found in Section 3.7.

FIGURE 3.16: Convergence speed of IAS.

In Fig. 3.16a, each grid point is the average of all K values in the 100,000 synchronization processes under a certain $\langle i_{\max}, j_{\max} \rangle$ setting. Fig. 3.16b shows the box plot for K under each setting where $i_{\max} = j_{\max}$. We note that all simulated synchronization processes converge. From the two figures, even if $i_{\max} = j_{\max} = 10$ (which means that the one-way communication delays are up to 200 ms for $T = 20$ ms), the average K is nine only. Although the K 's distribution has a long tail as shown in Fig. 3.16b, 75% of the K values are below 11. This result is consistent with our real experiment results in Table 3.2 of Section 3.7.3, where most K values are two only and the largest K is 12.

3.4.4.4 Discussions

First, we discuss how to address non-zero and time-varying ϵ . From the analysis in Eq. (3.7) that is based on $\epsilon = 0$, the difference between two δ solutions is multiple of T . This can be seen from Eqs. (3.8) and (3.9). In practice, ϵ can be non-zero and time-varying. It will be a major part of the δ estimation error. From Fig. 3.8b, the $|\epsilon|$ is at most 6 ms. Thus, the resulted variation to the δ solutions will be less than a half and one third of T , in the regions served by 60 Hz and 50 Hz power grids, respectively. Therefore, we can still correctly identify the correspondence among the δ elements in the set intersection operation. Specifically, if two δ elements have a difference smaller than $T/2$, they should be considered the same element in the set intersection operation; otherwise, they are different elements. For this correspondence identification to be correct, the ϵ needs to be smaller than $T/2$. After convergence, the final δ can be computed as the average of the δ elements

that are considered the same. We have incorporated this in our implementation of TouchSync.

Second, we discuss how the use of the prior knowledge (i.e., $i_{\min}$, $i_{\max}$, $j_{\min}$, and $j_{\max}$) impacts on the integer ambiguity solving. With the prior knowledge, we may shrink the search range for i and j to speed up the convergence of the IAS. The prior knowledge can be based on the statistical information obtained in offline experiments. For instance, a group of the box plots are the results for the IAS with the prior knowledge of $i_{\max}$ and $j_{\max}$. The IAS can search the i and j within the ranges of $[0, i_{\max}]$ and $[0, j_{\max}]$, respectively. The other group of results are for the IAS without the prior knowledge. Thus, the IAS has to search within the range of $\left[0, \frac{RTT - \theta_q - \theta_p}{T}\right]$ for both i and j . We can see that, if no prior knowledge is used, the K increases. But the IAS still always converges. Once the IAS converges, the synchronization error of TouchSync mainly depends on the time displacement ϵ .

3.5 TouchSync with Internal Periodic Signal

Our design of TouchSync in Section 3.4 is based on the periodic and synchronous iSEP signals available to the slave and master nodes. However, in certain scenarios where the wearables are extremely far away from the powerlines or in a Faraday cage (e.g., an elevator cabin), the wearables can hardly sense EMR or iSEP. This section extends the design of TouchSync to deal with such situations by letting the wearables generate internal periodic signals (IPSeS) with the same period by themselves. Such IPSeS are used to drive the clock synchronization. Section 3.5.1 presents our extended design. As the IPSeS generated by any two wearables may have a random and bounded time displacement, it is interesting to investigate how this time displacement affects the performance of TouchSync, which is the subject of Section 3.5.2.

3.5.1 Extended Design

The key extensions made to the design presented in Section 3.4 include the coordination between the slave and the master on using the same IPS time period and

the generation of their IPSes. Based on the generated IPS, each node follows exactly the procedures described in Section 3.4 to estimate the clock offset. In what follows, Sections 3.5.1.1 and 3.5.1.2 present the extensions to the TouchSync protocol and an Adaptive Period Mechanism that assists the convergence of TouchSync, respectively.

3.5.1.1 Protocol for IPS-based TouchSync

We extend the TouchSync protocol in Section 3.4.3.1 to use IPSes. When the slave is to estimate the offset between its clock and master's, it transmits an initial packet to the master, indicating the beginning of the synchronization process. The packet includes the time period T for generating the IPS. The two devices will record their time instants on transmitting and receiving the initial packet as t_0^{slave} and t_0^{master} . In our approach, these two time instants are used as the first ZCs of the two devices' IPSes. Then, same as the design in Section 3.4.3.1, one **request** and two **reply** packets will be exchanged for each synchronization session. Meanwhile, the time instants, i.e., t_1 , t_2 , t_3 , and t_4 as illustrated in Fig. 3.14, are recorded by the two nodes. For each recorded time instant, the elapsed time from the corresponding LI is computed. For instance, the slave node computes ϕ_1 as $\phi_1 = (t_1 - t_0^{slave}) \bmod T$; the master computes ϕ_2 as $\phi_2 = (t_2 - t_0^{master}) \bmod T$. Then, the slave finds all the possible estimated clock offsets based on the analysis in Section 3.4.3.2. TouchSync repeatedly runs synchronization sessions until the ambiguity is solved using the IAS presented in Section 3.4.4.

In this extended design, the absolute time displacement between the two nodes is $|t_0^{slave} - t_0^{master}| \bmod T$. The above protocol transmits an initial packet to establish the initial ZCs at t_0^{slave} and t_0^{master} . As the transmission time of the initial packet is random, the resulting time displacement is also random. An alternative approach is to use NTP to establish the initial ZCs. However, as NTP is susceptible to link asymmetry that is generally true under stochastic wireless link quality, this alternative approach may not give smaller time displacements.

FIGURE 3.17: Distribution of absolute clock offset estimation errors.

3.5.1.2 Adaptive Period Mechanism (APM)

When the IPS time period is larger, the number of IPS time periods elapsed during a synchronization session (i.e., $i + j$) is likely less. Hence, a large IPS period setting can generally reduce the TouchSync session's ambiguity and speed up the convergence. As an extreme example, when the IPS time period is very large such that $i + j = 0$ (i.e., no ambiguity) and TouchSync converges after one session. However, a larger IPS time period setting will lead to larger time displacements and therefore larger clock offset estimation errors. Thus, there exist a trade-off between the convergence speed and clock offset estimation accuracy. Based on this observation, we propose APM to dynamically increase IPS period to make sure that the system can always converge. Specifically, the IPS-based TouchSync starts with a small IPS time period setting. Whenever the system cannot converge within a predefined number of synchronization sessions, it increases the IPS time period and restarts IAS. In other words, APM keeps increasing the IPS period until the ambiguity is solved. APM speeds up the convergence at the cost of larger clock offset estimation error bounds. The preset maximum number of synchronization sessions allowed for each IPS time period setting is the knob provided to the system designer for choosing a satisfactory trade-off between the convergence speed and error bound. We evaluate this trade-off via numeric experiments in Section 3.5.2.2.

3.5.2 Numeric Experiments

We conduct numeric experiments to evaluate the performance of the IPS-based TouchSync in terms of clock offset estimation error and convergence speed, when the time displacement between the two self-generated IPSes varies. Specifically, we

FIGURE 3.18: An example of IPS-based TouchSync. (unit: ms)

FIGURE 3.19: Convergence speed under various settings of IPS period.

develop a simulator that can simulate the packet transmissions between the slave and the master. The ground-truth clock offset between the slave and the master is an arbitrary value. The first ZCs of the IPSes at the slave and the master are randomly selected. The transmission delays of the `request` and `reply` packets are uniformly and independently drawn from the range of $[0, 100 \text{ ms}]$. TouchSync repeatedly runs synchronization sessions and terminates when either the integer ambiguity is solved or the maximum session number limit of 200 is reached. The numeric experiment results are presented below.

3.5.2.1 Distribution of clock offset estimation errors

We run 10,000 synchronization processes with the IPS period fixed to 20ms. Fig. 3.17a shows the distribution of the clock offset estimation errors. We can see that the errors are bounded by 20 ms, i.e., the IPS period. Now, we use an example in Fig. 3.18 to explain this error bound. For this example, we assume that the integer ambiguity has been solved and only focus on the packet transmission from the slave to the master that is used to estimate the clock offset between the two nodes. Suppose the one-way transmission delay τ_q is 19 ms and the solved value for i is 0. In Fig. 3.18, the two trains of arrows represent the Dirac combs based on the two nodes' IPSes that are not synchronous. To facilitate illustration, we add dashed lines on the master's timeline that are synchronous with the slave's Dirac combs. As illustrated in the figure, $\phi_1 = 1 \text{ ms}$ and $\phi_2 = 2 \text{ ms}$. From Eq. (3.1), we have $\theta_q = 1 \text{ ms}$. As the two nodes independently generate IPSes, they do not know the value of the time displacement between their IPSes. Thus, from Eq. (3.5), the slave node will compute the clock offset as $\delta = t_1 - t_2 + \theta_q$. If the clocks of the two

nodes are actually synchronized such that $t_1 - t_2 = -19$, the computed $|\delta|$ of 18 ms is the absolute clock offset estimation error. From this example, if $\phi_1 \rightarrow 0$, $\phi_2 \rightarrow 0$, and $\tau_q \rightarrow 20$ ms, the absolute clock offset estimation will approach to 20 ms. This explains the error bound observed in Fig. 3.17a.

IPS-based TouchSync does not rely on any external signal, which is the same as the NTP. For comparison, we evaluate side-by-side the clock offset estimation errors of the NTP. Fig. 3.17b shows the results. The absolute clock offset estimation errors are up to 35 ms. Note that, NTP's synchronization error is $(\tau_2 - \tau_1)/2$, where τ_1 and τ_2 are the one-way transmission delays of the **request** and **reply** packets (Section 3.2.1). Hence, unlike IPS-based TouchSync that has an error bound of the IPS time period, NTP's error bound depends on the two one-way packet transmission delays. Over a highly asymmetric link, NTP's error can be much higher than the IPS period.

3.5.2.2 Convergence speed

As discussed in Section 3.5.1, if the system uses a larger IPS period setting, it will need fewer sessions to converge. To demonstrate this, we simulate the IPS-based TouchSync system under different period settings and record the number of sessions needed for solving the integer ambiguity. Fig. 3.19 shows the number of sessions needed for convergence versus the IPS period. The result clearly shows the trade-off discussed in Section 3.5.1.2. From the result, when the IPS period is 10 ms to ensure a 10 ms error bound, a total of 13 synchronization sessions are performed before convergence. If we loosen the error bound to 60 ms, the system needs 2 sessions only to converge. We note that the convergence speed is affected by the two one-way packet transmission delay, since a long transmission delay will lead to larger values of $i + j$ and thus increase the ambiguity. In Section 3.7, we will evaluate TouchSync's convergence speed under real-world settings.

3.5.3 Impact of Clock Skews

To simplify discussions, our analysis and the numeric experiments in Section 3.5.1 and Section 3.5.2 assume that the master and the slave have no clock skews. In practice, small clock skews will result in additional clock synchronization errors

for IPS-based TouchSync. According to our experiments in Section 3.5.2, when the IPS' period is 40 ms, the IPS-based TouchSync takes less than one second before the system converges. The clock drift over a second time duration is often little. For instance, the average clock drift rate of MSP430's clocks is 44.2 ppm (*MSP430 LFXT1 Oscillator Accuracy*, 2004), which will introduce a drift of 44 μ s per second. Thus, the time displacement $|t_0^{slave} - t_0^{master}|$ that ranges from 0 to the IPS period dominates the synchronization error of IPS-based TouchSync.

The validness of the domination of the time displacement over clock skew depends on the convergence time. However, the convergence time can be monitored and managed. For instance, if we desire to maintain the synchronization error caused by clock skew below 100 μ s, we can enforce a convergence time upper bound of 2 s given a drift rate of 50 ppm. If IPS-based TouchSync does not converge within 2s, the Adaptive Period Mechanism presented in Section 3.5.1.2 should be used to increase the IPS period.

