MotorBeat: Listen to the Hearts of Your Home Appliances

ABSTRACT

More and more home appliances are now connected to the Internet, thus enabling various smart home applications. However, a critical problem that may impede the further development of smart home is overlooked: Small appliances account for the majority of home appliances, but they receive little attention and most of them are cut off from the Internet. To fill this gap, we propose MotorBeat, a communication approach that connects small appliances to a smart speaker, which acts as a gateway, without additional hardware cost and without RF spectrum occupation. Our key idea is to exploit direct current (DC) motors, which are common components of small appliances, to transmit acoustic messages. By introducing Random Pulse Width Modulation (R-PWM) to drive DC motors, MotorBeat achieves the following 3C goals: (1) *Comfortable* to hear, (2) *Compatible* with multiple motor modes, and (3) *Concurrent* transmission. We implement MotorBeat with commercial devices and evaluate its performance on three small appliances and ten DC motors. The results show that the communication range can be up to 10m.

1 INTRODUCTION

The recent years have witnessed rapid advances in smart home [13, 42]. Many home appliances are now connected to the Internet and thus are endowed with interesting capabilities to improve the user experience. The users can interact with them remotely, and these appliances themselves can also cooperate in providing a seamless service [25].

Despite the trend of smart home, there are still a considerable number of home appliances cut off from the Internet, especially small appliances. Home appliances can be mainly classified into three categories [54]: small appliances, major appliances (a.k.a. white goods), and consumer electronics (a.k.a. brown goods). In contrast to major appliances, like refrigerators and washing machines, small appliances mainly refer to portable or semi-portable household devices. Examples include electric toothbrushes, humidifiers, and fans.

One basic fact is that small appliances now account for a major proportion of home appliances, and this proportion is expected to continue to grow. According to statistics in 2020 [44], the sales volume of small appliances has already been 8.3x larger than that of major appliances in the USA. Besides, small appliances are dedicated to meeting consumers' various and fine-grained demands, while major appliances mainly focus on accomplishing necessary housework tasks. This means that the demand for small appliances will increase with the quality of life.

Group	Keyword	Sales (k)	Price ($)	Ratio
Bathroom	Electric Toothbrush	384.4	66.9	3%
	Hair Trimmer	280.4	16.4	0%
	Electric Razor	194.4	60.9	0%
	Hair Dryer	182.6	46.5	0%
Kitchen	Rice Cooker	632.0	120.9	0%
	Electric Mixer	151.5	36.1	0%
	Coffee Maker	135.6	94.4	0%
	Blender	116.5	210.6	0%
Living Room or Bedroom	Humidifiers	366.9	45.2	3%
	Massager	340.2	90.8	1%
	Vacuum Cleaner	278.1	196.5	5%
	Fan	198.0	68.9	35%
	Average	271.8	87.8	3.9%

Table 1: The sales, prices and connection ratios of small appliances on Amazon.

However, the vast majority of small appliances are not connected to the Internet. We investigate three groups of small appliances, shown in Table 1. We separately select the four most popular appliances in each group based on their sales. The names of these appliances are used as keywords to search for the top 100 best-selling items on Amazon [2]. Table 1 shows the ratio of small appliances that are connected to the Internet, as well as their total sales[1] and average prices. As shown, a few fans have the communication ability (i.e., remote control), but most small appliances are isolated from the Internet. The average connection ratio is only 3.9%.

In a nutshell, the elephant in the room is the problem that small appliances account for the majority of home appliances, but they receive little attention, and few of them are connected to the Internet.

