This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing

MagneComm+: Near-Field Electromagnetic Induction Communication with Magnetometer

Guangtao Xue, *Member, IEEE,* Hao Pan, *Student Member, IEEE,* Yichao Chen, *Member, IEEE,* Xiaoyu Ji, *Member, IEEE,* and Jiadi Yu, *Member, IEEE*

Abstract—Near-field communication (NFC) technology emerges as a vital role with appealing benefits for users to improve mobile device's functionality. Although today's most smartphones and smartwatches come with NFC support, other mobile devices (*e.g.*, PC and laptops) and IoT devices that don't equip with dedicated radio modules cannot take advantage of wide-scale NFC capability. We design and develop **MagneComm+**, an NFC implementation scheme without dedicated hardware and propose a novel near-field communication protocol that is applicable to almost all mobile devices and IoT devices. The key idea is to utilize the electromagnetic induction (EMI) signal emitted from the computing devices (*e.g.*, CPUs) and captured by magnetometers on mobile devices for communication. We tackle challenges in *data encoding/decoding*, *preamble detection*, *retransmission and error correction*, *multi-transmitter*, and *full-duplex* schemes, to efficiently generate and reliably receive EMI signal with the hardware available on devices. We prototype MagneComm+ on both between laptops and smartphones, as well as between two laptops with an external magnetometer. Extensive evaluation results show that our MagneComm+ supports around $10\ cm$ communication distance with average **110 bit per second (bps)** data rate on the normal-speed mode, and maximum **17.28 kbps** on the full-speed mode.

Index Terms—near-field communication, electromagnetic induction, full-duplex.

1 Introduction

Mobile and IoT devices have taken our society by storm with their wide range of applications. Near-field communications (NFC), as a promising short-range wireless communication technology, have been attracting considerable attention in recent years. NFC facilitates applications to be more safe and convenient, including the use of wireless data transfer such as wireless keys, electronic money, electronic tickets, data documents, and mobile commerce. Therefore, NFC is a fairly common feature on most mid-range and high-end smartphones and smartwatches of today. Current NFC implementations are mainly base on the ISO-13157 standard which works with the use of a magnetic induction field, operating a radio field range of $13.56\ MHz$. Despite the safety and convenience, the NFC modules (with a built-in antenna and electric circuits for coding and decoding) are not integrated on common PCs, laptops, and most IoT devices. The need for external NFC hardware modules, however, incurs extra costs and enlarges volume to size-limited IoT devices and therefore greatly limits the applicability of NFC technology.

Several alternative methods have been devised to alleviate the pain brought by the standard NFC implementation. Methods based on the existing wireless antenna modules (*e.g.*, Bluetooth and Wi-Fi) in the smartphones may face some problems. First, they are vulnerable to security issues. For example, Bluetooth works in a relatively longer communication range than the standard NFC and it is easy for attackers to eavesdrop the transmitted information. Second, the communication channels endure noise and interference, *e.g.*, Bluetooth is easily interfered with Wi-Fi signals. Some fundamental approaches based on acoustic signal [1], [2], visible light [3], [4] are also the alternatives to achieve similar functions for short-distance wireless transmission of data. These mentioned approaches rely on the built-in multimedia sensors (*e.g.*, microphone or camera), which may not be equipped in some mobile devices like smartwatches. Meanwhile, they also suffer from the security issues. Therefore, existing alternative NFC implementations fail to satisfy both the security and performance requirements.

To meet the security and performance considerations, we present an NFC implementation scheme without dedicated hardware in this paper. Using electromagnetic induction (EMI) signal emitted from computing devices as a communication channel and the prevalent build-in magnetometer for signal reception, we show that NFC implementation based on EMI signal can completely abandon dedicated hardware design. Specifically, we elaborately control the CPU modules for transmission of EMI signals, and the magnetometers equipped on commodity mobile devices for signal decoding. By carefully regulating the EMI signals emitted from CPUs, and sensing it with the magnetometers on devices, two devices are able to communicate in a near-field manner. We develop this concept into a novel near-field communication system (referred to as MagneComm+) based on the EMI side-channel. In a normal-speed mode, MagneComm+ is designed using techniques such as proactive retransmission, multi-transmitter technology, and self-signal cancellation for full-duplex enabled communication. However, the above-mentioned mode extremely relies heavily on the proactive retransmission module based on real-

- *G. Xue, H. Pan, Y. Chen and J. Yu are with the College of Computer Science and Technology, Shanghai Jiao Tong University, Shanghai, CN. E-mail: gt_xue, panh09, yichao, jiadiyu@sjtu.edu.cn*
- *X. Ji is with the School of Electronic Information and Electrical Engineering, Zhejiang University, Hangzhou, CN. E-mail: xji@zju.edu.cn*

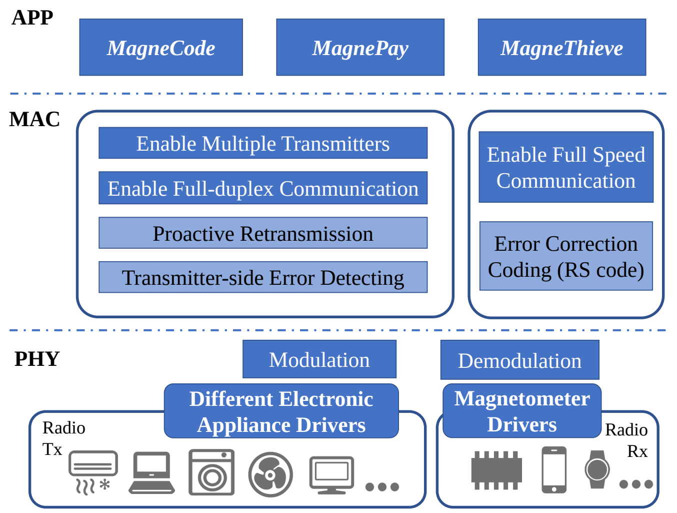

APP
MagneCode
MagnePay
MagneThieve
MAC
Enable Multiple Transmitters
Enable Full-duplex Communication
Proactive Retransmission
Transmitter-side Error Detecting
Enable Full Speed Communication
Error Correction Coding (RS code)
PHY
Modulation
Demodulation
Different Electronic Appliance Drivers
Magnetometer Drivers
Radio Tx
Radio Rx

Fig. 1. Overall architecture of MagneComm+.

time CPU core usage information to guarantee the communication quality, and the accuracy of CPU usage acquisition limits the throughput of our system. To cope with application scenarios requiring high near-field communication rates, we propose a full-speed mode based on an error correction method. The overall architecture of MagneComm+ is shown in Fig. 1 from a layered perceptive.

The design of MagneComm+ poses mainly the following three challenges. (i). Precision of controlling the CPU's working status via programming to generate EMI signal may be limited by the job scheduling mechanism on the mobile operating systems (OSes). With this constraint, frequency-based modulation schemes, such as FSK (frequency shift keying) and OFDM (orthogonal frequency division multiplexing) are essentially infeasible and not efficient. Therefore, we applied the pulse width & amplitude modulation (PWAM) as the fundamental modulation scheme to strike a balance between transmission speed and accuracy. (ii). Multi-task mechanism of OS determines that the concurrent running jobs on CPUs will bring about interference during the transmission process, which increases confusion and brings difficulties when receiving CPU's useful EMI signals. To guarantee the reliability of transmission in noisy environments, we first design a preamble pattern for signal detection. After that, in the normal-speed mode where users can normally use the transmitter device, a retransmission scheme is designed to monitors the real-time CPU usage and retransmits the previous corrupted packet if necessary; in the full-speed mode with all user's running tasks disabled, an error correction mechanism is utilized to handle with the random error bits caused by the OS's background running tasks during the transmission. (iii). The sampling rate of the magnetometers embedded in the mobile devices is too low to limit the performance of the entire communication system. Multiple transmitters scheme is designed in order to improve the throughput in the normal-speed mode, with the rate of EMI change of one CPU core at the transmitter matching the sampling rate of magnetometer at the receiver. In the case that both the transmitter and receiver are equipped with magnetometers, the signal cancellation technique with a subtraction algorithm is utilized to achieve full-duplex communication on the normal-speed mode.

We implement the MagneComm+ prototype and evaluate the performance between laptops and smartphones, as well as between laptops and external magnetic sensors. The encoded EMI signals can be successfully captured and decoded by the magnetometers. In our experiments, MagneComm+ under the normal-speed mode is able to achieve throughput of up to 110 bps in average between a laptop and an external magnetic sensor, 33 bps between a laptop and a smartphone. MagneComm+ under the full-speed mode can achieve throughput of up to 17.28 $kbps$ between a laptop and an external magnetic sensor with a high sampling rate. Our prototype supports a communication range of up to 10 cm between devices. In the last, we develop abundant applications upon the proposed MagneComm+ technologies.

Our main contribution can be summarized as follows:

- We propose MagneComm+, a novel implementation scheme of NFC which eliminates the need for dedicated antenna and codec chip by embedding data stream into the EMI signals of a device without affecting the normal function.
- A variety of communication schemes are devised using this system, including one-way communication (in normal-/full-speed modes), full-duplex and multiple-transmitter to take full advantage of the functionality of the communication devices.
- We build two prototypes of the proposed MagneComm+, and conducted expensive experiments to demonstrate the efficacy of our system: it can achieve throughput of 110 bps/17.28 $kbps$ over a communication distance of up to 10 cm on the normal-/full-speed modes respectively. The source code is available at: https://github.com/SolskyPan/MagneComm.