In this chapter, our analysis ignores clock skews and focuses on estimating the clock offset between the master and slave. Existing clock skew compensation approaches (e.g., (Rowe et al., 2009; L. Li et al., 2011; Hao et al., 2011; Z. Li et al., 2012)) can be integrated with TouchSync to address clock skew and further improve the clock synchronization performance.

3.6 Implementation of TouchSync

Designed as an application-layer clock synchronization approach, TouchSync can be implemented as an app or part of an app, purely based on the standard wearable OS calls to sample the iSEP signal, exchange network messages, and timestamp them in the application layer. To simplify the adoption of TouchSync by application developers, we have implemented TouchSync's platform-independent tasks in ANSI C and provide them in a `touchsync.h` header file (Yan et al., 2017b). These tasks include buffer management, iSEP signal processing, and IAS. Other tasks of TouchSync, i.e., sensor sampling, synchronization message exchange and timestamping, are platform dependent. We leave them for the application developer to implement. As these tasks are basics for embedded programming, by using the

TABLE 3.1: Storage and compute overhead of TouchSync.

Platform	Memory use (KB)		Processing time (ms)			
	ROM	RAM	BPF/MRF	ZCD	PLL	IAS
Z1	10	5	364	9	48	1
Flora	17	1.9*	1.3	3	15	0.8

* Estimated based on buffer lengths.

platform-independent algorithms provided by `touchsync.h`, application developers without much knowledge in signal processing can readily implement TouchSync on different platforms. Our own Arduino and TinyOS programs that implement TouchSync’s workflow have about 50 and 150 lines of code only, respectively.

To understand the overhead of TouchSync, we deploy our TinyOS and Arduino implementations to Zolertia’s Z1 motes and Floras, respectively. The Z1 mote is equipped with an MSP430 MCU (1 MHz, 8 KB RAM) and a CC2420 802.15.4 radio. Both implementations sample iSEP at 333 Hz. On Z1, we configure the length of the circular buffer defined in `touchsync.h` to be 512. Thus, this circular buffer can store 1.5 seconds of iSEP data. This is sufficient, because the time periods $[t_2, t_3]$ and $[t_1, t_4]$ that should be covered by the iSEP signal segments to be retrieved from the circular buffer and processed by the master and slave are generally a few ms and below 100 ms, respectively. On Flora, we configure the circular buffer length to be 400 and redefine its data type such that TouchSync can fit into Flora’s limited RAM space of 2.5 KB. Table 3.1 tabulates the memory usage of TouchSync and the computation time of different processing tasks. On Z1, a total of 421 ms processing time is needed for a synchronization session. The BPF uses a major portion of the processing time. Flora cannot adopt BPF because of RAM shortage. It uses MRF instead, which consumes much less RAM and processing time. Z1 and Flora are two representative resource-constrained platforms. The successful implementations of TouchSync on them suggest that TouchSync can be readily implemented on other more resourceful platforms.

We use a Monsoon power monitor to measure the energy consumption of TouchSync on a Flora node. With TouchSync running, the node consumes 57.16 mW on average. Without TouchSync, the node’s average idle power consumption is 54.99 mW when the node is not in the sleep mode. Thus, TouchSync consumes 2.17 mW or 0.658 mA (with 3.3 V battery) on average. Assume we would like to run TouchSync on two wearable devices, each with a 150 mAh battery and a

50 ppm crystal, which can continuously run without sleep for a whole day after fully charged. In order to keep the clock offset within 7 ms, TouchSync should be activated every 140 seconds. Under such settings, the battery time of the wearable is reduced by 39 seconds per day due to the running of TouchSync. In other words, the execution of TouchSync reduces the battery time by 0.045% only. Note that when the wearable has a sleeping schedule to prolong the battery time (e.g., around 10 days by a duty cycle of 10%), TouchSync can be executed when the wearable is not in the sleep mode.

3.7 Performance Evaluation

We conduct extensive experiments to evaluate the performance of TouchSync in various real environments. Each experiment uses two Flora nodes, which act as the TouchSync slave and master, respectively. As Flora does not support BLE master mode, the two Floras cannot communicate directly. Thus, we use a RPi that operates as a BLE master to relay the data packets between the two Floras. This setting is consistent with most body-area networks with a smartphone as the hub. If the hub can sample powerline radiation or iSEP, each wearable can also synchronize with the hub directly using TouchSync. We use the approach discussed in Section 3.3.1 to obtain the ground truth clock of each Flora. The details and the results of our experiments are presented below.

3.7.1 Signal Strength and Wearing Position

As the intensity of powerline radiation attenuates with distance, iSEPs have varying signal strength. Thus, we evaluate the impact of the iSEP signal strength on the performance of the signal processing pipeline in Section 3.4.1. We measure the signal strength as follows. For a full-scale sinusoid signal with a peak-to-peak amplitude of one (normalized using ADC's reference voltage), its standard deviation is $0.5/\sqrt{2} = 0.354$. The signal strength of a normalized sinusoid with a standard deviation of σ is defined as $\sigma/0.354$. Thus, a 100% signal strength suggests a full-scale signal for the ADC. For this experiment, we use a Flora to record an iSEP signal. The strength of this signal is 34%. We feed the signal processing pipeline with this signal to generate a series of *baseline* ZCs. Then, we

FIGURE 3.20: Impact of signal strength.

FIGURE 3.21: Wearing position.

scale down the amplitude of this signal, re-quantize it, and process it using the pipeline to generate another series of ZCs. We use the mean absolute error (MAE) of these ZCs with respect to the baseline ZCs as the error metric. Fig. 3.20 shows the MAE versus the strength of the scaled down signal. When we scale down the signal by 60 times, yielding a signal strength of 0.6%, the ZCs' MAE is 0.14 ms only. This suggests that TouchSync can still detect the ZCs accurately even when the iSEP signal is rather weak.

We evaluate the impact of the wearing position on the synchrony of iSEP signals. A researcher wears a Flora on his left wrist. Then, he conducts four tests by fixing the second Flora to his right wrist, right ankle, forehead, and waist, respectively. Each test lasts for two minutes. Fig. 3.21 shows the error bars (5%-95% confidence interval) for the absolute time displacement $|\epsilon|$ between the two Floras in these four tests. The average $|\epsilon|$ values in the four tests are close. This suggests that the wearing positions have little impact on the synchrony of iSEP signals and the synchronization accuracy of TouchSync.

3.7.2 Impact of High-Power Appliances on TouchSync

Since the periodic iSEP signal is induced by the EMR field, EMR interferences may distort the iSEP signals and result in large and dynamic time displacement between two iSEP signals. This may consequently cause clock synchronization errors. High-power electrical appliances may generate time-varying EMR interferences. In this section, we investigate the impact of these appliances on the EMR waveforms and the time displacement between the EMR signals collected from two nearby Floras. We conduct experiments with an electric oven and a microwave oven as

FIGURE 3.22: EMR signals near an electric oven and a microwave oven.

FIGURE 3.23: Errors introduced into time displacements of the EMR signals.

the electrical appliances, two representative high-power appliances found in home and office environments. Their rated powers are 800 W and 1050 W, respectively.

In the experiment for each appliance, we place the two Floras close to the tested appliance. The electric oven is placed on a table in a kitchen with no other appliances nearby, whereas the microwave is in an office pantry with other appliances running nearby including a fridge and a water heater. In each experiment, we switch on the tested appliance and switch it off by unplugging it from the power outlet. By investigating the data traces collected by the two Floras when the appliance is on and off, we can understand the impact of the appliance on the EMR and iSEP signals.

Fig. 3.22 shows the raw EMR signals, collected from the electric oven and the microwave oven. From Fig. 3.22a, the EMR amplitude decreases significantly when the electric oven is switched off. Note that the electric oven is a resistive load. Its change of operating status leads to a significant change of electric current through the appliance and the related powerlines. Thus, the operating status of the electric oven significantly affects the nearby EMR. In contrast, from Fig. 3.22b, the EMR amplitude does not change when the microwave is switched off. A potential cause is

that, the EMR sensed by the Flora is dominated by several other nearby appliances including a fridge and an automatic water heater. Microwave ovens generate EMR at 2.45 GHz. However, such high-frequency EMR cannot be effectively received by the Floras. From Fig. 3.22b, the EMR signal received by the Flora is primarily at the 50 Hz. In other words, the 2.45 GHz EMR emitted by the microwave, if present, does not disrupt the 50 Hz EMR.

We evaluate the impact of the electric and microwave ovens on the synchrony of the EMR signals collected from the two Floras. Fig 3.23 shows the time displacement over the courses in Fig. 3.22. From Fig. 3.23a, after the electric oven is turned off, the intensity of time displacement fluctuation becomes larger. This is because, when the oven is switched off, the EMR strength and its signal-to-noise ratio decreases, resulting in more fluctuating time displacement. This suggests that, TouchSync’s clock synchronization accuracy is better when a nearby high-power resistive load is operating. But the mean value of the time displacement is still around zero. From Fig. 3.23b, the time displacement keeps stable. From the results in Fig. 3.23, we can see that the nearby high-power appliances introduce little impact on the time displacement between two EMR signals and thus the performance of TouchSync.

3.7.3 Evaluation in Various Environments

We evaluate the iSEP signal strength and the accuracy of TouchSync in various indoor environments.

3.7.3.1 Laboratory

We conduct experiments in a computer science laboratory with about 100 seats and various office facilities (such as lights, computers, printers, projectors, or meeting rooms). Fig. 3.24 show the floor plans of laboratory. We arbitrarily select nine test points, marked by “Lx” in the figure. A researcher carries the Floras to each test point and conducts two experiments. In the first experiment, the Floras have no physical contact with human body; in the second experiment, the researcher wears the two Floras. Thus, the experiment evaluates the same-wearer scenario. The example applications mentioned in Section 1.1, i.e., motion analysis, and muscle

TABLE 3.2: Signal strength and TouchSync accuracy.

	Test point	Without skin contact			With skin contact		
		Signal strength	K	error (ms)	Signal strength	K	error (ms)
Laboratory	L1	2.6%	3	-0.2	84.7%	2	-0.7
	L2	3.2%	2	0.3	31.5%	3	-0.7
	L3	2.3%	2	-2.5	26.1%	2	0.5
	L4	4.0%	1	-0.6	33.7%	2	0.0
	L5	0.8%	15	1.1	3.3%	2	-0.2
	L6	5.7%	10	-0.4	39.5%	10	-0.0
	L7	2.3%	n.a.	n.a.	3.0%	2	-0.9
	L8	4.6%	2	-1.5	8.3%	2	0.6
	L9	2.6%	1	-1.2	67.4%	2	-0.9
Home	H1	4.2%	2	-1.1	8.9%	2	-0.8
	H2	3.4%	1	-0.9	14.5%	2	-1.0
	H3	4.6%	1	-1.3	44.9%	2	0.2
	H4	7.8%	n.a.	n.a.	39.2%	2	0.3
	H5	3.8%	1	-1.6	3.9%	1	-2.8
	H6	3.9%	4	-4.4	9.9%	2	-2.3
	H7	5.0%	2	-1.9	6.8%	1	-2.9
	H8	8.2%	1	-11.5	54.7%	4	-1.3
	H9	2.9%	1	-2.4	9.1%	1	-1.3
Office	O1	4.0%	n.a.	n.a.	3.3%	4	0.4
	O2	5.6%	1	-7.9	2.9%	2	-1.6
	O3	1.7%	1	-0.4	3.9%	2	-0.3
	O4	5.4%	3	-2.5	5.8%	2	-0.8
	O5	4.8%	6	-6.2	5.6%	12	-0.2
Corridor	C1	3.6%	12	0.1	4.4%	11	0.7
	C2	6.2%	2	0.6	44.2%	2	-1.0
	C3	5.8%	1	-7.6	4.4%	1	-1.1
	C4	1.9%	1	-6.0	2.2%	1	-2.8
	C5	1.9%	2	-3.7	5.8%	1	-1.0

* n.a. means that TouchSync cannot converge due to large ϵ .

activation monitoring, belong to this scenario. Each synchronization session takes about 150 ms. A synchronization process completes once the IAS converges.