One straightforward solution is to add RF modules to small appliances directly. However, this solution is infeasible in practice due to the following two reasons: (1) Limited hardware budget. In contrast to major appliances which are the necessities of life, consumers are more sensitive to the prices of small appliances. This generally requires manufacturers to control the hardware budget of small appliances as low

[1]The sales are calculated as the total number of customer reviews

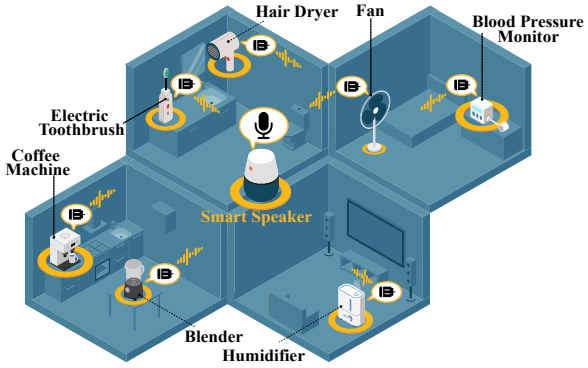


Figure 1: An illustration of MotorBeat.

as possible. (2) Limited RF spectrum resource. In a modern family, small appliances are an order of magnitude more than major appliances. Given the fact that the ISM band has already been quite crowded, if so many small appliances are all networked via RF modules, they would seriously interfere with each other, as well as the ongoing RF channels.

In this paper, we ask *can we connect small appliances to the Internet (1) without additional hardware cost and (2) without additional RF spectrum occupation?* We present MotorBeat to this end. Figure 1 demonstrates our key idea. The idea is to exploit the existing electric motors of small appliances to talk to the smart speaker in the acoustic channel. Our idea is supported by two key observations:

The first observation is that nearly 80% of small appliances contain an electric motor[2]. This observation ensures that MotorBeat can be widely applied to most small appliances. The second observation is that smart speakers are now very popular, which have access to the Internet and contain always-on microphones. Smart speakers can then behave as free gateway for small appliances by receiving and relaying their acoustic messages to the Internet.

We notice that a recent pioneering work, Bleep [6], also exploits motors to enable UAVs to communicate with each other. The solution of Bleep is acceptable in industrial environments, but is unreasonable in home environments: First, Bleep encodes information in linear chirps and generates sounds that are unfriendly to human ears. Second, Bleep equips each UAV with a microphone to sense channels and choose an idle channel. In our case, small appliances hardly contain a microphone to avoid collisions, and it's unrealistic to equip each appliance with a microphone additionally.

Therefore, MotorBeat should also achieve the following **3C** (Comfortable, Compatible, and Concurrent) goals:

- **Comfortable** to hear. Small appliances are designed to serve customers. The quality of experience is critical for small appliances. Therefore, MotorBeat should not disturb the users while transmitting acoustic messages.
- **Compatible** with multiple motor modes. The motors of small appliances typically have multiple working modes, and these modes could be changed unpredictably by the users at any time. This requires that MotorBeat should not affect the function of these motors, and the transmission should not be disturbed by users.
- **Concurrent** transmission. Small appliances typically have no microphone to monitor the transmission of other appliances. This means that collision detection or collision avoidance is infeasible for small appliances, and collisions are inevitable in our scenario. To tolerate the collisions, MotorBeat should allow concurrent transmission and support decoding from collided signals.

To achieve the above **3C** goals, MotorBeat introduces a novel modulation technique, Random Pulse Width Modulation (R-PWM). At a high level, small appliances drive the voltage of their motors with a pseudo-random switching frequency. By doing so, the harmonics of the acoustic signals, which sound uncomfortable, are significantly reduced (**Comfortable**). Furthermore, even though the switching frequency is random, the duty cycle of the voltage can be configured online without impeding the transmission, thus supporting multiple working modes of motors (**Compatible**). Last, we assign each appliance with a unique R-PWM symbol. These symbols are orthogonal to each other. Similar to Code-Division Multiple Access (CDMA), multiple appliances are allowed to transmit concurrently. The MotorBeat receiver is able to separately detect and decode messages of these appliances (**Concurrent**). Our contributions are as follows:

- We propose MotorBeat, a novel motor-based communication paradigm that enables small appliances to talk in an acoustic channel. Our solution requires no extra hardware budget and does not introduce additional RF traffic.
- We disclose the acoustic characteristics of DC motors, and show the opportunity to communicate based on them. We introduce a novel modulation technique, R-PWM, to drive the motors and achieve **3C** goals required in the real world.
- We implement MotorBeat and evaluate its performance on three small appliances (electric toothbrush, blood pressure monitor, and electric razor) and ten different DC motors. The results show that MotorBeat can be widely applied to small appliances. The communication range can be up to 10 m, ensuring that MotorBeat can cover a standard house.