2 Background

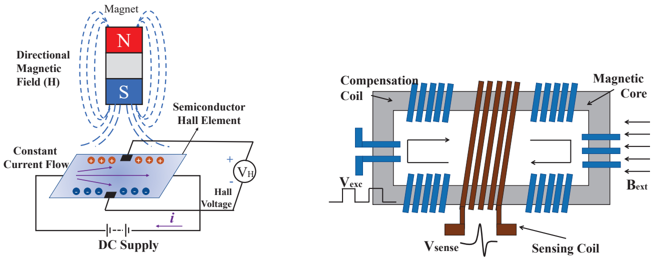
We introduce the working principle of the integrated magnetometers and investigate the EMI signals from computing devices (*e.g.*, laptops) in this section. These are the fundamental issues related to the design of MagneComm+.

2.1 Magnetometer

A magnetometer is an instrument that measures the strength and/or direction of the magnetic field at a point in space. The most common types of magnetic sensors are hall effect sensors and fluxgate sensors. Hall effect sensors are small and cheap while the sensitivity is low ($\leq 5mV/mT$). They are commonly employed for electric compass functions in mobile devices. As illustrated in Fig. 2(a), the voltage develops in a direction transverse to the current flow in p-type semiconductor in a magnetic field owing to Lorentz Force [5]. Hall effect magnetometers utilize the varying output voltage as the indicator to the magnetic field.

Fluxgate magnetometers provide greater sensitivity than hall effect sensors. They are commonly used to measure DC or high-frequency magnetic field vectors. As shown in Fig. 2(b), a fluxgate sensor employs a saturable inductor to sense the magnetic field produced by an external source.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing



(a) Hall effect sensor.

(b) Fluxgate sensor.

Fig. 2. The working principle of two types of magnetometers.

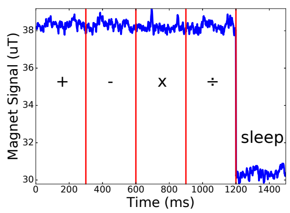


Fig. 4. Emitted EMI signals (with $40\ kHz$ sampling rate) while CPU executing various instructions.

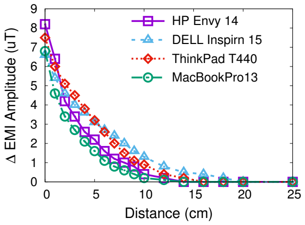
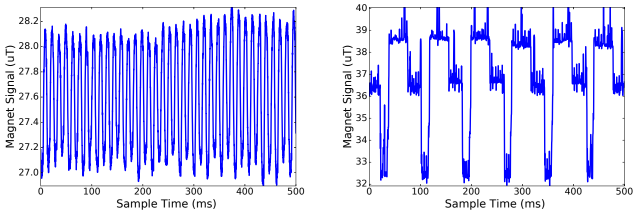


Fig. 5. The amplitude change of EMI signals of one CPU core at different distances.



(a) Electronic fan.

(b) CPU.

Fig. 3. EMI signals emitted from a CPU and an electronic fan.

The fluxgate sensor is repeatedly driven in and out of saturation, thereby enabling hysteresis-free operation with a high degree of accuracy. The internal compensation coil ensures stable gain and high linearity.

In this section, instead of using built-in hall effect magnetometers in smartphones, we employ a high-performance fluxgate sensor, DRV425 from Texas Instruments [6] with high-accuracy ($\pm 0.1\%$ linear sensitivity) sensing range of $\pm 2\ mT$ and a measurement bandwidth of up to $47\ kHz$, to understand the characteristic of EMI signals emitted from the mobile devices. The output analog signals from DRV425 are sampled by a USB digital oscilloscope measurement device – Analog Discovery 2 (called AD2) [7]. And we set the sampling rate of AD2 at $40 kHz$ with 14-bit output resolution to digitize the collected magnetic field signals.

2.2 Electromagnetic Induction from Laptops

A laptop is a sophisticated device with numerous electronic units, such as a CPU, GPU, electronic fan, speaker, hard disk, battery and etc. As the laptop executes one task, an electric current flows through each electronic units on the motherboard and generates EMI signals theoretically. The electronic units that directly related to the CPU (*e.g.*, south/north bridge and RAM) emit the similar EMI pattern with the CPU when the laptop is executing one task. While others emit inconvenient and uncharacteristic signal [8]. By carefully controlling the working pattern of these units, we can embed a data stream in the generated EMI signals. As shown in Fig. 3(b), the strength of EMI signals can be adjusted by controlling the calculation load of the CPU. Fig. 3(a) presents EMI signals generated by a computer fan in the form of sine waves. Controlling the rotation speed of the fan to vary the phases and frequencies of these sine waves also makes it possible to embed information in the EMI signals.

Despite the multiple choices, we focus primarily on the EMI signals generated by CPUs in this work. The most direct advantage of using CPUs is that the CPU is a must for computing devices (*e.g.*, laptops, smartphones, and smart watches). Another advantage is that CPU usage can be easily controlled into working and idle states via running loop and `sleep()`-like commands, which are available for the whole programming languages and operating systems. Furthermore, it is easier to monitor CPU usage in real-time in order to estimate channel conditions. Finally, the multiple cores in a CPU can be used to design a multiple-transmitter communication protocol. The details of the protocols are discussed in Sec. 4.

3 Electromagnetic Induction from CPU

The precise control over CPUs and their emitted EMI signals are the key to realize MagneComm+. In this section, we investigate the characteristics of the EMI signals from CPUs and explore the design space of modulating data bits.

3.1 Characteristics

EMI signals generated from the working CPUs are affected by many factors.

Multi-core CPU: Computing devices can increased the performance of a CPU and run multiple instructions in parallel efficiently by using multi-core technologies. Unless otherwise specified, a program with multiple threads may attempt to use the CPU cores assigned by the operating system. The fact that each core emits its own EMI signals means that the EMI signals from a CPU is actual the superposition of all of the cores' EMI signals.

CPU working status: different instructions can lead to different EMI patterns, which can be identified by a customized antenna with an extremely high sampling frequency sampling rate ($> 1.7\ MHz$) [9]. The sensing bandwidth of most integrated magnetic sensor ICs on the market ranges from $50\ Hz$ to $50\ kHz$. Based on experiment results, we can only determine whether a CPU is working or idle, but little else. As shown in Fig. 4, we run a program on the same CPU core, including four instructions: addition (+), subtraction (-), multiplication (×), division (÷), and then a `sleep()` command for doing nothing. Omitting the glitch signal leaves only two levels of magnitude in the EMI signals, corresponding to the status of the CPU core.

Distance: the attenuation effects of the EMI signal emitted from CPUs were exponential with the distance [10]. Furthermore, the constructed material of the mobile devices may also affect the magnitude, due to the shielding effect.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing

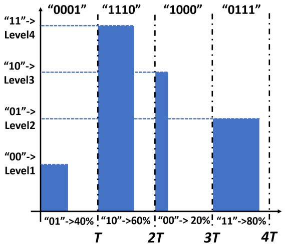
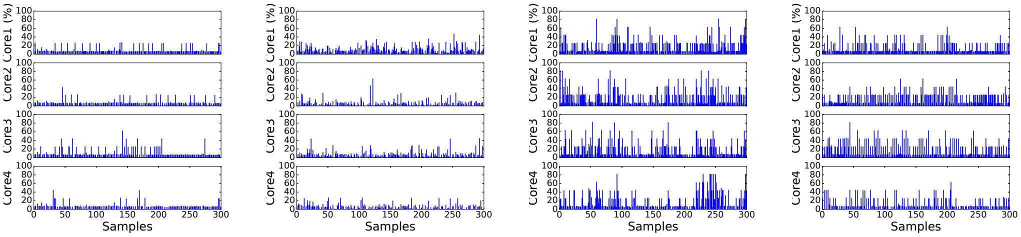


Fig. 6. Example of pulse width and amplitude modulation data symbols.



(a) Doing Nothing. (b) Watching Video. (c) Surfing Websites. (d) Playing Games.

Fig. 7. CPU usage of each core recorded every $50ms$ when the user using the laptop under four different conditions.

Fig. 5 shows the changes in the amplitude of EMI signals over distance when using various types of laptops. At each distance, we measure EMI signals emitted from one CPU core in both working and idle states at a time, and calculate the amplitude difference between two states. We can see that at distances exceeding 10 cm, the magnitude change becomes too small to be distinguished by the DRV425.

3.2 Controlling Electromagnetic Induction Signals

In the following, we examine the methods used to control EMI signals from the CPU and modulate the data.

Modulation: commercial integrated magnetic sensors can only measure the working and idle status of a CPU core; therefore, one intuitive way to embed data is to use On-Off Keying (OOK). That is, we switch a CPU core between idle and working to represent 0 and 1, respectively. Obviously, reducing the switching period will correspondingly increase the data transmission rate. However, in most practical situations, the period and precision of controlling CPU's states are limited by the OSes.