The first part of Table 3.2 shows the iSEP's signal strength, the number of synchronization sessions until convergence (K), and the clock offset estimation error at each test point. Without skin contact, the signal strength is a few percent only. But TouchSync can still achieve a 3 ms accuracy. At L7, TouchSync cannot converge because of large and varying ϵ . The skin contact significantly increases the signal strength. Moreover, TouchSync converges after two synchronization sessions

FIGURE 3.24: Laboratory floor plan with test points marked.

in most cases. For $K = 2$, a synchronization process takes less than one second. The absolute clock offset estimation errors are below 1 ms, lower than those without skin contact. However, without skin contact, the accuracy does not substantially degrade. This suggests that, TouchSync is resilient to the loss of skin contact due to say loose wearing.

3.7.3.2 Home

FIGURE 3.25: Home floor plan with test points marked.

We conduct experiments in a 104 m² three-bedroom home with typical home furniture and appliances. Fig. 3.25 show the floor plans of the home, marked by “Hx” in the figure. The second part of Table 3.2 shows the results. Without skin contact, the signal strength results are similar to those obtained in the laboratory. With skin contact, the signal strength increases and the absolute clock offset estimation errors are below 3 ms.

3.7.3.3 Office

The third part of Table 3.2 shows the results obtained at five test points in a 15 m² office. At test points O1 and O2, the signal strength with skin contact is slightly lower than that without skin contact. This is possible as the two tests were conducted during different time periods and the powerline radiation may vary over time due to changed electric currents. With skin contact, TouchSync gives sub-ms accuracy except at O2.

3.7.3.4 Corridor

We select five test points with equal spacing in a 200 m corridor of a campus building. The fourth part of Table 3.2 gives the results. With skin contact, TouchSync yields absolute clock offset estimation errors of about 1 ms except at C4. The errors with skin contact are lower than those without skin contact.

In summary, with skin contact, TouchSync gives sub-ms clock offset estimation errors at 20 test points out of totally 28 test points in Table 3.2. All errors are below 3 ms.

From Table 3.2, at 3 out of 28 test points, the test without skin contact gives higher signal strength than that with skin contact. This is because the two tests are conducted sequentially and the EMR may change over time. Moreover, the signal strength exhibits significant variation at different locations. This is because the EMR decays with the distance from the powerline. Nevertheless, the above results show the pervasive availability of iSEP in indoor environments.

FIGURE 3.26: One-way delays over a ngrok tunnel.

FIGURE 3.27: Accuracy of TouchSync-over-Internet.

3.7.4 TouchSync-over-Internet

Tightly synchronizing wearables over long physical distances is often desirable. For instance, in distributed virtual reality applications, tight clock synchronization among participating sensing and rendering devices that may be geographically distributed is essential. Although the synchronization can be performed in a hop-by-hop manner (e.g., wearables $\leftrightarrow$ smartphone $\leftrightarrow$ cloud), errors accumulate over hops. In particular, tightly synchronizing a smartphone to global time has been a real and challenging problem – tests showed that the synchronization through LTE and Wi-Fi experiences hundreds of ms jitters ([Lazik et al., 2015](#)). In contrast, TouchSync can perform end-to-end synchronizations for wearables distributed in a geographic region served by the same power grid. The basis is that, the 50/60 Hz power grid voltage, which generates the powerline radiation, is highly synchronous across the whole power grid ([Viswanathan et al., 2016](#)). TouchSync can synchronize wearables directly with a cloud server in the same region. The smartphone merely relays the messages exchanged among the wearables and the cloud server if the wearables cannot directly access Internet. The cloud server can use a sensor directly plugged in to a power outlet to capture the power grid voltage. Owing to the Internet connectivity, the end-to-end synchronization scheme greatly simplifies the system design and implementation.

We conduct a proof-of-concept experiment of TouchSync-over-Internet as follows. Two researchers carry a Flora-RPi setup each to two buildings that are about 10 km apart. The RPi is attached with an Adafruit GPS receiver to obtain ground-truth coordinated universal time (UTC) with μs accuracy. The two nodes, one as TouchSync master and the other as TouchSync slave, communicate through a tunnel established by ngrok 1.7, an open-source reverse proxy often adopted for

creating IoT networks. Fig. 3.26 shows the distributions of the two one-way delays over the ngrok tunnel. We can see that the ngrok exhibits significant dynamics. We evaluate TouchSync-over-Internet for eight times during a day. Fig. 3.27 shows the box plots of the time displacements (i.e., ϵ) between the iSEPs captured by the two nodes. We can see that ϵ varies from -2 ms to 9 ms during the day. From the building managements, the two rooms where the master and slave nodes are located draw electricity from the R and Y phases of the power grid, respectively. There is a phase difference of $20/3 = 6.7$ ms between these two phases. Moreover, from power engineering, the difference between the power grid voltage phases at different geographic locations is non-zero and time-varying. These factors lead to the non-zero and time-varying ϵ in Fig. 3.27. The dotted line in Fig. 3.27 shows the synchronization errors of TouchSync-over-Internet (i.e., δ) in various experiment runs. They are within the range of ϵ , since ϵ is the major source of TouchSync's synchronization error. The largest δ is 7 ms. The integer ambiguity solver converges within 4 to 13 synchronization sessions.

Chapter 4

Touch-to-Access Device Authentication Using iSEPs

In this chapter, we present TouchAuth, a device authentication approach based on iSEPs¹. Section 4.1 reviews related work. Section 4.2 presents the system and threat models, the approach overview, and the research objective. Section 4.3 presents the measurement study. Section 4.4 and Section 4.5 design and evaluate TouchAuth, respectively. Section 4.6 discusses several issues.

4.1 Background and Related Work

4.1.1 Device Authentication and Key Generation

Various physiological signals have been exploited for contact-based *device authentication* and *key generation*. Key generation establishes a secret symmetric key for a pair of nodes on the same human body. Using ECG and PPG for the above two tasks has received extensive research. An early work (Poon et al., 2006) encodes the interpulse intervals (IPIs) of ECG or PPG into a bit sequence and performs authentication by comparing the Hamming distance of two bit sequences with a threshold. The study (F. Xu et al., 2011) generates IPI-based symmetric key for an Implantable Medical Device (IMD) and an external device. PSKA

¹This chapter is partially published on (Yan, Song, Tan, Li, & Kong, 2019).

(Venkatasubramanian et al., 2010) and OPFKA (Hu et al., 2013) generate keys from certain ECG/PPG features. Rostami et al. (Rostami et al., 2013) quantify ECG’s randomness in terms of entropy and design the H2H authentication protocol. However, ECG/PPG sensors often have large form factors due to the required physical distances between electrodes. Moreover, ECG/PPG sensing can be vulnerable to video analytics (Poh, McDuff, & Picard, 2010; Wu et al., 2012). TouchAuth achieves comparable detection accuracy within shorter sensing times. Recent studies have exploited EMG (L. Yang et al., 2016) and gait (W. Xu, Revadigar, Luo, Bergmann, & Hu, 2016) for key generation. However, the multi-electrode EMG sensor (L. Yang et al., 2016) is sizable and must be placed close to muscles. Walking to generate keys (W. Xu et al., 2016) may be inconvenient and the used inertial measurement units (IMUs) may be vulnerable to remote acoustic attack (Trippel, Weisse, Xu, Honeyman, & Fu, 2017). In Section 4.6.6, we will compare the performance of our iSEP-based approach with the above physiological sensing approaches.

4.1.2 Human Body Coupled Capacitive Sensing

The iSEP sensing belongs to a broader area of capacitive sensing. A recent survey (Grosse-Puppenthal et al., 2017) provides a taxonomy of capacitive sensing. We review the paper that passively sense the mutual impact between the human body and ambient EF. The iSEP has been used for touch (Cohn et al., 2011) and motion sensing (Cohn, Gupta, et al., 2012), keyboard stroke detection (Elfekey & Bastawrous, 2013), and gesture recognition (Cohn, Morris, et al., 2012). Several studies use a single off-body electrode to sense the change of ambient EF due to human’s electrophysiological signals (R. Prance, Beardsmore-Rust, Watson, Harland, & Prance, 2008) and body movement (Takiguchi, Wada, & Toyama, 2007; H. Prance, Watson, Prance, & Beardsmore-Rust, 2012). Platypus (Grosse-Puppenthal et al., 2016) uses an EF sensor array on the ceiling to localize and identify a human walker. The EF change is due to the triboelectric effect and changes in capacitive coupling between the walker and the environment. Wang et al. (Wang et al., 2015) use an external sound card as the ADC and three magneto-inductive coil sensors to collect the electromagnetic interference (EMI) radiated from various devices. The signatures contained in the EMI are used for identifying the device the user is touching. Laput et al. (Laput, Yang, Xiao, Sample, &

(Harrison, 2015) attach a modified software-defined radio to the human body for sampling iSEP. When the user touches an object, the class of the object can be recognized based on the sampled iSEP signal. Yang et al. (C. Yang & Sample, 2016) develop a follow-up research of (Laput et al., 2015) to recognize the identity, rather than the class, of the touched object. However, the needed training phase of (Wang et al., 2015; Laput et al., 2015; C. Yang & Sample, 2016) introduces overhead.

The human body can be used as a communication channel. Early studies (Zimmerman, 1995; Matsushita et al., 2000; Park et al., 2006; Baldus et al., 2009) build customized transmitter and receiver for intra-body communication (IBC). Vu et al. (Vu et al., 2012) design a wearable transmitter to convey identification data to a touchscreen as the receiver. Holz et al. (Holz & Knaust, 2015) use a wrist wearable and touchscreen to measure bioimpedance and identify the user. Hesar et al. (Hesar, Iyer, & Gollakota, 2016) uses fingerprint scanner and touchpad as the transmitter and a software-defined radio attached to skin as the receiver. Yang et al. (C. J. Yang & Sample, 2017) show that the transmitters can be LEDs, buttons, I/O lines, LCD screens, motors, and power supplies. Roeschlin et al. (Roeschlin et al., 2018) design an IBC approach that estimates the body channel characteristics to pair on-body devices. Although IBC can be used for contact-based device authentication, it often requires non-trivial transmitter/receiver devices. In contrast, our approach requires a ubiquitous low-speed ADC only.

4.1.3 Other Related Studies

VAuth (Feng, Fawaz, & Shin, 2017) uses a wearable token device to sense the vibrations caused by the speech of the wearer and match the vibration signal with the received voice signal. The matched vibration and voice signals are further used to verify that the voice signal received by a voice assistant is really from the token wearer. Nyemkova et al. (Nyemkova & Shandra, 2018) study the distinguishability of various electronic devices based on the fluctuations of their internal EMI.

4.2 System Overview and Research Objective

4.2.1 System Model

We consider an authentication system with two devices that have been previously paired, i.e., an *authenticator* and an *authenticatee*. We assume that the two devices have a wireless communication channel, e.g., Wi-Fi, BLE, or Zigbee. The pairing enables them to communicate. The authenticator is a trustworthy device that can sense the iSEP signal $s(t)$, $\forall t$, at a location $\mathcal{L}$ on the body of a user $\mathcal{U}$. To be authenticated, the authenticatee presents its sensed iSEP signal $s'(t)$, $t \in [t_1, t_2]$, to the authenticator. The $\ell = t_2 - t_1$ is called *signal length*. The authenticatee is *valid* only if it has physical contact with a location $\mathcal{L}'$ on $\mathcal{U}$ which is close to $\mathcal{L}$ such that $s'(t) \approx s(t)$, $\forall t \in [t_1, t_2]$; otherwise, it is *invalid*. The valid authenticatee will be granted a certain access; the invalid authenticatee will be denied the access. We assume that the clocks of the authenticator and the authenticatee are synchronized, such that the authenticator can select a segment of $s(t)$ in the time duration $[t_1, t_2]$ to check the similarity between $s(t)$ and $s'(t)$ for same-body contact detection. Before the authentication process, clock synchronization can be achieved using existing approaches ([Maróti et al., 2004](#); [Z. Li et al., 2012](#)).

We now discuss the roles of different devices in the scenarios discussed in Section 1.2. When the user with a wrist wearable token touches a smartphone to unlock its screen, the wearable token is the authenticator, whereas the automatic unlock program on the phone is the authenticatee. On detecting human touch (by either button/touchscreen press or increased iSEP intensity), the unlock program presents its captured iSEP signal to the wearable token that will perform the same-body detection. A positive detection result allows the program to unlock the smartphone; otherwise, the program should not unlock the phone. In the example of worn medical sensor access, the medical sensor is the authenticator, whereas the wireless reader is the authenticatee. Only the reader that has physical contact with the sensor wearer will receive a one-time password to access the data on the sensor. In the less access-critical examples of personalizing smart objects, the wearable token (i.e., the authenticator) transmits the user's identity to the touched smart object (i.e., the valid authenticatee) for personalization.

FIGURE 4.1: Left: A use scenario where the smart watch personalizes a remote control and the associated media system by a touch; Right: authentication process.

4.2.2 Threat Model

We adopt the same threat model that is used for an ECG-based device authentication system in (Rostami et al., 2013). Specifically, we consider an adversary who fully controls the communication channel between the authenticator and any valid authenticatee and aims at impersonating the valid authenticatee. The channel control includes eavesdropping, dropping, modifying, and forging messages as desired. The adversary can corrupt neither the authenticator nor the valid authenticatee.

4.2.3 Approach Overview

Fig. 4.1 illustrates an authentication process of our approach. The authentication begins once the authenticatee detects a human touch based on iSEP. After a handshake with a nearby authenticator, a Transport Layer Security (TLS) connection is set up between the authenticator and the authenticatee to ensure data confidentiality, integrity, and freshness of consequent communications. TLS is feasible on mote-class platforms (Fouladgar, Mainaud, Masmoudi, & Afifi, 2006; Schmitt, Kothmayr, Hu, & Stiller, 2017). Because the authenticatee’s certificate presented during the TLS setup needs not to be validated by the authenticator, our approach does not involve a cumbersome public key infrastructure (PKI). Then, the two parties synchronize their clocks and sample their respective iSEPs $s(t)$ and $s'(t)$ synchronously for ℓ seconds. After that, following an existing protocol H2H (Rostami et al., 2013) that is designed for ECG-based device authentication, the two parties perform a commitment-based data exchange to ensure the security of the system

against the threat defined in Section 4.2.2. Note that, without using H2H, a naive approach of transmitting $s'(t)$ from the authenticatee to the authenticator over the TLS connection for contact detection is vulnerable to a man-in-the-middle attack based on full channel control (Rostami et al., 2013). After obtaining $s'(t)$, the authenticator runs a same-body contact detection algorithm with $s(t)$ and $s'(t)$ as inputs to decide whether the authenticatee is valid. Lastly, the authenticator notifies the authenticatee of acceptance or rejection.

Note that in the less security-critical use scenarios such as smart object personalization, the TLS connection setup can be skipped and the commitment-based data exchange procedure can be replaced with a normal data exchange procedure. This reduces overhead.

4.2.4 Research Objective

As discussed in Section 2.3, the human body in an EF is an equipotential conductor. Considering two sensors with their ADC pins connected to the same human body, the potentials of their ADC pins will be the same. If they are close to each other, their grounds will have similar potentials. Thus, their readings will be similar. If the two sensors are attached to two locations on the human body which are far from each other, their grounds will have different potentials. As a result, though their ADC pins have the same potential due to the human body contact, their readings will be different.

Now, we discuss the case where the two sensors are on different human bodies. The human bodies will most likely have different potentials. Moreover, even if we ignore the impact of the two human bodies on the EF, because the two sensors are at two different locations, the gradients of the indoor EF at the two locations will be most likely different. As a result, the two sensors' measurements will be different. This difference will be further intensified by the different impacts of the two human bodies on their nearby EFs.

Our research objective is two-fold. First, we aim to verify the above inferences from the iSEP electrostatics via an extensive measurement study, which is the subject of Section 4.3. If the measurement results are supportive of the inferences,

FIGURE 4.2: Z1.

FIGURE 4.3: Sensor placements.

we will inquire whether iSEP sensing can be exploited to implement the desirable touch-to-access scheme. This will be addressed in Section 4.4 and Section 4.5.

4.3 Measurement Study

4.3.1 Measurement Setup

Our experiments are conducted using several Zolertia Z1 motes ([Zolertia Inc, 2018](#)) and a Kmote ([KETI, 2009](#)). Both types of motes are equipped with MSP430 microcontroller and CC2420 802.15.4 radio. The Z1 motes are used to collect iSEP data from human bodies, whereas the Kmote is used as a base station to synchronize the Z1 motes' clocks and collect their iSEP data over wireless. Each Z1 mote is powered by a lithiumion polymer battery; the Kmote base station is connected to a desktop computer through a USB cable. Each Z1 mote has two Phidgets sensor ports connected to several ADC pins of its microcontroller. We use a conductive wire as an electrode to create a physical contact between a pin in one Phidgets sensor port and the skin of the Z1 wearer. Fig. 4.2 shows a Z1 worn on a wrist. Each experiment involves two people. Fig. 4.3 shows the sensor placements on the bodies. The motes run TinyOS 2.1.2. The program running on the Z1 mote samples the ADC at a rate of 500 sps. The samples are timestamped using the Z1's clock. The program uses a reliable transmission protocol called Packet Link Payer ([Moss & Levis, 2007](#)) to stream the samples to a Kmote base station. The

FIGURE 4.4: Time-varying magnetic field generator.

program integrates the Flooding Time Synchronization Protocol (FTSP) ([Maróti et al., 2004](#)) to synchronize the Z1's clock to the Kmote base station.

4.3.2 Measurement Results

We conduct three sets of experiments in a lab office.

4.3.2.1 Insensitivity to time-varying magnetic field (MF)

To verify that the body antenna effect is mainly caused by EF, rather than MF, we build an MF generator and examine its impact on iSEP sensing. Fig. 4.4 shows the schematic and the implemented MF generator. It consists of a *power amplifier* that weighs 2.6 kg, a 320 mH inductor, and a 1 μ F capacitor. The power amplifier admits a specified signal waveform and outputs the corresponding current to induce the inductor to generate time-varying MF. The capacitor is used to smooth the output signal during the induction. In this experiment, the specified signal is a 85 Hz sinusoid. The choice of this frequency has two reasons: a) As 85 Hz is close to the grid frequency of 50 Hz in our region, the iSEP sensor have similar signal reception performance as for the 50 Hz signal from powerlines; b) The choice of 85 Hz is to avoid the harmonics of the grid frequency, i.e., 100 Hz, 150 Hz and so on. We configure the power amplifier to use its maximum gain. From our tests, this generator causes strong interference to nearby tank circuits that can sense MF changes. However, from our experiments, it generates little impact on nearby on-body Z1-based iSEP sensors. From a frequency analysis, the power density of the iSEP signal at 85 Hz is 66 times weaker than that at 50 Hz when the on-body

iSEP sensor is only 2 cm away from the inductor. The intensity of the signal at 85 Hz is similar to that of the ambient noise. The reason is that the iSEP sensor (i.e., an ADC with floating ground) is an open circuit, which cannot be induced by the time-varying magnetic field. This result confirms that the body antenna effect is mainly caused by EF.

4.3.2.2 iSEPs on the same body

First, Person A sits in a chair and uses his right hand palm to hold two Z1 sensors steadily. The ADCs of both sensors have direct contact with the palm skin. In Fig. 4.3, the nodes numbered ❶ and ❷ illustrate the placement of the two sensors. Fig. 4.5a shows the iSEPs captured by the two sensors over two seconds. Fig. 4.5b shows a zoomed-in view of Fig. 4.5a. From the two figures, we can see that the two iSEP signals are synchronous and of the same amplitude level. This shows that, when the two sensors are in proximity on the same human body, their measurements are similar.

Second, we investigate the impact of spatial location on iSEP. As discussed in Section 2.1, the indoor EF has an intensity distribution over space. Thus, the potential difference between the human body and the ground of the sensor varies with location. In this experiment, Person A holds a sensor in his palm with skin contact and stands at two spots in the lab. Fig. 4.6a and Fig. 4.6b show the iSEPs at the two spots that are about one meter apart. From the two figures, the amplitude of the iSEP at Spot X is larger than that at Spot Y. Note that Spot X is closer to a cubicle with a number of electrified power cables and power extensions.

Third, we investigate the impact of the sensor placement on the received iSEP signal. We place three sensors on Person A, two on the right arm and the remaining one on the left arm. The two sensors on the right arm are separated by about 15 cm, one of which is close to the wrist and the other is close to the elbow. In Fig. 4.3, the nodes numbered ❸, ❹, and ❺ illustrate the placement of the three sensors. In this experiment, the person stands and keeps a side lateral raise posture. Fig. 4.7 shows the iSEP signals collected from the three sensors in the same time period. Fig. 4.7a shows the iSEPs measured by the two sensors on the right arm. Fig. 4.7b shows the iSEPs measured by two sensors on different arms. Fig. 4.7c shows the zoomed-in view for the signals in Figs. 4.7a and 4.7b. From the results, we can

(A) Measurements of two sensors in two seconds.

(B) Zoomed-in view.

FIGURE 4.5: iSEPs measured by two sensors in the same palm when the holder sits in a chair.

(A) Spot X.

(B) Spot Y.

FIGURE 4.6: iSEPs measured by a sensor in the same palm when the wearer stands at different spots in the lab.

FIGURE 4.7: iSEPs at different locations of Person A.

see that the signals measured by the two sensors on the right arm have similar amplitudes, but a phase shift of about 180° . This can be caused by that the ADC-to-ground directions of the two sensors in the EF are different. Ignoring the phase shift, the signals measured by the two sensors 15 cm apart on the same arm exhibit higher similarity than those measured by the two sensors on different arms, but lower similarity than those measured by the two sensors in the same palm as shown in Fig. 4.5b.

We use the absolute Pearson correlation coefficient (APCC) to quantify the similarity between two iSEP signals. The two bars in the first bar group labeled (a) in Fig. 4.8 show the APCCs between two iSEP signals collected from the same and different arms on the same person, respectively. The above results suggest that the correlation between the iSEPs is affected by the distance between the sensors. When the two sensors are closer, their iSEPs exhibit higher correlation. This is supportive of our discussion in Section 4.2.4.

FIGURE 4.8: APCC in different placements.

FIGURE 4.9: APCC under different clock offsets.

(A) Two nodes in A's right palm.

(B) Two nodes on two persons.

(c) Zoomed-in view.

FIGURE 4.10: iSEPs measured by three sensors on two persons who sit steadily 1 m apart.

4.3.2.3 iSEPs on different bodies

In the first experiment, we place two sensors in the palm of Person A and another sensor in the palm of Person B. The two persons sit steadily 1 m apart. In Fig. 4.3, the nodes numbered ❶, ❷, and ❸ illustrate the placement of the three sensors.

FIGURE 4.11: iSEPs measured by three sensors on two persons who sit 1 m apart and perform random hand movements.

Fig. 4.10 shows the results. Fig. 4.10a and Fig. 4.10b show the iSEPs measured by the two sensors in Person A's palm and the two persons' palms, respectively. Fig. 4.10c shows the zoomed-in view. From the results, we can see that the iSEP on Person B is clearly different from that on Person A, in terms of both signal amplitude and waveform. In contrast, the iSEPs on Person A are very similar. The two bars in the second bar group labeled (b) in Fig. 4.8 show the APCCs for the cases shown in Figs. 4.10a and 4.10b. Clearly, the iSEPs from the same body exhibit higher correlation than those from different bodies. This result is supportive of our discussion in Section 4.2.4.