We therefore selected Pulse Width and Amplitude Modulation (PWAM) instead of the OOK to increase the data rate with the limited controlling period. Specifically, each data symbol has period T and is presented using the proportion of CPU working time during that period. A data symbol consists of $M+N$ data bits, where M bits are translated to a set of pre-defined 2^M levels of amplitude of CPU magnetic induction signals (PAM), N bits are translated to a set of pre-defined 2^N levels of CPU working percentage (PWM). Take Fig. 6 as an example, each data symbol is 30 ms long and consists of 4 bits. The amplitude of each symbol could be $1, 4, 3, 2$ which represents data bits: 00, 11, 10 and 01. The CPU working percentage of each symbol could be 40%, 60%, 20%, 80% which represents bits: 01, 10, 01 and 11, respectively. We can adjust T, M and N according to the current channel condition as well as how precise we can control a CPU core.

Processor affinity: modern operating systems (OSes) optimize the scheduling of tasks specifically for multi-core CPUs to ensure efficiency in the execution of commands. The OS maintains queues of global tasks, which can be assigned to any core with free resources. In controlling specific CPU cores to generate a desired magnetic signal, processor affinity [11] can be used to bind a given process to a specific CPU core. As shown in Fig. 3(b), multiple cores can be used to generate the same PWAM symbols in order to increase transmission range, or generate different PWAM symbols for parallel communication.

3.3 Noise from Other Running Programs

It is inevitable that the CPU will be used by the user and/or the OS during the transmission of data. CPU usage from other programs introduces noise to the EMI signals used for communication. We seek to characterize this noise by collecting traces of CPU core usage while a user performs various activities, including watching online streaming content, surfing websites, playing games, as well as doing nothing. Fig. 7 shows the corresponding CPU usage on each core. We also observed that watching videos and streaming only occasionally increase the CPU usage, whereas surfing websites and playing games greatly affect CPU usage. It is also observed that when the device is not used, the OS continues accessing the CPU for brief periods in the processing of background apps. Therefore, even for CPU intensive operations, there are numerous intervals of low CPU usage, which could be used for communication. The above observation indicates that a retransmission mechanism with the detection of packet corruption and/or error correction mechanism are required due to the interference.

4 MagneComm+ Design

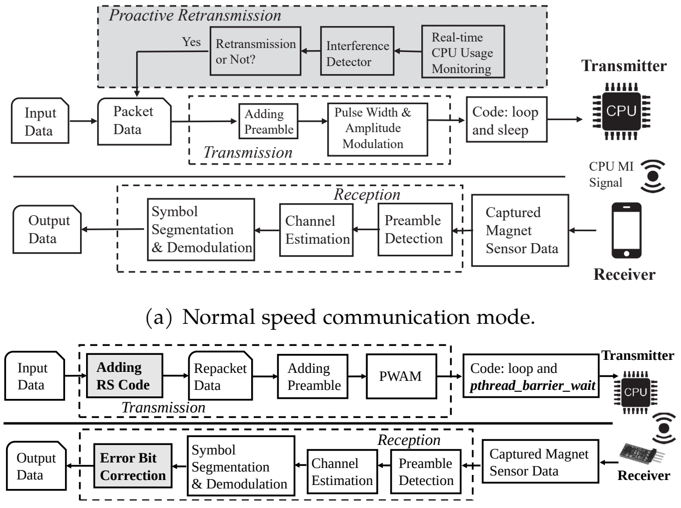
4.1 System Overview

We design a near-filed electromagnetic induction communication system called MagneComm+, that is an NFC implementation protocol for the off-the-shelf mobile devices without adding any dedicate antennas and codec chips. Our MagneComm+ consists two communication codes: normal-speed mode (see Fig. 8(a)) is designed for the scenarios that users are allowed to normally use the transmitter devices; full-speed mode (see Fig. 8(b)) is designed for the scenarios that requires extreme communication rate. Our MagneComm+ system architecture consists of two major parts: a transmitter (*e.g.*, CPU) and a receiver (*e.g.*, magnetometer). The transmitter embeds data with the EMI signals generated by CPU cores, while the magnetometer captures the corresponding magnetic field signals for the further transmission data extraction.

4.2 Transmitter and Receiver Design

4.2.1 Transmitter Design

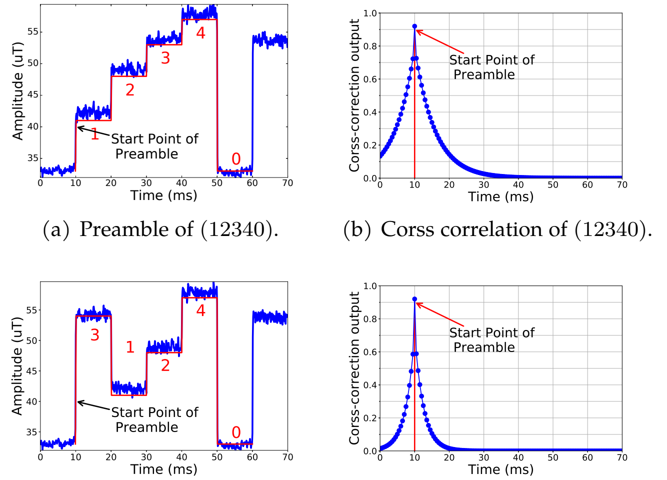
Preamble Design: Preamble is used for two purposes. One is to synchronize the transmitter and the receiver. An



(a) Normal speed communication mode.

(b) Full speed communication mode.

Fig. 8. MAGNECOMM+ operation flow in different communication modes.



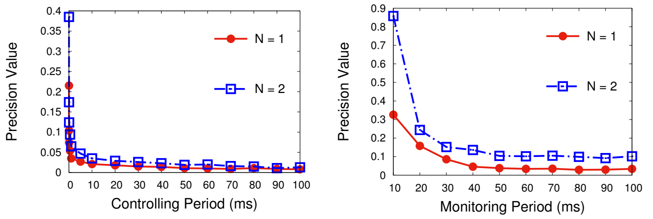
(a) Preamble of (12340).

(b) Corss correlation of (12340).

(c) Preamble of (31240).

(d) Corss correlation of (31240).

Fig. 9. Preamble pattern design when using four CPU cores with corresponding cross correlations of preamble detection.



(a) CPU Controlling Error.

(b) CPU Monitoring Error.

Fig. 10. The precision of CPU controlling (with two programming methods) and CPU usage monitoring.

unique EMI pattern is introduced as the preamble at the beginning of each data frame, thereby allowing the receiver to use cross-correlation to identify the start time of each transmission. The other purpose is to enable the receiver to estimate communication channel conditions. Because the amplitude of EMI signals varies when different numbers of CPU cores are being used as well as under different channel conditions. The receiver must be aware of different amplitude values of a PWAM symbol in order to decode it.

When using four CPU cores as transmitter whose communication parameter M sets as 2, the preamble can be designed as a permutation sequence of four different amplitude levels with a end mark $[0]$. Different preamble patterns have different performance of detection rates. We takes two permutations: $(1, 2, 3, 4, 0)$ and $(3, 1, 2, 4, 0)$ as examples, their corresponding EMI signal patterns and preamble detection cross correlations are shown in Fig. 9. From the cross correlation results, we select the $(3, 1, 2, 4, 0)$ (see Fig. 9(c)) as the preamble pattern in the condition of four CPU cores.

Modulation: We opted for PWAM for the modulation of data bits due to its effectiveness over OOK. The choice of T, M and N depends on the precision of CPU controlling and monitoring.

The control precision is defined as: $Prec_{control} = \frac{|U_{real}-U_{desired}|}{[1/(2^N-1)]}$, where $U_{desired}$ is the desired CPU core usage, U_{real} is the actual CPU core usage generated, and $1/(2^N-1)$ is the interval between 2^N levels (*e.g.*, when $N = 2$, the interval between 4 levels of CPU core usage is 33%). The closer the value of $Prec_{control}$ is to 0, the more preciously we can control the CPU cores. To implement the CPU control, we used the thread barrier objects (in the *Pthread* library[1]) with `pthread_barrier_wait()` to synchronize the initiation and termination of the CPU's working states. Fig. 10(a) shows the single CPU core control precision under various T and N, and experimental results shows that the smaller T and larger N reduces CPU controlling precision. The minimal symbol length T we can choose is $0.05ms$ with $N = 1$ bit per symbol. Note that the precision of `sleep()` is not sufficient for the small controlling period, and the minimal symbol length T can only be set as $10ms$ with $N = 1$ bit per symbol [12].

Considering that the transmitter device will inevitably execute other tasks from users or OS itself, and generate the interference EMI signals. The transmitter must continuously monitor CPU usage to check that packets are transmitted correctly or not. Thus, the precision with which the CPU core usage is monitored also limits the selection of T and N. We use EMI measurements as a proxy for CPU usage and define monitoring precision in a similar manner: $Prec_{monitor} = \frac{|U_{sensor}-U_{monitor}|}{[1/(2^N-1)]}$, where U_{sensor} is the CPU core usage estimated using measured EMI signal by sensor, and $U_{monitor}$ is the monitored CPU core usage by OS itself. The results are shown in Fig. 10(b). We can see that the minimal symbol length can be correctly monitored is $20ms$. So after combining the limitation of CPU controlling and monitoring, we will select $T = 20ms$, $N = 2$ as our fastest transmission rate to modulate the information data bits with the correct CPU usage monitoring.