In the second experiment, we investigate whether movements affects the distinctiveness. We ask the two persons to perform some random hand movements. Fig. 4.11 and the third bar group labeled (c) in Fig. 4.8 show the results. We can see that, in the presence of movements, the iSEPs from the same body still exhibit higher correlation than those from different bodies.

In the above experiments, the clocks of the sensors are tightly synchronized using FTSP that uses MAC-layer timestamping to achieve microsecond-level synchronization accuracy. Platforms without MAC-layer timestamping can achieve millisecond-level synchronization accuracy (Z. Li et al., 2012). We now assess the impact of a clock synchronization error of up to 10 ms on the APCC. As our collected iSEP signals are tightly synchronized, we simulate the clock synchronization error by offsetting an input iSEP signal. Fig. 4.9 shows the APCC under different simulated clock offsets among the signals in Fig. 4.11. We can see that, in the presence of clock synchronization error, the APCC for the signals from the same person is generally higher than that for different persons. Moreover, when the synchronization error is around -5 ms or 5 ms, the APCCs are nearly zero. This is because the two signals have a phase difference of 90° , resulting in near-zero correlations. Chapter 3 shows that by using iSEP, wearables on the same person or two nearby persons can maintain the synchronization errors below 3 ms. Such synchronization errors do not subvert the APCC as an effective similarity metric.

4.4 Same-Body Contact Detection

From Section 4.3.2, iSEP is promising for touch-to-access device authentication. In this section, we present the design of the same-body contact detection algorithm (Section 4.4.1) and discuss a *mimicry attack* that aims at subverting the algorithm (Section 4.4.3).

4.4.1 Detection Algorithm

Before TouchAuth detects the same-body contact, it checks the iSEP signal strength. Specifically, if the standard deviation of either $s(t)$ or $s'(t)$ is below a predefined threshold, TouchAuth rejects the authentication request without performing same-body contact detection. This ensures that the detection is made based on meaningful iSEP signals. From our offline tests, a standard deviation threshold of 0.06 V is a good setting for the Z1 platform. Similar offline tests can be performed for other platforms. In what follows, we present the same-body contact detection algorithm. The detection performance is evaluated in Section 4.5.

The detector compares a similarity score between $s(t)$ and $s'(t)$, $\forall t \in [t_1, t_2]$, with a threshold denoted by η . If the similarity score is larger than η , TouchAuth accepts the authenticatee; otherwise, it rejects the authenticatee. We adopt the reciprocal of the root mean square error (RMSE) and the absolute Pearson correlation coefficient (APCC) as our similarity metrics. The RMSE is a variant of the Euclidean distance which has been used as a dissimilarity metric by physiological sensing approaches (Poon et al., 2006). The Pearson correlation coefficient measures the linear correlation between two variables. As shown in Fig. 4.7, the iSEP signals collected from the same arm have a phase shift of 180° , resulting in a Pearson correlation of about -1 . However, the authenticatee on the same arm as the authenticator may be accepted. This motivates us to use the APCC as the similarity metric that ranges from 0 to 1, with 0 and 1 representing the lowest and the highest similarity values, respectively. In the rest of this chapter, the TouchAuth based on the RMSE and APCC is called *RMSE-TouchAuth* and *APCC-TouchAuth*, respectively.

Dynamic time warping distance (DTWD) is a widely adopted dissimilarity metric that can address time-varying phase shift. From our experiments, it may wrongly help the invalid authenticatee who is spatially close to the authenticator. Thus, we do not adopt DTWD.

4.4.2 Assessment metrics

This chapter uses the false acceptance rate (FAR or simply α) and the true acceptance rate (TAR or simply β) as the main detection performance metrics. The α and β are the probabilities that an invalid or valid authenticatee is wrongly or correctly accepted, respectively. The detection threshold η and the signal length ℓ are two important parameters. The receiver operating characteristic (ROC) curve of β versus α by varying η depicts fully the performance of a detector under a certain ℓ . The signal length ℓ characterizes the sensing time needed by the authentication process. In this chapter, we use the ROC curves to compare the detection performance of various detectors. In practice, the settings of η and ℓ can follow the Neyman-Pearson lemma to enforce an upper bound for α . A stringent α is often required by authentication. For instance, with $\alpha = 1\%$, an invalid authenticatee needs to repeat the authentication process 100 times on average to be successful,

FIGURE 4.12: Detection performance of APCC-TouchAuth when ℓ is 0.5 s, 1 s, and 2 s, respectively.

which is frustrating if some after-rejection freeze time is enforced. Moreover, an authenticatee device can be banned if it is continuously rejected for many times.

Fig. 4.12 shows the detection performance of APCC-TouchAuth assessed by using the data shown in Fig. 4.11. Fig. 4.12a shows the α and the false rejection rate (FRR) versus the detection threshold η when ℓ is 0.5 s, 1 s, and 2 s, respectively. Note that $\text{FRR} = 1 - \beta$. Fig. 4.12b shows the ROC curves when α is from 0 to 2%. Note that the α and β values of each point on the ROC are measured based on 500 tests. From the figure, we can see that when $\ell = 2$ s, APCC-TouchAuth achieves a β value of 100% (i.e., correctly accepts all 500 tests when the authenticatee is valid) while keeping $\alpha = 0\%$ (i.e., correctly rejects all 500 tests when the authenticatee is invalid). This suggests that APCC-TouchAuth can achieve a very high detection accuracy. The equal error rate is 3.2%, 1.6%, 0.5% when ℓ is 0.5 s, 1 s, and 2 s, respectively. From Fig. 4.12b, the ROC curve under a smaller ℓ setting becomes lower, suggesting lower detection accuracy. Section 4.5 extensively evaluates the detection performance of TouchAuth under a wider range of settings among a larger group of users.

In addition to the ROC that characterizes detection performance, we use the *signal-to-difference ratio* (SDR) to assess the quality of iSEP sensing. Specifically, let $P[x(t)]$ denote the average power of a signal $x(t)$. Ideally, if the authenticator and the valid authenticatee are very close to each other on the same human body, their iSEP signals $s(t)$ and $s'(t)$ should be very similar. Thus, we define the SDR in decibel as $\text{SDR} = 10 \log_{10} \frac{P[s(t)]}{P[s(t) - s'(t)]}$ dB. A high SDR suggests high-quality iSEP sensing.

4.4.3 Mimicry Attack

We now discuss a *mimicry attack* that attempts to obtain the authenticator's $s(t)$. Due to the complex spatial distribution of the indoor ambient EF, it is generally difficult for the attacker to estimate the authenticator's $s(t)$. In this attack, the attacker wearing an iSEP sensor mimics the body movements of the victim user wearing the authenticator. To be effective, the mimicry attacker should stay as close as possible to the victim user to sense the same/similar ambient EF. Thus, it is unrealistic in practice, because the strange mimic behavior in proximity can be easily discerned by the user. Note that this attack is beyond the threat model defined in Section 4.2.2 that concerns the security of data communications between the authenticator and the authenticatee. Thus, our approach described in Section 4.2.3, which is based on the secure protocol H2H, does not guarantee security against this mimicry attack. In Section 4.5, we will show the ineffectiveness of this attack experimentally.

4.5 Evaluation

We conduct a set of experiments to evaluate TouchAuth's same-body contact detection performance under a wide range of settings including different wearers, various indoor environments, multiple possible interfering sources, device proximity, skin moisture, and heterogeneous devices.

4.5.1 Performance across Different Wearers

We collect a set of data involving a wearer $\mathcal{R}$ and 12 other wearers $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_{12}$. The experiments are conducted in a computer science lab. In the i th experiment ($i = 1, \dots, 12$), $\mathcal{R}$ holds an authenticator device and a valid authenticatee device in his palm, whereas $\mathcal{P}_i$ holds an invalid authenticatee device in his palm. Thus, in this set of experiments, we evaluate the detection performance of TouchAuth for a certain user with a valid authenticatee against different users with invalid authenticatees. In each experiment, $\mathcal{R}$ and $\mathcal{P}_i$, which are about 0.5m apart, are allowed to perform some uncoordinated and random hand movements. The data

FIGURE 4.13: ROCs for 12 different wearers with the invalid authenticatee device. The x -axis and y -axis of each subfigure are α and β , respectively.

collection of each experiment lasts for two minutes. We measure the detection performance of APCC-TouchAuth and RMSE-TouchAuth as follows. Let N_L , or N_I , denote the total number of tests between the authenticator and the valid authenticatee, or between the authenticator and the invalid authenticatee. Accordingly, let N_{TA} and N_{FA} denote the total numbers of true acceptances and false acceptances, respectively. The β and α are measured by N_{TA}/N_L and N_{FA}/N_I , respectively.

Fig. 4.13 shows the APCC- and RMSE-TouchAuth's ROC curves for different wearers with the invalid authenticatee device when the signal length ℓ is 1 s. Different data points on an ROC represent the results under different detection threshold η . The SDR assessed using the authenticator's and the valid authenticatee's iSEP signals in each experiment is included in the corresponding subfigure. We can see that across different wearers with the invalid authenticatee, APCC-TouchAuth is comparable or superior to RMSE-TouchAuth in terms of the detection performance. This is because that APCC inherently captures the correlation between the iSEP signals on the same moving hand. In contrast, as the RMSE captures sample-wise differences between two signals, two uncorrelated signals with similarly small amplitudes can give a small RMSE value, leading to a false acceptance.

FIGURE 4.14: ROCs for 8 different wearers with the valid authenticatee device. The x -axis and y -axis of each subfigure are α and β , respectively.

Note that the RMSE has been adopted as a dissimilarity metric for physiological sensing (Poon et al., 2006). However, it is ill-suited for iSEP sensing because the iSEP signal amplitude has a large dynamic range depending on the ambient EF's gradient. This is different from physiological signals that often have stable ranges of signal amplitude. From Fig. 4.13, we can see that APCC-TouchAuth achieves a high β value (100%) subject to an α upper bound of 1%, except for the wearer $\mathcal{P}_{11}$. For $\mathcal{P}_{11}$, APCC-TouchAuth achieves a β value of 100% subject to an α upper bound of 4%.

We collect another set of data, where $\mathcal{R}$ wears an invalid authenticatee and $\mathcal{P}_i$ holds an authenticator and a valid authenticatee. Thus, this set of experiments evaluate the detection performance of TouchAuth for different users wearing the valid authenticatee against a certain user wearing the invalid authenticatee. Fig. 4.14 shows the ROCs for eight different wearers with the valid authenticatee. Similar to the results in Fig. 4.13, APCC-TouchAuth achieves high-profile ROCs and outperforms RMSE-TouchAuth. The results in Figs. 4.13 and 4.14 show that the detection performance of TouchAuth is not wearer-specific.

FIGURE 4.15: Experiments in indoor environments.

FIGURE 4.16: ROCs in various environments.

FIGURE 4.17: ROCs with various nearby appliances.

4.5.2 Various Indoor Environments

Two wearers conduct experiments in eight different indoor environments as shown in Fig. 4.15, which include a living room, a study room, a kitchen, two bedrooms, a corridor, a meeting room, and an open area of a lab. One wearer carries the authenticator and a valid authenticatee and the other carries the invalid authenticatee. Fig. 4.16 shows the snapshots of some environments and the measured SDRs and ROCs in the eight environments. We can see that in certain environments, the RMSE-TouchAuth performs poorly. Investigation on the raw iSEP signals shows that in these environments, the iSEP signals of the authenticator and the invalid authenticatee have similar amplitudes. In all the eight environments, APCC-TouchAuth achieves high β values ($\geq 97\%$) subject to an α upper bound of 1%. If the α upper bound is relaxed to 4%, the β value of 100% can be achieved.

4.5.3 Various Possible Interfering Sources

From our discussion in Chapter 2 and the measurement results in Section 4.3.2.1, the iSEP measurement is mainly caused by the ambient EF. The MFs generated by the operating currents of electric appliances have little impact on the iSEP

FIGURE 4.18: ROCs with interference and skin moisture.

sensing. However, some appliances, especially those based on motors and high-frequency switched-mode power, may generate interference to the iSEP sensing. This is because that unlike the 50 Hz current-induced MF that generates little/no EF, the high-frequency currents caused by the frictions between the motor's brush and stator as well as the switched-mode power may generate propagating electromagnetic waves. As a result, the EFs generated by the appliances and powerlines may weaken each other, making the overall EF weaker. Thus, we conduct a set of experiments with various home appliances including toaster, electric kettle, hair dryer, ceiling fan, blender, and induction cooker. Specifically, two wearers, one with a valid authenticatee and the other with an invalid authenticatee, stand close to a certain appliance to collect iSEP traces. Fig. 4.17 shows the SDR and the APCC-TouchAuth's ROCs for various appliances when the appliance is on and off. We can see that, for a certain appliance, the SDR may increase or decrease when the appliance is switched on. This is because that the interference from the appliance may be constructive or destructive to the EF generated by the building's power cabling. The operating status of the induction cooker causes the largest SDR change of more than 5 dB. This is due to the high-frequency switched-mode current in the cooker's internal inductor. As a result, the ROC drops slightly when the induction cooker is switched on. However, APCC-TouchAuth still achieves a high β value (100%) subject to an α upper bound of 2.5%.

The signaling phase of a cell phone call often interferes with audio systems because of the intermittent wireless power pulses. Thus, we evaluate the impact of cell phone calls on APCC-TouchAuth. In the experiments, the wearer holds a smartphone, a valid authenticatee, and the authenticator in one palm. Another wearer holding an invalid authenticatee stands 0.5 m away. Fig. 4.18(a) shows the ROCs

FIGURE 4.19: β versus ℓ .

FIGURE 4.20: Sensor proximity.

at different phases of a phone call. We can see that the phone call does not affect the detection performance of APCC-TouchAuth.

Secondly, we use a circuit seeker (Greenlee CS-8000) that is capable of up to 4 miles circuit tracing (Co., 2018) to inject noises into the power network serving the lab in which we conduct experiments. The injector of CS-8000 is plugged into a power outlet, injecting a 15 kHz signal into the power network; the seeker can detect the 15 kHz electromagnetic emanation from the powerlines. We conduct experiments in proximity of a powerline close to the injector. Fig. 4.18(b) shows the ROCs when the injector is in operation or not. We can see that the noise injection does not affect TouchAuth.

Lastly, we evaluate the impact of the skin moisture conditions on TouchAuth. We conduct two experiments, in which the user holds the authenticator using a wet hand. Meanwhile, he holds a valid authenticatee in the same hand. Another user stands 0.5 m away holding an invalid authenticatee. Fig. 4.18(c) shows the ROCs for dry and wet skin moisture conditions. We can see that the skin moisture has little impact on the performance of TouchAuth.

4.5.4 Impact of Signal Length ℓ

We evaluate the impact of the signal length ℓ on the detection performance of TouchAuth. We combine the data collected from 12 different wearers in Section 4.5.1 into a single dataset. Based on the combined dataset, Fig. 4.19 shows the β achieved by APCC-TouchAuth and RMSE-TouchAuth versus ℓ when $\alpha \leq 1\%$ or $\alpha \leq 2\%$. APCC-TouchAuth's β increases sharply when $\ell \leq 1$ s. When $\ell > 1$ s, its β increases with ℓ slowly. This suggests that a setting of $\ell = 1$ s well balances

FIGURE 4.21: β vs. ℓ under mimicry attack.

FIGURE 4.22: A Flora as the valid authenticatee.

the detection performance and sensing time. The β - ℓ curves for RMSE-TouchAuth exhibits a similar pattern. Moreover, consistent with the results in Section 4.5.1 and 4.5.2, RMSE-TouchAuth is inferior to APCC-TouchAuth.

4.5.5 TouchAuth Devices' Proximity

In the previous subsections, the authenticator and the valid authenticatee are in the same palm. In this set of experiments, they are placed at different locations on the user's body. Fig. 4.20 shows APCC-TouchAuth's ROC curves. When the two devices are on the palm and the wrist of the same hand, respectively, a high-profile ROC is achieved. When the two devices are on (i) the right palm and the right elbow, respectively, or (ii) the right palm and the head, respectively, the detection performance is degraded. This shows that TouchAuth is applicable to the example use scenarios discussed in Section 1.2 where the two devices are in proximity on the same body. In Section 4.6, we will further discuss the impact of the proximity requirement on the usability of TouchAuth.

4.5.6 Mimicry Attack

We follow the data collection methodology described in Section 4.5.1 to collect another dataset in the lab, except that each wearer $\mathcal{P}_i$ with the invalid authenticatee

mimics the hand movements of the wearer $\mathcal{R}$ with the authenticator and the valid authenticatee. The $\mathcal{R}$ performs simple and repeated hand movements, such that $\mathcal{P}_i$ can follow easily. The distance between $\mathcal{R}$ and $\mathcal{P}_i$ is about 0.5 m. Fig. 4.21 shows the APCC-TouchAuth's β versus ℓ subject to various α upper bounds. The error bars show the minimum, maximum, and mean of the β values among different $\mathcal{R}$ - $\mathcal{P}_i$ pairs in the dataset. Compared with Fig. 4.19, when ℓ is small (e.g., 0.1 s), the mimicry attack degrades APCC-TouchAuth's detection performance. However, the attack impact can be fully mitigated by adopting a larger ℓ setting (e.g., $\ell = 1$ s).

4.5.7 Heterogeneous Devices

In this set of experiments, the authenticator and the invalid authenticatee are based on Z1 motes (denoted by Z1-1 and Z1-2); the valid authenticatee is based on an Adafruit's Flora (Adafruit, 2018), an Arduino-based wearable platform. The top part of Fig. 4.22 shows a Flora-based TouchAuth prototype device with a 3D-printed insulating wristband and a conductive thread creating the body contact. We use a laptop computer to relay the communications between the Bluetooth-based Flora and the Zigbee-based Z1. The bottom part of Fig. 4.22 shows the zoomed-in view of the signals captured by the three devices. We can see that although the Z1 and Flora have different DC lines, the Z1-1 authenticator and the Flora authenticatee are highly correlated. Based on this setup, with $\ell = 1$ s, APCC-TouchAuth achieves a β value of 100% subject to an α upper bound of 1%. This result shows that TouchAuth can be applied on heterogeneous devices.

4.6 Limitations and Discussions

This section shows the limitations and discussions of TouchAuth. We first describe TouchAuth's requirements of sensor's proximity in Section 4.6.1. Then, in Section 4.6.2, 4.6.3, 4.6.4, 4.6.5, we discuss TouchAuth's applicability of iSEP injection attack, other interferences, outdoor environments, and implantable medical devices (IMD), respectively. Finally, we compare TouchAuth with existing approaches in Section 4.6.6.

4.6.1 Proximity Requirement

From the measurement study in Section 4.3 and the evaluation results in Section 4.5.5, our approach requires that the authenticator and the authenticatee are in proximity on the same human body. For instance, in the example of personalizing smart objects, the user should use the hand with the wrist wearable to touch the objects. A wireless reader needs to be placed close to a worn medical sensor to be authenticated. We believe that this proximity requirement introduces little overhead of using TouchAuth-based devices. Nevertheless, TouchAuth offers a low cost and small form factor solution based on ubiquitous ADCs only. Although exiting IBC and physiological sensing approaches may not have this proximity requirement, they generally require non-trivial sensing devices that are more costly and of larger form factors. In particular, the proximity requirement increases the barrier for active attackers to steal the iSEP signals, since they have to place a sensor close to the authenticator. In contrast, if the body-area property is effective for the whole body like for ECG/PPG, the attackers may attach a miniature sensor to the clothing of the victim to steal the signal.

4.6.2 iSEP Injection Attack

If an attacker can generate a strong ac EF that overrides the ambient EF, the attacker can infer the $s(t)$ sensed by the authenticator and spoof it to accept an invalid authenticatee. However, the strong EF generation is non-trivial and inevitably requires bulky equipment. Overriding the power grid voltage is generally impossible unless the building's power network is disconnected from the mains grid and supplied by a power generator controlled by the attacker. Another possible approach is to surround the victim TouchAuth devices with two metal plates connected with an ac generator. The bulky setting of the EF generation renders the attack easily discernible by the TouchAuth user and costly, unattractive to the attacker. Another possible attack is to generate power surges in the power network by frequently switching on and off high-power appliances like space heaters. However, the surges generate easily discernible disturbances to other appliances such as lights and audio systems. Thus, we believe that the iSEP injection attack, though possible, is unrealistic or easily discernible.

4.6.3 Other Interferences

TouchAuth is based on the instantaneous similarity of the iSEP signals in close proximity induced by the ambient EF. Hence, the similarity does not depend on the user's physiological state. Certain limited scenarios may affect the iSEP. For example, a temporary charging caused by taking off a sweater may override the iSEP in a short time. However, such situations do not happen frequently.

4.6.4 Applicability to Outdoors

The outdoor naturally occurring EF is too weak to be exploited by TouchAuth. Thus, TouchAuth is not applicable outdoors, where it rejects all authentication requests due to too weak iSEP signal strength (Section 4.4.1). As most smart objects are indoors and we spend most of our time indoors (e.g., 87% on average for Americans (Klepeis et al., 2001)), TouchAuth gives a satisfactory availability. Note that the wide availability of the iSEP signals in indoor environments has been shown in existing studies (Cohn et al., 2011; Cohn, Gupta, et al., 2012; Cohn, Morris, et al., 2012). According to our experiment results in 4.5 using the SDR value defined in 4.4.2, TouchAuth shows wide availabilities in indoor locations, such as household rooms, offices, and corridors.

4.6.5 Applicability to Implantable Medical Devices (IMDs)

Our measurement study (Section 4.3) and evaluation (Section 4.5) are based on iSEPs collected from skins. We now discuss the applicability of TouchAuth to devices implanted into human bodies. From our discussion in Section 2.1, an electrostatically induced human body is an equipotential body. Thus, the ADC pin and the ground of an IMD that is fully implanted into the human body have the same potential. As a result, the iSEP measurement is zero. We conducted a set of experiments to verify this. We bought two types of homogeneous meat from a supermarket. We wrapped a Z1 mote using cling film but leaved its ADC-connected electrode out of the wrap. We *fully* and *partially* implanted the mote into the meat. Under both settings, the electrode has significant contact with the meat. The partial implanting means that a small portion of the cling film was still

visible. The peak-to-peak amplitudes of the iSEP signals measured by the Z1 mote fully and partially implanted are about 0.05 V and 0.1 V, respectively. The peak-to-peak amplitude of the latter case is comparable to some of our measurement results on human skins (Section 4.3). Frequency analysis shows that the former is close to white noise and the latter clearly exhibits a frequency of 50 Hz. These results suggest that TouchAuth is applicable to partially implanted devices, such as insulin pumps, cochlear implants, and foot drop implants.

4.6.6 Comparison with Existing Approaches

TABLE 4.1: Comparison with existing approaches.

Ref.	Signal	Sensing time (s)	α (%)	β (%)
TouchAuth		1	2.0	94.2%
		5	2.0	98.9%
(Poon et al., 2006)	ECG+PPG	~ 60 (67 IPIs)	2.1	93.5
		~ 30 (34 IPIs)	4.5	90.5
(Venkatasubramanian et al., 2010)	PPG	12.8	0.1	99.9
(Hu et al., 2013)	ECG	~ 90 (90 IPIs)	$\sim 0^*$	$\sim 100^*$

* (Hu et al., 2013) fuzzily states that its FAR and FRR are almost zero.