[1] *Pthread*, as a set of general multi-thread library proposed by POSIX, can be supported on the Unix-like OSes (*e.g.*, Ubuntu, MacOS and Android). And the windows OS also has its ported version – pthread-win32.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing

4.2.2 *Receiver Design*

Preamble Detection: The receiver applies cross-correlation to locate the preamble pattern accurately for use in synchronization, and performs channel estimation to determine the levels of EMI amplitude prior to extract symbols.

Symbol Segmentation: Then the receiver can segment the symbols after preamble with the fixed PWAM length known in advance. After channel estimation, the receiver determines the amplitude and width information in each symbol, whereupon the original data bits can be demodulated.

4.3 Normal-speed Communication Mode

Normal-speed mode in MagneComm+ is designed for the communication scenarios with small data rate, while allowing the users to normally use programs on the transmitter devices. In this part, we will introduce the techniques, such as proactive retransmission, multi-transmitter and self-signal cancellation (for enabled full-duplex), that are be utilized to enhance the communication performance.

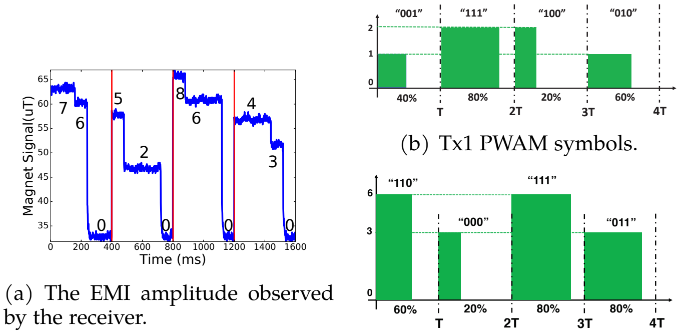
4.3.1 *Proactive Retransmission*

In the design of MagneComm+, the transmitter is not able to get acknowledgement from the receiver to check if a packet is successfully transmitted. Therefore, a corruption detection mechanism on transmitter is needed for the communication reliability. Fortunately, due to the near-field communication property, the major noise is actually generated by background running programs. By continuously monitoring the usage value on all CPU cores, the transmitter can know if a packet is corrupted due to the noise. Note that the precision of CPU usage monitoring limits the PWAM parameters selection (T and N), which further limits the transmission throughput of the MagneComm+.

Algorithm 1 Proactive retransmission on transmitter.

```
function RETRANSMISSION
    while packet $x$ is sent do
        $SymbolNoise_j \leftarrow \sum_{core\ i} Usage_i$
        $ReFlag_x \leftarrow \sum_{symbol\ j} Rule(SymbolNoise[j])$
        if $ReFlag_x \neq 0$ then
            Retransmit(packet $x$)
        else
            Transmit(packet $x + 1$)
        end if
    end while
end function
```

Algorithm 1 shows our Proactive Retransmission mechanism. The transmitter controller monitors the real-time CPU usage on all cores with the same period of PWAM symbols, and after transmitting one packet, it uses the history experience to determine whether existed interfered symbols in the packet. For example, when over two non-transmitter CPU cores have more than 20% usage, the transmitter controller classifies this symbol into the interfered category and $Rule(SymbolNoise[j])$ would be set as 1. Because we did not employ error correction schemes, the transmitter retransmits the whole packet for any corrupted symbols.



(a) The EMI amplitude observed by the receiver.

(b) Tx1 PWAM symbols.

(c) Tx2 PWAM symbols.

Fig. 11. The working principle for multiple transmitters: an example that uses eight CPU cores to serve as two transmitters.

4.3.2 *Enabling Multiple Transmitters*

In wireless communications, it is well known that multi-antenna systems perform better than single-antenna systems by increasing transmission bandwidth and effectively resisting multipath and interference. Our system MagneComm+ imitate this mechanism and use multi-core CPU to serve as multiple transmitters. For example, in a laptop with the eight-core CPU, one core group consisting of two cores can be treated as the first transmitter, while the other core group consisting of six cores can be treated as the second transmitter. Two core groups can send data concurrently to increase the throughput.

In the transmitter, we rely on process affinity for the generation of PWAM symbols using different CPU cores. However, in the receiver, we need a mechanism to distinguish PWAM symbols from different transmitters. Our solution is based on a simple idea: each transmitter uses different number of CPU cores so the amplitude values of PWAM symbols from different transmitters are different. In the previous example, we allocate two cores to Transmitter one with $M = 2$, so it can generate EMI signals with amplitudes $[0, 1, 2]$; we allocate six cores to Transmitter two with $M = 2$, so it generates $[0, 3, 6]$ amplitudes. And the superposition of amplitude values are shown in Table. 1. The mixed EMI amplitude corresponds to only one pair: $Tx1$ signal, $Tx2$ signal. So the receiver can easily to separate the mixed signals into two signals from Table. 1. Fig. 11 shows a two-transmitter example. When two transmitters are transmitting data bits simultaneously, the first transmitter uses two CPU cores to generate PWAM symbols, with the parameters are: $T = 300ms$, $M = 1$ and $N = 2$, and the second transmitter uses six CPU cores with the same parameters. When the receiver receive PWAM symbol, it detects the amplitude level of each period using the preamble signal as reference, and in this condition the preamble can be designed as the pattern like, $(1, 2, 3, 4, 5, 6, 7, 8, 0)$. After obtaining the amplitude levels of each symbol, the receiver checks Table. 1 and rebuilds $Tx1$ and $Tx2$ signals to extract data bits separately.

4.3.3 *Enabling Full Duplex Communication*

Full-duplex communication can be achieved when both mobile devices are equipped with magnetometers (*i.e.*, communication between two mobile phones.) The challenge of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing

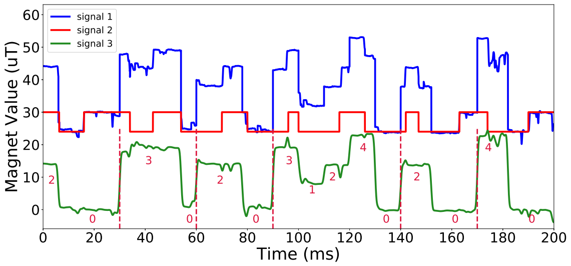


Fig. 12. The cancellation principle of full-duplex communication: signal 1 represents the mixed EMI signals, signal 2 represents the estimated EMI signals generated by itself, and signal 3 represents the targeted EMI signals.

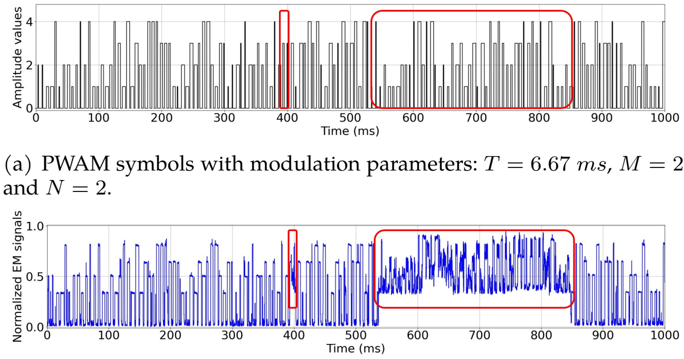
TABLE 1
EMI amplitudes observed by receivers when two transmitters are transmitting concurrently using different number of cores.

Mixed EMI Signal	0	1	2	3	4	5	6	7	8
Tx 1 EMI Signal	0	1	2	0	1	2	0	1	2
Tx 2 EMI Signal	0	0	0	3	3	3	6	6	6

enabling full-duplex communication is that, when a mobile device is transmitting and receiving data at the same time, the received PWAM symbol is a linear combination of its own transmitted PWAM symbol and targeted PWAM symbol to decode. Fortunately, since the device knows what it is transmitting and has ability to estimate the amplitude of its own PWAM symbol, the device can subtract the transmitted PWAM symbol from the received one and recover the the targeted PWAM symbol. Fig. 12 shows an example to perform full-duplex communication. Considering that the magnetometers equipped on the smartphones are not sensitive enough to capture the other smartphone CPU's EMI signals, we use two laptops both equipped with DRV425 magnet fluxgate sensors to verify the feasibility of our full-duplex design. Specifically, one laptop sets the transmission parameters to: $T = 30ms$, $M = 0$, $N = 2$, while the other sets it to: $T = 30ms$, $M = 2$, $N = 2$.

4.4 Full-speed Communication Mode

Considering the communication scenarios that require high data rates, we design a full-speed communication mode in which the user should disable all other user's tasks running. From the CPU controlling experimental results (see Fig. 10(a)), we use four CPU core as a transmitter to generate *PWAM* symbols ($M = 2$, $N = 2$) and set the windows length T of each symbol as $6.67ms$, we use an external magnetic sensor (with $40\ kHz$ sampling rate) to collect the EMI signals and show them in Fig. 13. After comparing the original modulated PWAM symbol patterns (Fig. 13(a)) and the received EMI signals generated from the transmission device's CPU (Fig. 13(b)) there still exists bit errors (see red boxes in the Fig. 13) caused by the background running tasks derived from the OS itself and/or services registered by the third-party applications. Another key point is that although small T can boost the throughput, too small T will also distort the CPU's EMI signal patterns thus increasing the error of decoding bits in the receiver, therefore there is a balance between the window length and the communication rate.