We compare the performance of APCC-TouchAuth (from Fig. 4.19) and several ECG/PPG device authentication approaches introduced in Section 4.1.1. The results are shown in Table 4.1. Compared with the approach using ECG+PPG (Poon et al., 2006), APCC-TouchAuth achieve higher TAR β when FAR α is 2%. APCC-TouchAuth takes less than a tenth of sensing time that ECG+PPG approach requires. For methods using PPG (Venkatasubramanian et al., 2010) and ECG only (Hu et al., 2013), APCC-TouchAuth achieves a competitive high TAR β with shorter sensing time and a simpler implementation. With quick and accurate authentication process, compared to the existing approaches, APCC-TouchAuth applies to more IoT objects using unobtrusive and convenient interactions.

Chapter 5

Conclusion and Future Work

In this thesis, we explain the electrostatics of iSEP with supporting measurement results. Based on the understanding, we first designed TouchSync that synchronizes the clocks of wearables by exploiting the wearers' iSEP. Different from existing WSN clock synchronization approaches that find difficulties in being applied on diverse IoT platforms due to their need of hardware-level packet timestamping or non-trivial extra hardware, TouchSync can be readily implemented as an app based on standard wearable OS calls. Extensive evaluation shows TouchSync's synchronization errors of below 3 ms and 7 ms on the same wearer and between two wearers 10 km apart, respectively. With the capability to run on the application layer, more wearable applications are possible, such as virtual reality gaming, remote surgery, and synchronous audio streaming. We then designed TouchAuth for device authentication based on iSEP signals. We evaluated its same-body contact detection performance via extensive experiments under a wide range of real-world settings, including various signal strength, wearing positions, environments, and resistance to interferences. Results showed that TouchAuth achieves comparable detection accuracy as existing physiological sensing approaches, but within much shorter sensing times. Moreover, the uni-electrode iSEP sensor can be miniaturized. TouchAuth offers a low-cost, lightweight, and convenient approach for the authorized users to access the smart objects found in indoor environments. With the help of TouchAuth, users wearing a smart token device can unlock smartphones, personalize IoT devices, and access touchable objects with a simple touch.

In the future, we will study iSEP signal to implement other basic system functions such as indoor localization and key generation. As shown in our prior measurement studies of TouchAuth, iSEP is location-dependent. As the distance from the sensor to the powerline affects the iSEP signal's amplitude, iSEP may be employed for human body localization. When exploring the similarity of two iSEP signals, we find that iSEP signals from two different people are more likely to be different. Given the distinguishable shapes of each person, a unique key may be generated from iSEP signal.

Another IoT system function, i.e., voice commanding, can be enhanced with the cyber-physical approach. Because of the energy and size constraints, traditional user interfaces like keyboards or touchscreens are infeasible for various IoT devices. Hence, many IoT devices like TV remote controllers and home assistants choose voice commanding as the main user interface. However, determining the genuineness of a received voice command is a non-trivial problem. Several attack approaches have been designed to generate a voice clip that can mislead the IoT objects to execute the task designated by the attacker, while legitimate people can hardly notice the attack. Using the sensors embedded in IoT devices admitting voice commanding, two-factor verification approaches using physical signals can be designed to counteract the existing attacks.

List of Author's Publications

Conference papers

- **Zhenyu Yan**, Yang Li, Rui Tan, Jun Huang, “Application-Layer Clock Synchronization for Wearables Using Skin Electric Potentials Induced by Powerline Radiation”, in *Proceedings of the 15th ACM Conference on Embedded Networked Sensor Systems (SenSys 2017)*, November 5-8, 2017, Delft, The Netherlands.
- **Zhenyu Yan**, Qun Song, Rui Tan, Yang Li, Adams Wai Kin Kong, “Towards Touch-to-Access Device Authentication Using Induced Body Electric Potentials”, in *Proceedings of the 25th Annual International Conference on Mobile Computing and Networking (MobiCom 2019)*, October 21-25, 2019, Los Cabos, Mexico.
- Qun Song, **Zhenyu Yan**, Rui Tan, “Moving Target Defense for Deep Visual Sensing against Adversarial Examples.” in *Proceedings of the 17th ACM Conference on Embedded Networked Sensor Systems (SenSys 2019)*, November 10-13, 2019, New York, NY, USA.

Journal articles

- **Zhenyu Yan**, Rui Tan, Yang Li, Jun Huang, “Wearables Clock Synchronization Using Skin Electric Potentials”, *IEEE Transactions on Mobile Computing*.

References

- Adafruit. (2018). *Adafruit FLORA*. Retrieved from <https://www.adafruit.com/category/92> 21, 81
- Apple Inc. (2015). *Device synchronization over bluetooth*. Retrieved from <https://www.google.com/patents/US20150092642> 3
- Baldus, H., Corroy, S., Fazzi, A., Klabunde, K., & Schenk, T. (2009). Human-centric connectivity enabled by body-coupled communications. *IEEE Communications Magazine*, 47(6), 172-178. 7, 8, 59
- Barill, T. (2003). An ECG primer. In (pp. 63–100). Vancouver, BC, Canada: Nursecom Educational Technologies. 8
- Casas, R., Gracia, H., Marco, A., & Falcó, J. (2005). Synchronization in wireless sensor networks using bluetooth. In *Proceedings of the third workshop on intelligent solutions in embedded systems*. 4
- Chan, M., EstèVe, D., Fourniols, J.-Y., Escriba, C., & Campo, E. (2012). Smart wearable systems: Current status and future challenges. *Artificial intelligence in medicine*, 56(3), 137–156. 2
- Chang, S.-Y., Hu, Y.-C., Anderson, H., Fu, T., & Huang, E. Y. (2012). Body area network security: Robust key establishment using human body channel. In *Proceedings of the 3rd usenix workshop on health security and privacy*. Bellevue, WA: USENIX. 8
- Chen, Y., Wang, Q., Chang, M., & Terzis, A. (2011). Ultra-low power time synchronization using passive radio receivers. In *Proceedings of the 10th international conference on information processing in sensor networks*. 18
- Co., E. E. (2018). *Greenlee CS-8000 circuit seeker*. Retrieved from https://www.greenlee.com/catalog/product.aspx?product_id=19199 79
- Cohn, G., Gupta, S., Lee, T.-J., Morris, D., Smith, J. R., Reynolds, M. S., ... Patel, S. N. (2012). An ultra-low-power human body motion sensor using static electric field sensing. In *Proceedings of the 2012 acm conference on ubiquitous computing*. Pittsburgh, PA, USA: ACM. 8, 58, 83
- Cohn, G., Morris, D., Patel, S., & Tan, D. (2012). Humantenna: using the body as an antenna for real-time whole-body interaction. In *Proceedings of the 2012 sigchi conference on human factors in computing systems*. Austin, TX, USA: ACM. 8, 11, 58, 83
- Cohn, G., Morris, D., Patel, S. N., & Tan, D. S. (2011). Your noise is my command: sensing gestures using the body as an antenna. In *Proceedings of the sigchi*

- conference on human factors in computing systems* (pp. 791–800). Vancouver, BC, Canada: ACM. 8, 58, 83
- Dinescu, M. C., Mazza, J., Kujanski, A., Gaza, B., & Sagan, M. (2015, April 7). *Synchronizing wireless earphones*. US Patent 9,002,044. Retrieved from <https://www.google.com/patents/US9002044> 22
- Duo. (2018). *Using duo with a hardware token - guide to two-factor authentication · duo security*. Retrieved from <https://guide.duo.com/tokens> 7
- Elfekey, H., & Bastawrous, H. A. (2013). Design and implementation of a new thin cost effective ac hum based touch sensing keyboard. In *Proceedings of the 21st ieee international conference on consumer electronics*. Las Vegas, NV, USA: IEEE. 8, 11, 58
- Elson, J., Girod, L., & Estrin, D. (2002). Fine-grained network time synchronization using reference broadcasts. In *Proceedings of the 5th usenix symposium on operating systems design and implementation*. 3
- Feng, H., Fawaz, K., & Shin, K. G. (2017). Continuous authentication for voice assistants. In *Proceedings of the 23rd annual international conference on mobile computing and networking*. Snowbird, UT, USA: ACM. 59
- Ferrari, F., Zimmerling, M., Thiele, L., & Saukh, O. (2011). Efficient network flooding and time synchronization with glossy. In *Proceedings of the 10th international conference on information processing in sensor networks*. 17
- Fouladgar, S., Mainaud, B., Masmoudi, K., & Afifi, H. (2006). Tiny 3-tls: A trust delegation protocol for wireless sensor networks. In *Proceedings of the 2006 european workshop on security in ad-hoc and sensor networks*. Hamburg, Germany: Springer. 61
- Ganeriwal, S., Kumar, R., & Srivastava, M. B. (2003). Timing-sync protocol for sensor networks. In *Proceedings of the first acm conference on embedded networked sensor systems*. 3
- Gartner. (2014). *Gartner says a typical family home could contain more than 500 smart devices by 2022*. Retrieved from <https://www.gartner.com/en/newsroom/press-releases/2014-09-08-gartner-says-a-typical-family-home-could-contain-more-than-500-smart-devices-by-2022> 6
- Gartner. (2017). *Gartner says worldwide wearable device sales to grow 17 percent in 2017*. Retrieved from <http://gtmr.it/2AAHfgG> 2
- Grosse-Puppenthal, T., Dellangnol, X., Hatzfeld, C., Fu, B., Kupnik, M., Kuijper, A., ... Gruteser, M. (2016). Platypus: Indoor localization and identification through sensing of electric potential changes in human bodies. In *Proceedings of the 14th annual international conference on mobile systems, applications, and services*. Singapore: ACM. 58
- Grosse-Puppenthal, T., Holz, C., Cohn, G., Wimmer, R., Bechtold, O., Hodges, S., ... Smith, J. R. (2017). Finding common ground: A survey of capacitive sensing in human-computer interaction. In *Proceedings of the 2017 chi conference on human factors in computing systems*. Denver, CO, USA: ACM. 58
- Halperin, D., Heydt-Benjamin, T. S., Ransford, B., Clark, S. S., Defend, B., Morgan, W., ... Maisel, W. H. (2008). Pacemakers and implantable cardiac

- defibrillators: Software radio attacks and zero-power defenses. In *Proceedings of the 2008 IEEE symposium on security and privacy*. Oakland, CA, USA: IEEE. 7
- Hao, T., Zhou, R., Xing, G., Mutka, M. W., & Chen, J. (2011). WizSync: Exploiting Wi-Fi infrastructure for clock synchronization in wireless sensor networks. In *Proceedings of the 32nd IEEE real-time systems symposium*. 18, 19, 35, 45
- He, J., Duan, X., Cheng, P., Shi, L., & Cai, L. (2014). Distributed time synchronization under bounded noise in wireless sensor networks. In *Proceedings of the 53rd IEEE conference on decision and control*. 19
- Hessar, M., Iyer, V., & Gollakota, S. (2016). Enabling on-body transmissions with commodity devices. In *Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing* (pp. 1100–1111). Heidelberg, Germany: ACM. 59
- Holz, C., & Knaust, M. (2015). Biometric touch sensing: Seamlessly augmenting each touch with continuous authentication. In *Proceedings of the 28th annual ACM symposium on user interface software & technology*. Charlotte, NC, USA: ACM. 7, 8, 59
- Hu, C., Cheng, X., Zhang, F., Wu, D., Liao, X., & Chen, D. (2013). Opfka: Secure and efficient ordered-physiological-feature-based key agreement for wireless body area networks. In *Proceedings of the 32th IEEE international conference on computer communications*. Turin, Italy: IEEE. 7, 9, 58, 84
- IEEE. (2008). IEEE standard for a precision clock synchronization protocol for networked measurement and control systems. *IEEE Std 1588-2008*, 1-269. 3
- KETI. (2009). *Kmote - KETI mote*. Retrieved from <http://tinyos.stanford.edu/tinyos-wiki/index.php/Kmote> 63
- Klepeis, N., Nelson, W., Ott, W., Robinson, J., Tsang, A., Switzer, P., ... Engelmann, W. (2001). The national human activity pattern survey (nhaps): a resource for assessing exposure to environmental pollutants. *Journal of Exposure Science and Environmental Epidemiology*, 11(3), 231. 83
- Laput, G., Yang, C., Xiao, R., Sample, A., & Harrison, C. (2015). EM-Sense: Touch recognition of uninstrumented, electrical and electromechanical objects. In *Proceedings of the 28th annual ACM symposium on user interface software & technology*. Charlotte, NC, USA: ACM. 58, 59
- Lazik, P., Rajagopal, N., Sinopoli, B., & Rowe, A. (2015). Ultrasonic time synchronization and ranging on smartphones. In *Proceedings of the 21st IEEE real-time and embedded technology and applications symposium*. 18, 54
- Lenzen, C., Sommer, P., & Wattenhofer, R. (2009). Optimal clock synchronization in networks. In *Proceedings of the 9th ACM conference on embedded networked sensor systems*. 17
- Li, L., Xing, G., Sun, L., Huangfu, W., Zhou, R., & Zhu, H. (2011). Exploiting FM radio data system for adaptive clock calibration in sensor networks. In *Proceedings of the 9th annual international conference on mobile systems, applications, and service*. 4, 18, 19, 45
- Li, Y., Tan, R., & Yau, D. K. (2017). Natural timestamping using powerline electromagnetic radiation. In *Proceedings of the 16th international conference on information processing in sensor networks*. 18