(a) PWAM symbols with modulation parameters: $T = 6.67\ ms$, $M = 2$, and $N = 2$.

(b) The EMI signals received by the external magnetoemter (DRV425+AD2).

Fig. 13. An example of error bit distribution caused by the background running tasks/services during the high-speed communication.

4.4.1 Error Bit Correcting

The proactive retransmission in Sec. 4.3.1 cannot work in the full speed communication mode because the CPU usage monitoring method will generate large error with the small monitoring period (see Fig. 10(b)). Fortunately, we find that the error bit distribution caused by OS's tasks is almost unchanged over time. Therefore, we adopted an error-correcting mechanism (Reed-Solomon coding [13]) commonly used in wireless communication to deal with the random EMI channel errors. Reed-Solomon (RS) coding requires a deterministic input for hard-decision decoding, which means that we need to determine the RS code parameters (R, K) in advance, where R indicates the transmission data bit rate and K indicates the additional redundant input data bit rate for each package capacity, which are stored on both the transmitter and receiver sides. The process of determining the suitable PWAM parameters (especially, window length T) and RS code parameters (R, K) for the full speed communication mode is detailed in Sec. 6.1.2.

5 System Prototype

Two prototypes are implemented to evaluate MagneComm+.

Prototype I: Laptop and Smartphone. The first prototype employs transmitters on laptops running Windows, Ubuntu, or MacOS X, while the receiver is an Android phone. Laptops are equipped with Intel Core CPU (4 or 8 cores). The Android phone is a Huawei P20Pro equipped with a Hall effect magnetometer operating at a sampling rate of $100Hz$. In this prototype, we implement one-way communication with multiple-transmitter enabled. All experiments in Section 6 are repeated using all laptops.

Prototype II: Laptops with Magnetometers. The second prototype employs transmitters and receivers on the laptops with external magnetometers (DRV425) attached. In this system, we used the same set of laptops that were used in the first prototype. The magnetometer's output is sampled with a AD2 sampling device whose sampling rate is set as $40\ kHz$. This prototype involves in implementing the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing

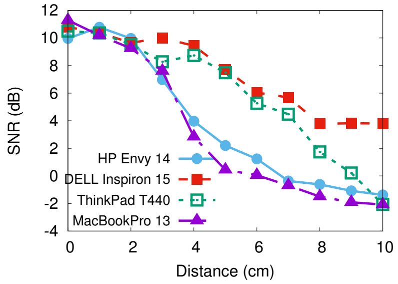


Fig. 14. SNR vs. distance using different models of laptops.

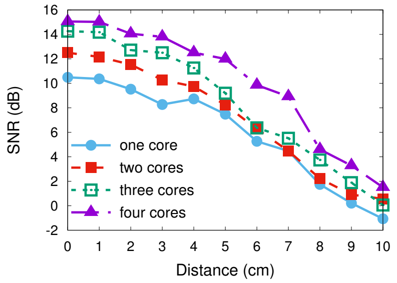


Fig. 15. SNR vs. number of working CPU cores.

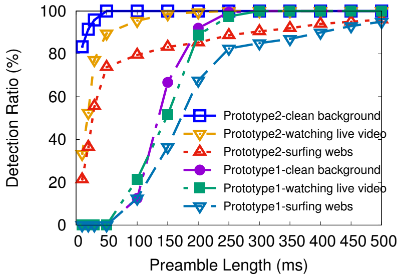


Fig. 16. Preamble detection accuracy under various preamble lengths with $M = 2$.

multiple-transmitter and full-duplex communication functions in the normal-speed mode, and the full-speed one-way communication mode.

6 Evaluation

6.1 Micro Benchmark

Communication Distance : we first use Prototype II where two laptops equipped with external magnetometers to examine the valid communication ranges. Fig. 14 shows the SNR under various transmission distances using four different laptops. We can observe that SNR is similar for all laptops when distance is shorter than 3 cm. SNR for HP Envy 14 and MacBookPro 13 drops quickly after 3 cm while that from DELL Inspiron 15 remains high at 10 cm. We can boost the transmission distance by utilizing multiple CPU cores. As shown in Fig. 15, the more CPUs used, the higher SNR we can get. At the distance of 5 cm, increasing the number of CPU cores from 1 to 4 increases SNR by 4 dB.

Device Orientation : to verify the impact of the device orientation to the communication, we change the angles of the receiver by $15°$, $30°$, $45°$, $60°$, and $75°$ while being placed at the same location. The results show that the device orientation has no influence to the observed EMI signals. It's because that there are three orthogonal magnet sensors on a magnetometer and we calculate the square root of sum of the magnetism from three dimensions so it becomes independent to the device orientation.

Preamble Design : designing a good preamble is crucial to the MagneComm+ system, specifically allows the receiver to detect the preamble efficiently and stably. A longer preamble makes it easier to control CPU cores to generate the desired preamble pattern and is more robust against the noise. However, the longer preamble also results in the larger communication overhead. To obtain the best preamble length for MagneComm+ while background noise exists, we vary the preamble length and use Prototype I and II to test the ratio where preambles are correctly detected. Fig. 16 shows the results. We can see that $80\% - 100\%$ of preambles can be correctly detected when the length is 300 ms in Prototype I, and 50 ms in Prototype II.

6.1.1 *Normal speed communication mode*

PWAM parameters : In Sec. 4.2 we show that the symbol length T needs to be 20 ms or longer in order to precisely control and monitor the CPU usage. Nonetheless, channel condition also has impact to the selection of T. A shorter symbol period implies a higher data rate while it can also be more fragile to the noise. To find the best parameters for PWAM modulation, we vary T, M, N under different levels of noise using Prototype II. The level of noise is defined as the CPU usage caused by background programs. Each experiment lasts for 10 minutes. The corresponding throughputs are shown in Fig. 17. We can observe that when the noise is small, using shorter symbol period T yields a larger throughput, as expected. When the noise value is 40% or larger, lower data rates with larger T start to out-perform because they can tolerate more noise.

Corruption Detection from Transmitter: the success of one-way communication relies on that if a transmitter can correctly detect and retransmit corrupted packets. We transmit PWAM symbols with T varying from 20 to 500 ms and $N = 2$ for 10 minutes in Prototype II, and evaluate the corruption detection accuracy.

The results are shown in Fig. 18. True Positive Rate (TPR) represents the number of symbols which are actually corrupted and are detected as corrupted to the number of actually corrupted symbol. False Positive Rate (FPR) represents the number of symbols which are correct but are detected as corrupted to the number of actually correct symbol. False Negative Rate (FNR) represents the number of symbols which are actually corrupted but are detected as correct symbol to the number of actually corrupted symbols. We can see that TPR is 99.6% or higher while FNR is 0.5% or lower when the symbol length is 30 ms or longer, which indicates that our corruption detection algorithm on transmitter can correctly detect corrupted symbols. Therefore, in our following experiments, we select the 30 ms as the fastest transmission rate instead of 20 ms.

6.1.2 *Full speed communication mode*

PWAM parameters: We utilized the Prototype II as transmitter and an external magnetometer as receiver and fix the distance between the transmitter and receiver as $3cm$, then we vary the PWAM parameters $T(ms), M, N$ and modulate the transmission data without an error correction scheme on the transmitter and demodulate these data on the receiver. We only focus on the parts that are not disturbed by background running tasks and/or services. Each experiment lasts for 10 minutes. The corresponding experimental results are shown in Fig. 19. Raw throughput accounts for all transmitted data bits, and data goodput means the correctly decoded data bits on the receiver. Therefore, according to

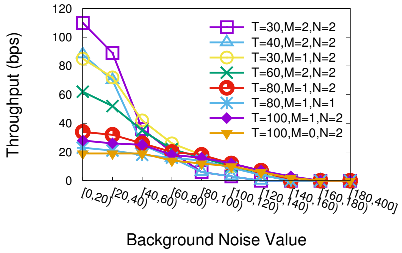


Fig. 17. Throughput with various PWAM parameters under different levels of noise.

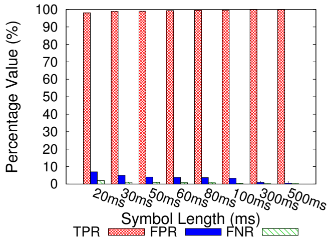


Fig. 18. Corruption detection performance with different symbol lengths.

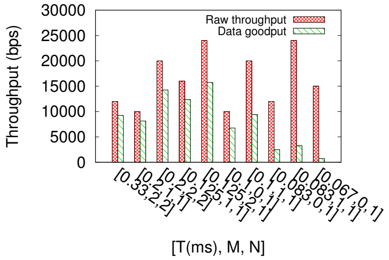
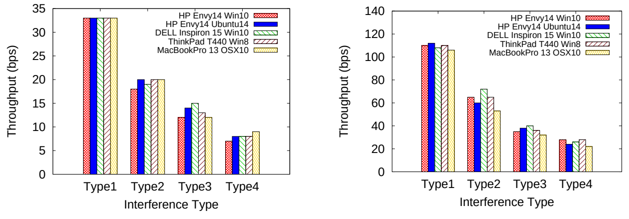


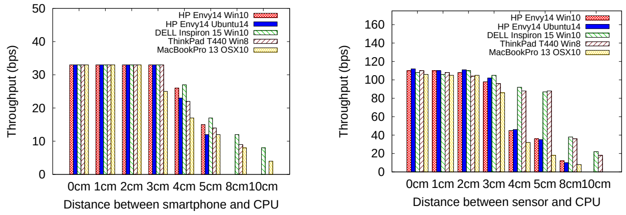
Fig. 19. Determining the PWAM parameters of the full speed communication mode.