- Li, Z., Li, C., Chen, W., Dai, J., Li, M., Li, X.-Y., & Liu, Y. (2012). Clock calibration using fluorescent lighting. In *Proceedings of the 18th annual international conference on mobile computing and networking*. Istanbul, Turkey: ACM. 18, 19, 45, 60, 70
- Lim, R., Maag, B., & Thiele, L. (2016). Time-of-flight aware time synchronization for wireless embedded systems. In *Proceedings of the 2016 international conference on embedded wireless systems and networks*. 17, 18
- Lorincz, K., Chen, B.-r., Challen, G. W., Chowdhury, A. R., Patel, S., Bonato, P., & Welsh, M. (2009). Mercury: a wearable sensor network platform for high-fidelity motion analysis. In *Proceedings of the 7th acm conference on embedded networked sensor systems*. 2, 22
- Maróti, M., Kusy, B., Simon, G., & Lédeczi, A. (2004). The flooding time synchronization protocol. In *Proceedings of the 2nd international conference on embedded networked sensor systems*. Baltimore, MD, USA: ACM. 3, 4, 20, 60, 64
- Matsushita, N., Tajima, S., Ayatsuka, Y., & Rekimoto, J. (2000). Wearable key: device for personalizing nearby environment. In *Digest of papers. fourth international symposium on wearable computers* (pp. 119–126). Atlanta, GA, USA: IEEE. 7, 8, 59
- Mayo, B. (2018). *Face ID deemed too costly to copy, Android makers target in-display fingerprint sensors instead*. Retrieved from <https://9to5mac.com/2018/03/23/face-id-premium-android-fingerprint-sensors/> 6
- Mokaya, F., Lucas, R., Noh, H. Y., & Zhang, P. (2015). Myovibe: Vibration based wearable muscle activation detection in high mobility exercises. In *Proceedings of the 2015 acm international joint conference on pervasive and ubiquitous computing*. 2, 3
- Mokaya, F., Lucas, R., Noh, H. Y., & Zhang, P. (2016). Burnout: a wearable system for unobtrusive skeletal muscle fatigue estimation. In *Proceedings of the 15th international conference on information processing in sensor networks*. 2, 3, 22
- Moss, D., & Levis, P. (2007). *Packet link layer*. Retrieved from <https://github.com/tinyos/tinyos-main/blob/master/doc/txt/tep127.txt> 63
- MSP430 LFXT1 oscillator accuracy*. (2004). Retrieved from <http://www.ti.com/lit/an/slaa225/slaa225.pdf> 45
- NTP: The network time protocol*. (2017). Retrieved from <http://www.ntp.org/> 3
- Nyemkova, E., & Shandra, Z. (2018). Fluctuations of internal electromagnetic fields as the means of electronic device authentication. In *Proceedings of the 14th international conference on perspective technologies and methods in mems design*. Lviv, Ukraine: IEEE. 59
- Park, D. G., Kim, J. K., Sung, J. B., Hwang, J. H., Hyung, C. H., & Kang, S. W. (2006). Tap: Touch-and-play. In *Proceedings of the 2006 sigchi conference on human factors in computing systems*. Montréal, Québec, Canada: ACM. 7, 8, 59
- Poh, M.-Z., McDuff, D. J., & Picard, R. W. (2010). Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Optics Express*, 18(10), 10762–10774. 58

- Poon, C. C., Zhang, Y.-T., & Bao, S.-D. (2006). A novel biometrics method to secure wireless body area sensor networks for telemedicine and m-health. *IEEE Communications Magazine*, 44(4), 73–81. 7, 9, 57, 71, 75, 84
- Prance, H., Watson, P., Prance, R., & Beardsmore-Rust, S. (2012). Position and movement sensing at metre standoff distances using ambient electric field. *Measurement Science and Technology*, 23(11), 115101. 58
- Prance, R., Beardsmore-Rust, S., Watson, P., Harland, C., & Prance, H. (2008). Remote detection of human electrophysiological signals using electric potential sensors. *Applied Physics Letters*, 93(3), 033906. 58
- Purcell, E. M., & Morin, D. J. (2013). *Electricity and magnetism* (3rd ed.). Cambridge, UK: Cambridge University Press. 14
- Rabadi, D., Tan, R., Yau, D. K., & Viswanathan, S. (2017). Taming asymmetric network delays for clock synchronization using power grid voltage. In *Proceedings of the 2017 acm on asia conference on computer and communications security*. 18, 34
- Raspberry Pi 3 Model B. (2017). Retrieved from <http://bit.ly/1WTq1N4> 21
- Reilly, J. P. (1998). *Applied bioelectricity: From electrical stimulations to electropathology*. New York: Springer New York. 14
- Roeschlin, M., Martinovic, I., & Rasmussen, K. B. (2018). Device pairing at the touch of an electrode. In *Proceedings of the 2018 network and distributed system security symposium*. Reston, VA: Internet Society. 7, 8, 59
- Rostami, M., Juels, A., & Koushanfar, F. (2013). Heart-to-heart (H2H): authentication for implanted medical devices. In *Proceedings of the 2013 acm sigsac conference on computer & communications security*. Berlin, Germany: ACM. 7, 58, 61, 62
- Rowe, A., Gupta, V., & Rajkumar, R. R. (2009). Low-power clock synchronization using electromagnetic energy radiating from ac power lines. In *Proceedings of the 7th acm conference on embedded networked sensor systems*. 4, 18, 19, 45
- Schmitt, C., Kothmayr, T., Hu, W., & Stiller, B. (2017). Two-way authentication for the internet-of-things. In *Internet of things: Novel advances and envisioned applications* (pp. 27–56). Cham: Springer International Publishing. 61
- Takiguchi, K., Wada, T., & Toyama, S. (2007). Human body detection that uses electric field by walking. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, 1(3), 294–305. 58
- Terraneo, F., Leva, A., Seva, S., Maggio, M., & Papadopoulos, A. V. (2015). Reverse flooding: Exploiting radio interference for efficient propagation delay compensation in wsn clock synchronization. In *Proceedings of the 2018 ieee real-time systems symposium*. 17
- Trippel, T., Weisse, O., Xu, W., Honeyman, P., & Fu, K. (2017). WALNUT: Waging doubt on the integrity of MEMS accelerometers with acoustic injection attacks. In *Proceedings of the 2nd ieee european symposium on security and privacy*. Paris, France: IEEE. 58
- Venkatasubramanian, K. K., Banerjee, A., & Gupta, S. K. S. (2010). Pska: Usable and secure key agreement scheme for body area networks. *IEEE Transactions*

- on *Information Technology in Biomedicine*, 14(1), 60–68. 7, 9, 58, 84
- Vistnes, A. I. (2001). Electromagnetic fields at home. In *Radiation at home, outdoors and in the workplace* (pp. 286–305). Scandinavian Science Publisher. 12
- Viswanathan, S., Tan, R., & Yau, D. K. (2016). Exploiting power grid for accurate and secure clock synchronization in industrial IoT. In *Proceedings of the 37th IEEE real-time systems symposium*. 18, 54
- Vu, T., Baid, A., Gao, S., Gruteser, M., Howard, R., Lindqvist, J., ... Walling, J. (2012). Distinguishing users with capacitive touch communication. In *Proceedings of the 18th annual international conference on mobile computing and networking*. Istanbul, Turkey: ACM. 7, 8, 59
- Wang, E. J., Lee, T.-j., Mariakakis, A., Goel, M., Gupta, S., & Patel, S. N. (2015). MagnifiSense : Inferring Device Interaction using Wrist - Worn Passive Magneto - Inductive Sensors. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing* (pp. 15–26). Osaka, Japan: ACM. 58, 59
- Wire, B. (2011). *Symmetricon announces general availability of industry's first commercially-available chip scale atomic clock*. Retrieved from <https://www.businesswire.com/news/home/20110118005182/en/Symmetricon-Announces-General-Availability-Industry%E2%80%99s-Commercially-Available-Chip> 17
- Wu, H.-Y., Rubinstein, M., Shih, E., Guttag, J., Durand, F., & Freeman, W. (2012, July). Eulerian video magnification for revealing subtle changes in the world. *ACM Transactions on Graphics*, 31(4), 65:1–65:8. 58
- Xu, F., Qin, Z., Tan, C. C., Wang, B., & Li, Q. (2011). IMDGuard: Securing implantable medical devices with the external wearable guardian. In *Proceedings of the 30th IEEE international conference on computer communications*. Shanghai, China: IEEE. 7, 57
- Xu, W., Revadigar, G., Luo, C., Bergmann, N., & Hu, W. (2016). Walkie-talkie: Motion-assisted automatic key generation for secure on-body device communication. In *Proceedings of the 15th international conference on information processing in sensor networks*. Vienna, Austria: IEEE. 58
- Yan, Z., Li, Y., Tan, R., & Huang, J. (2017a). Application-layer clock synchronization for wearables using skin electric potentials induced by powerline radiation. In *Proceedings of the 15th ACM conference on embedded network sensor systems*. Delft, Netherlands. 17
- Yan, Z., Li, Y., Tan, R., & Huang, J. (2017b). *Touchsync implementation*. 5, 45
- Yan, Z., Song, Q., Tan, R., Li, Y., & Kong, A. W. K. (2019). Towards touch-to-access device authentication using induced body electric potentials. In *Proceedings of the 25th annual international conference on mobile computing and networking*. Los Cabos, Mexico. 57
- Yan, Z., Tan, R., Li, Y., & Huang, J. (2019). Wearables clock synchronization using skin electric potentials. *IEEE Transactions on Mobile Computing*, 18(12), 2984–2998. 17

- Yang, C., & Sample, A. P. (2016). Em-id: Tag-less identification of electrical devices via electromagnetic emissions. In *Proceedings of the 2016 ieee international conference on rfid*. Orlando, FL, USA: IEEE. 59
- Yang, C. J., & Sample, A. P. (2017). Em-Comm: Touch-based communication via modulated electromagnetic emissions. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 1–24. 59
- Yang, L., Wang, W., & Zhang, Q. (2016). Secret from muscle: Enabling secure pairing with electromyography. In *Proceedings of the 14th acm conference on embedded networked sensor systems*. Stanford, CA, USA: ACM. 7, 8, 58
- Zimmerman, T. G. (1995). *Personal area networks (pan): Near-field intra-body communication* (Unpublished doctoral dissertation). Massachusetts Institute of Technology. 7, 59
- Zolertia Inc. (2018). *Z1 mote*. Retrieved from <https://github.com/Zolertia/Resources/wiki/The-Z1-mote> 63