(a) Prototype I.

(b) Prototype II.

Fig. 20. Throughput under various scenarios for one-way communication. In Prototype I, $T = 300ms, M = N = 2$. In Prototype II, $T = 30ms, M = N = 2$. All laptops use 4 CPU cores for transmission. Scenario type 1 to 4 correspond to that the user is doing nothing, watching live video, surfing websites, and playing games, respectively.



(a) Prototype I.

(b) Prototype II.

Fig. 21. Throughput under various distances for one-way communication. In Prototype I, $T = 300ms, M = N = 2$. In Prototype II, $T = 30ms, M = N = 2$. All laptops use 4 CPU cores for transmission.

the maximum data goodput, we select $[0.125, 2, 1]$ as the PWAM parameters for the fastest transmission rate.

RS coding parameters: To determine the RS coding parameters, we should first understand the proportion of error bits in the communication process. Error bit ratio caused by the disturbed symbol patterns is mainly accounted for the background running tasks and/or services on the transmission devices. One example of the error bit distribution tested on the HP Envy 14-j 104TX with Windows 10 has been shown in Fig. 13(b), we found that in the whole communication process of 1000 ms, the symbol patterns in the 300 ms time period were disturbed, so the bit error ratio was about 30%. Therefore, when we use RS code encoding [14] to add the error correcting code (ECC) symbols for each package, we can set the parameters of $[R, K]$ as $R = 255$ (symbols) and $K = 156$ (bytes) to correct max 78 error symbols when transmitting 255 symbols. Considering that different transmission devices have different error bit distributions, we transmitted a specific bit stream with the target transmission device and calculated the error bit ratio to determine the most suitable RS coding parameter. We also tested the error bit ratio on the HP Envy 14-j 104TX with Ubuntu 14, and determined the RS code parameters as $[255, 60]$. With the assigned PWAM parameters of $[0.125, 2, 1]$, we finally achieved a maximum throughput of $17.28kbps$ in the high-speed communication mode.

6.2 One-Way Communication Performance

We evaluate the one-way communication performance using two prototypes described in Sec. 5. In Prototype I where a laptop is the transmitter and the mobile phone (Huawei P20Pro) is the receiver, PWAM symbol length is set to $100ms$ and each symbol contains 4 bits ($M = N = 2$). In Prototype II where one laptop acts as the transmitter and the other laptop acts as the receiver, the symbol length is set to 30 ms and each symbol contains 4 bits ($M = N = 2$). Each experiment lasts for 10 minutes and is repeated for 10 times. The average throughput is reported.

Fig. 20(b) and Fig. 20(a) first show how MagneComm+ perform while using different laptops as transmitters. The 4 laptops have various OSes (Win8, Win10, Ubuntu14 and MacOSX10) and CPU models (Intel i5 with 4 cores and i7 with 8 cores). When there is no other program running, MagneComm+ achieves 33 bps in Prototype I and 110 bps in Prototype II. While the noise is present, the throughput decreases. For example, while the user is surfing websites, the throughputs drop to 19 bps and 60 bps, respectively.

Fig. 21(b) and Fig. 21(a) show the throughput under various transmission ranges when there is no background noise. We make several observations. First, the throughput remains high for all types of transmitters when the range is 3 cm. Second, due to the SNR drops over distance, the valid communication for all transmitters is 5 cm and 8 cm for Prototype I and II, respectively. Finally, after 3 cm, the throughput of different transmitters varies significantly. More specifically, the throughput of ThinkPad T440 and DELL Inspiron 15 only drops by 26% at the distance of 5 cm while that of other laptops drops by more than 60% at the same distance.

In additional to Prototype I and II, we also implement another prototype where an Android phone acts as the transmitter and a laptop with an external magnetometer acts as the receiver. The Android phone is a Huawei P20Pro equipped with a Octa-core CPU, and the laptop is equipped with an external magnetometer. One-way communication is installed in this prototype with the throughput of 95 bps

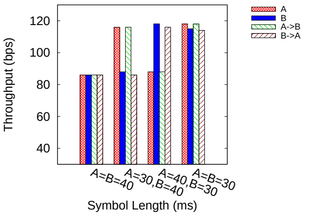


Fig. 22. Full-duplex Performance Analysis. A→B means laptop A transmit data to B. Similar definition is also applied to B→A.

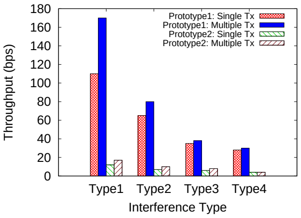
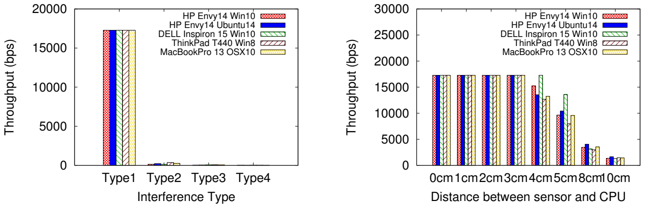


Fig. 23. Multiple Transmitters performance analysis. Scenario type 1 to 4 are the same with Fig. 20.



(a) Under various scenarios.

(b) Under various distances.

Fig. 24. Throughput under different environment settings of full speed communication. Scenario type 1 to 4 are the same with Fig. 20.

and maximal communication range of $4\ cm$. We exclude the details of evaluation due to the page limit.

6.3 Full-Duplex Communication Performance

We evaluate the full-duplex communication performance using Prototype II where two laptops are equipped with magnetometers and transmit data at the same time. Two laptops' CPUs are $3\ cm$ away from each other. In Fig. 22, we show the throughput of two transmission directions in full-duplex communication and the throughput in one-way communication. Note that A and B represents the throughput in one-way communication where A and B acts as the receiver, respectively. $A \rightarrow B$ represents the throughput in full-duplex communication where A transmits data to B. We can see that when the total throughput in full-duplex communication (*i.e.*, $A \rightarrow B + A \rightarrow B$) is 195.5% of that in one-way communication. It implies that our cancellation mechanism works effectively so the total throughput is almost doubled.

6.4 Multiple Transmitters Performance

We evaluate the performance of Multiple Transmitters protocol using both prototypes under one-way communication. Fig. 23 shows the results. By increasing the number of tx antennas (*i.e.*, the number of cores) from 1 to 2, the throughput is increased by 141.6% and 154.5% in Prototype I and II under the clean background, respectively. In noisy scenarios, the performance of the Multiple Transmitters protocol decreases because using multiple cores increases the risk of being interfered by the noise.

6.5 Full Speed Communication Performance

We evaluate the performance of full-speed communication mode using Prototype II. In Fig. 24(a), we show the throughput of under the different interference scenarios. When there is no other programs running (except the OS's tasks), the full-speed communication achieves around the effective throughput of 17280 *bps* in Prototype II. While the noise is present, the throughput decoded by the receiver drops sharply to approximately 1100 *bps*, note that these decoded data has no meaning because it cannot recover the original transmission data. Therefore, we suggest that when users want to use MagneComm+ without affecting their daily use of devices, they can choose the normal-speed communication mode. Fig. 24(b) show the throughput under various transmission ranges when there are no other programs running. We obverse that the full-speed communication mode has a similar communication range to that of the normal-speed mode.

6.6 Energy Consumption

We measure the power consumption of MagneComm+ when HP Envy 14-j 104TX with Intel Core i7 6700HQ CPU acts as the transmitter and DRV425 magnetometer acts as the receiver. The transmitter uses one core to generate one-way traffic for 1.5 hours. The average power consumption is reported in the Table. 2. Although the energy consumption of MagneComm+ is higher than other protocols, its capability of providing additional bandwidth using existing hardware is still valuable.

TABLE 2
Energy consumption of various communication protocols.

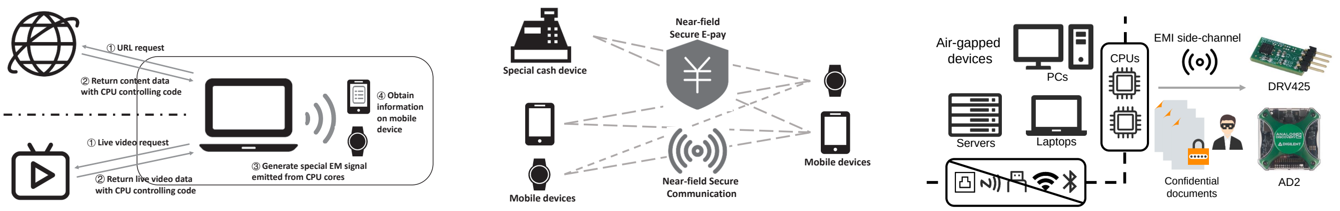
Protocols	Throughput	Watt	J/bit
Bluetooth 4.0 (BLE)	$0.27\ Mbps$	$0.5W$	1.9×10^{-6}
Wi-Fi 802.11ac	$350\ Mbps$	$2.4W$	6.8×10^{-9}
NFC ISO-13157	$424\ kbps$	$0.6W$	1.4×10^{-6}
Pulse [15]	$44\ bps$	$8.3 \times 10^{-4}W$	1.9×10^{-5}
MagneComm (Normal-speed)	$110\ bps$	$5.2W$	4.8×10^{-2}
MagneComm (Full-speed)	$17.28\ kbps$	$5.7W$	3.9×10^{-4}

7 Applications of MagneComm+

7.1 MagneCode & Expanded Screen

When using a laptop or mobile device, it is common nowadays to obtain information using a QR code; however, this is not always possible when using a smaller mobile device, such as a smart watch, because they may not have a camera. Nonetheless, even the smallest devises are equipped with a magnetometer for e-compass functionality. Thus, *MagneCode* can serve as an alternative to QR code when camera is not available. As shown in Fig. 25(a), when a user wants to use a mobile device to obtain additional information, they first request the URL, which returns the controlling code that allows the laptop to generate *MagneCode*. Moreover, MagneComm+ can be used to add background information, such as player statistics or advertisements, to the video content being viewed by the user and shown on an expanded screen without interfering with the original content.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3133481, IEEE Transactions on Mobile Computing


① URL request
② Return content data with CPU controlling code
① Live video request
② Return live video data with CPU controlling code
③ Generate special EM signal emitted from CPU cores
④ Obtain information on mobile device
Special cash device
Near-field Secure E-pay
Near-field Secure Communication
Mobile devices
Mobile devices
Air-gapped devices
PCs
Servers
Laptops
CPUs
EMI side-channel
DRV425
Confidential documents
AD2

(a) *MagneCode* and Expanded Screen. (b) *MagnePay*. (c) *MagneThieve*.

Fig. 25. Potential applications for MAGNECOMM+. (a) *MagneCode* is similar with QR codes that enables communication between two devices, and MAGNECOMM+ can also enable expanded screen; (b) *MagnePay* is a secure and convenient mobile payment method based on the MAGNECOMM+ protocol; (c) *MagneThieve* can steal the confidential documents on the air-gapped devices with the convert EMI communication channel.

7.2 MagnePay

NFC and RFID are mainstays of the near-field wireless communication systems used for e-commerce. However, the need for NFC hardware modules greatly limits the application and popularity of convenient e-payment. MAGNECOMM+ technology could be used to design an unified e-payment system without the requirement of NFC chips, based on the fact that magnetometers are used in nearly all mobile devices. As shown in Fig. 25(b), we develop a device capable of generating EMI signals specific for businesses. The device can be used to embed business-related information within EMI signals, which can be accessed by customers on a mobile device (smartphone or smartwatch). This could also be used to make payments for friends, wherein one smartphone is used to generate the special EMI signals and acts as a transmitter, while the other smartphone acts as a receiver, without the need for any special devices.

7.3 MagneThieve

Financial organizations, critical infrastructure, and commercial industries always use the air-gapped devices [16] in order to reduce the leakage risk of sensitive or confidential data. And air-gapped devices are kept isolated from the Internet or other less secure networks, and their physical communication peripherals (*e.g.*, Wi-Fi, Bluetooth, USB and etc.) are also unusable. Therefore, we take the advantage of the naturally convert characteristic of EMI communication channel and design *MagneThieve* to break the communication limitation on the air-gapped devices. As shown in Fig. 25(c), an attacker can easily transmit data via controlling the CPUs on the target devices with no permission required and utilize a magnetometer to decode the transmission data. The full-speed communication mode can be applied when stealing large-size confidential documents.

8 RELATED WORK

Near-field wireless communication has attracted considerable attention due to the richness of the potential applications. Numerous techniques have been developed using a variety of media to enable the communications.

Visual: Visible light communication (VLC) provides over bandwidth $10K\times$ greater than that of the common radio spectra. Due to its directionality and containment, VLC is also a good candidate for near-field communication [3]. In [17], the researchers proposed a novel near-field communication system for smartphones based on visible light. In [18], a secure vlc system is proposed with the two-dimensional spatially aliased patterns. Unfortunately, VLC generates redundant light pollution and its line of sight characteristics means that it cannot pass through obstacles. [4] presented a method by which to use VLC in dark environments; however, the LED drivers currently available are unable to modulate data bits into light pulse, and replacing existing LEDs would be very expensive.

Quick Response Codes (QR codes) and other 2D barcodes, also belong to VLC. QR codes are widely used in advertising and payments. Strata [19] proposed a layered coding scheme to enable visual communications and improve the scalability of QR codes. mQRCode [20], [21] proposed a secure QR code communication using nonlinearity of spatial frequency in light. Unfortunately, camera sensors are still not available on most wearable mobile devices.

Vibration: Ripple [22] explored the possibility of using physical vibrations as a mode of wireless communication. Ripple achieved data transmission rates of 80 bps on Android smartphones. Ripple II [23] achieved a transmission rate of 30 $kbps$ with the application of OFDM. An ideal application of vibration communication would be in the development of body networks, where human bone could be used to transfer vibration signals. Unfortunately, in other scenarios, noise caused by the vibration limits applicability.

Audio: Acoustic communication over speaker-microphone links has also been explored for near-field communication. Dolphin [1] is a real-time acoustic-based dual-channel communication system in which a speaker is used as a transmitter and microphones are used as receivers. Data signals can be directly embedded within the original audio signal without any perceived difference, and Dolphin has achieved data transmission rates of 500 bps on smartphones. In [2], a near-ultrasound communication system was developed using the speakers of TVs and the microphone on a smartphone. This system is able to transmit data at 15 bps in a typical TV-watching environment.

Magnet: NFC Standards are mature short-range communication technologies based on magnetic induction. These systems rely on near-field coupling (approximately $5cm$) without the need for a discovery mechanism. NFC can achieve data rates of up to 424 $kbps$. Pulse [15] is a system that avoids the need for specialized hardware on the receiver. Rather, it uses two solenoids as transmitters (generating a modulated magnetic signal), and the magnetometer on mobile phones are used to decode the magnetic signals. Nonetheless, Pulse still requires a purpose-built solenoid for use as a transmitter. A number of commercial products based on magnetic communication, such as LibertyLink docker [24] and FreeLinc's Near Field Magnetic Induction [25], have been implemented in situations requiring a secure and reliable communications channel. However, the need for antennas limits their popularity.

MagneComm+ enables near-filed EMI communication using components that are already available in almost all mobile devices. Thus, implementation requires only software for the transmitter and receiver without any additional dedicated antennas and codec chips. We have also developed a one-way communication protocol for scenarios in which transmitters are not equipped with a magnetometer. MagneComm+ also works in full-duplex mode while making full use of the transmitter hardware.

Other Magnet Application Works: One of the most common applications for magnet sensing is localization. In [26], [27], features of the Earth's magnetic field were used to implement indoor navigation systems. [26] proposed an unique cloud platform that runs disruptive geomagnetic positioning in its core to accurately pinpoint locations within a building using the magnetometer in smartphones. [27] demonstrated the feasibility of using an array of e-compasses to measure disturbances in the geomagnetic field caused by structural steel elements for use in indoor localization applications.

Devices with currents inside their electronic units produce EMI signals is also explored for many applications. [28], [29], [30], [31], [32] utilize magnetometers to track electromagnetic changes emitted from different CPU operations and infer private mobile app usage behaviors. Finexus [33] and uTrack [34] proposed to use the change in magnetic field to track motions of multiple fingertips. DOSE [35] used time-varying electromagnetic interference to monitor the operating states of appliance and infer human activities. MagPrint [36] proposed a deep learning based classification model to implement the continuous user authentication on mobile devices. These applications are different from our work and focus on utilizing magnetic field in security domain or explore the usage for human-computer interaction.

9 Conclusion

We present a novel secure near-field communication protocol based on the electromagnetic induction (EMI) side-channel signals for mobile and IoT devices that are not equipped with NFC chips. The proposed system, called MagneComm+, is using EMI signals emitted from working CPU cores as the communication medium and the magnetometers for receiving transmission data. Normal-speed mode of the MagneComm+ can be used for communication scenarios where users can normally use the transmitter devices, and full-speed mode is designed for high-throughput communication scenarios. The efficacy of our MagneComm+ is demonstrated in experiments where it achieves throughput of 33 *bps* over a distance of up to 10 *cm* between a laptop and a smartphone. When equipped with an external high-performance magnetometer, our proposed MagneComm+ can achieve throughput of **110 *bps*** on the normal-speed communication mode without affecting uses' usage on the transmission devices. When a user switches the MagneComm+ on the high-speed communication mode and disables all other running user's tasks on the transmission device, we achieve a maximum throughput of **17.28 *bps*** that can support efficient near-field data transmission.

Acknowledgements

This work was supported by NSFC grant 61572324, Shanghai Talent Development Fund, and Fundamental Research Funds for the Central Universities 2017QNA4017.

References

[1] Q. Wang, K. Ren, M. Zhou, T. Lei, D. Koutsonikolas, and L. Su, "Messages behind the sound: real-time hidden acoustic signal capture with smartphones," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, 2016, pp. 29–41.

[2] S. Ka, T. H. Kim, J. Y. Ha, S. H. Lim, S. C. Shin, J. W. Choi, C. Kwak, and S. Choi, "Near-ultrasound communication for tv's 2nd screen services," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, 2016, pp. 42–54.

[3] N. Chi, H. Haas, M. Kavehrad, T. D. Little, and X.-L. Huang, "Visible light communications: demand factors, benefits and opportunities [guest editorial]," *IEEE Wireless Communications*, vol. 22, no. 2, pp. 5–7, 2015.

[4] Z. Tian, K. Wright, and X. Zhou, "The darklight rises: Visible light communication in the dark," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, 2016, pp. 2–15.

[5] Wikipedia, "Lorentz force," https://en.wikipedia.org/wiki/Lorentz force, 2021.

[6] T. Instruments, "Drv425: Fully-integrated fluxgate magnetic sensor," https://www.ti.com/product/DRV425, 2021.

[7] Digilent, "Analog discovery 2: 100ms/s usb oscilloscope, logic analyzer," https://reference.digilentinc.com/test-and-measurement/analog-discovery-2/, 2021.

[8] D. Brodić, "Analysis of the extremely low frequency magnetic field emission from laptop computers," *Metrology and Measurement Systems*, vol. 23, no. 1, pp. 143–154, 2016.

[9] D. Genkin, L. Pachmanov, I. Pipman, and E. Tromer, "Stealing keys from pcs using a radio: Cheap electromagnetic attacks on windowed exponentiation," in *International workshop on cryptographic hardware and embedded systems*. Springer, 2015, pp. 207–228.

[10] THOMAS, "Measuring emi shielding effectiveness," https://www.thomasnet.com/articles/automation-electronics/effective-emi-shielding/, 2021.

[11] T. Technologies, "White paper processor affinity multiple cpu scheduling," http://www.tmurgent.com/WhitePapers/ProcessorAffinity.pdf, 2003.

[12] H. Pan, Y.-C. Chen, G. Xue, and X. Ji, "Magnecomm: Magnetometer-based near-field communication," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, 2017, pp. 167–179.

[13] S. B. Wicker and V. K. Bhargava, *Reed-Solomon codes and their applications*. John Wiley & Sons, 1999.

[14] "reedsolo 1.5.4," 2021, https://pypi.org/project/reedsolo/.

[15] W. Jiang, D. Ferreira, J. Ylioja, J. Goncalves, and V. Kostakos, "Pulse: low bitrate wireless magnetic communication for smartphones," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2014, pp. 261–265.

[16] M. Guri, M. Monitz, and Y. Elovici, "Bridging the air gap between isolated networks and mobile phones in a practical cyber-attack," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 4, 2017.

[17] J. Niu, W. Song, C. Liu, L. Shu, and C. Chen, "Necas: Near field communication system for smartphones based on visible light," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2014, pp. 2426–2431.

[18] H. Pan, Y.-C. Chen, and G. Xue, "Poster: Secure visible light communication via two-dimensional spatially aliased patterns," in *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, 2019, pp. 51–52.

[19] W. Hu, J. Mao, Z. Huang, Y. Xue, J. She, K. Bian, and G. Shen, "Strata: Layered coding for scalable visual communication," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, 2014, pp. 79–90.

[20] H. Pan, Y.-C. Chen, G. Xue, C.-W. B. You, and X. Ji, "Secure qr code scheme using nonlinearity of spatial frequency," in *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, 2018, pp. 207–210.

[21] H. Pan, Y.-C. Chen, L. Yang, G. Xue, C.-W. You, and X. Ji, "mqrcode: Secure qr code using nonlinearity of spatial frequency in light," in *The 25th Annual International Conference on Mobile Computing and Networking*, 2019, pp. 1–18.

[22] N. Roy, M. Gowda, and R. R. Choudhury, "Ripple: Communicating through physical vibration," in *12th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 15)*, 2015, pp. 265–278.

[23] N. Roy and R. R. Choudhury, "Ripple {II}: Faster communication through physical vibration," in *13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16)*, 2016, pp. 671–684.

[24] D. Wolfson, "The libertylink docker wireless headset," [*product review*], *Computing Unplugged*, vol. 1, 2004.

[25] FREELINC, "Near field magnetic induction communication (nfmi) by freelinc," http://www.freelinc.com/technology, 2019.

[26] IndoorAtlas, "Last meter accuracy through technology fusion," https://www.indooratlas.com/positioning-technology/, 2021.

[27] J. Chung, M. Donahoe, C. Schmandt, I.-J. Kim, P. Razavai, and M. Wiseman, "Indoor location sensing using geo-magnetism," in *Proceedings of the 9th international conference on Mobile systems, applications, and services*, 2011, pp. 141–154.

[28] A. Zajić and M. Prvulovic, "Experimental demonstration of electromagnetic information leakage from modern processor-memory systems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 56, no. 4, pp. 885–893, 2014.

[29] D. Genkin, I. Pipman, and E. Tromer, "Get your hands off my laptop: Physical side-channel key-extraction attacks on pcs," *Journal of Cryptographic Engineering*, vol. 5, no. 2, pp. 95–112, 2015.

[30] Z. Zhu, H. Pan, Y.-C. Chen, X. Ji, F. Zhang, and C.-W. You, "Magattack: remote app sensing with your phone," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 241–244.

[31] Y. Cheng, X. Ji, W. Xu, H. Pan, Z. Zhu, C.-W. You, Y.-C. Chen, and L. Qiu, "Magattack: Guessing application launching and operation via smartphone," in *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, 2019, pp. 283–294.

[32] H. Pan, L. Yang, H. Li, C.-W. You, X. Ji, Y.-C. Chen, Z. Hu, and G. Xue, "Magthief: Stealing private app usage data on mobile devices via built-in magnetometer," in *2021 18th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2021.

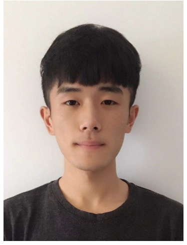
[33] K.-Y. Chen, S. N. Patel, and S. Keller, "Finexus: Tracking precise motions of multiple fingertips using magnetic sensing," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016, pp. 1504–1514.

[34] K.-Y. Chen, K. Lyons, S. White, and S. Patel, "utrack: 3d input using two magnetic sensors," in *Proceedings of the 26th annual ACM symposium on User interface software and technology*, 2013, pp. 237–244.

[35] K.-Y. Chen, S. Gupta, E. C. Larson, and S. Patel, "Dose: Detecting user-driven operating states of electronic devices from a single sensing point," in *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2015, pp. 46–54.

[36] L. Yang, Y.-C. Chen, H. Pan, D. Ding, G. Xue, L. Kong, J. Yu, and M. Li, "Magprint: Deep learning based user fingerprinting using electromagnetic signals," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 696–705.

Guangtao Xue received the PhD degree from the Department of Computer Science and Engineering at the Shanghai Jiao Tong University in 2004. He is currently a professor in the Department of Computer Science and Engineering at Shanghai Jiao Tong University, China. His research interests include vehicular ad hoc networks, wireless networks, mobile computing, and distributed computing. He is a member of the IEEE and the IEEE Communication Society.



Hao Pan is a PhD student from the Department of Computer Science and Engineering at Shanghai Jiao Tong University. He has received his bachelor's degree from the Yingcai Honors College at University of Electronic Science and Technology of China in 2016. His current research interests reside in mobile computing, with special focuses on wireless communication and sensing, security for mobile systems and mobile human-computer interaction.

Yi-Chao Chen joined Shanghai Jiao Tong University as a tenure-track Assistant Professor in the Department of Computer Science and Engineering in 2018. He received the B.S. and M.S. in the Department of Computer Science and Information Engineering at National Taiwan University in 2004 and 2006, respectively. He got his Ph.D. in Computer Science at the University of Texas at Austin in 2015. Prior to joining SJTU, he spent a year as a Researcher in Huawei Future Network Theory Lab in Hong Kong and then worked as a Co-founder in Hauoli LLC. His research interests focus on networked systems and span the areas of wireless networking, network measurement and analytics, and mobile computing.

Xiaoyu Ji received his BS degree in Electronic Information & Technology and Instrumentation Science from Zhejiang University, Hangzhou, China, in 2010. He received his Ph.D. degree in Department of Computer Science from Hong Kong University of Science and Technology in 2015. From 2015 to 2016, he was a researcher at Huawei Future Networking Theory Lab in Hong Kong. He is now an associate professor with the Department of Electrical Engineering of Zhejiang University. His research interests include IoT security, including sensor, network, and AI security. He won the best paper award of ACM CCS 2017, ACM AsiaCCS 2018. He is a member of IEEE.

Jiadi Yu received the PhD degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2007. He is currently a professor in the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. Prior to joining Shanghai Jiao Tong University, he was a postdoctoral fellow in the Data Analysis and Information Security (DAISY) Laboratory at Stevens Institute of Technology from 2009 to 2011. His research interests include cyber security and privacy, mobile and pervasive computing, cloud computing, and wireless sensor networks. He is a member of the IEEE and the IEEE Communication Society.