Roadmap. Sections 2 and 3 introduce how a DC motor works and its acoustic characteristics. Section 4 gives MotorBeat's overview. In Sections 5, 6, 7, and 8, we elaborate on the design. Section 9 presents the implementation and evaluation results. Section 10 discusses the related work. Sections 11 and 12 respectively discuss and conclude this work.

[2]We check all 56 default small appliances presented on Amazon [3], and 44 of them contain an electric motor, which accounts for 78.6% (44/56).

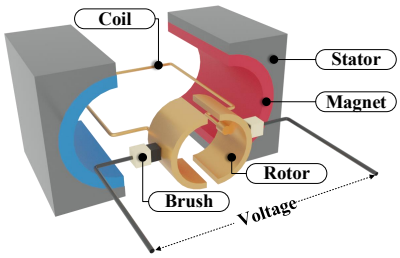


Figure 2: DC motor structure.

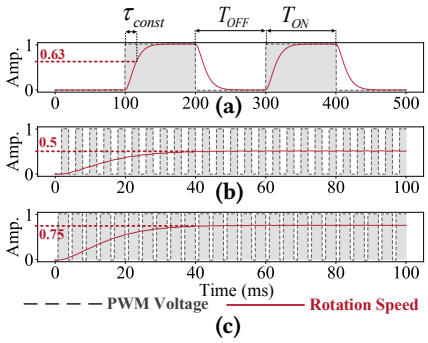


Figure 3: PWM voltage vs. speed.

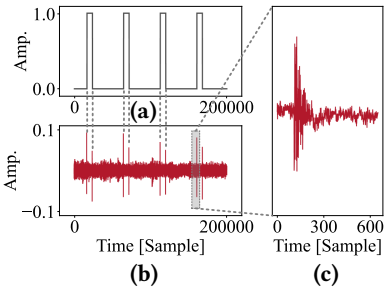


Figure 4: (a) The voltage. (b) The audio. (c) The spike (EM signal).

2 PRIMER

2.1 DC Motor

A direct current (DC) motor[3] shown in Figure 2 consists of two main components: stationary part (stator) and rotating part (rotor). The stator contains permanent magnets on the side, and the rotor comprises coils. The working process of a DC motor is as follows. According to the Lorentz law, when the DC motor is powered, an additional magnetic field will be created inside the rotor. The rotor is then attracted or repelled by the magnets on the stator. Two electronic brushes connect the rotor to the voltage. These brushes can switch the direction of the coil current during rotation and thus switch the polarity of the electromagnet. By doing so, the rotor can keep rotating in the same direction.

2.2 PWM Voltage

Pulse Width Modulation (PWM) voltage is widely adopted to drive small DC motors for its simplicity. PWM voltage is a periodic signal with two states, ON and OFF, and is described by two parameters: (1) *duty cycle* α and (2) *switching period* T_{sw}. Here, duty cycle $\alpha = \frac{T_{ON}}{T_{ON}+T_{OFF}}\%$, where T_{ON} and T_{OFF} denote the durations of states ON and OFF within a period, respectively (see Figure 3(a)). Another parameter switching period T_{sw} is defined as $T_{ON} + T_{OFF}$.

The DC motor can work steadily when the switching frequency f_{sw} satisfies the following condition:

$$f_{sw} = \frac{1}{T_{sw}} \gg \frac{1}{\tau_{const}} \tag{1}$$

where τ_{const} denotes the **time constant**[4] of a DC motor and it can be used to quantify how fast a DC motor responds to the input voltage (see Figure 3(a)).

One way to understand Equation 1 is to model the DC motor as a circuit with a resistor and an inductor in series, which is equivalent to a first-order low pass filter with cut-off frequency $1/\tau_{const}$. Therefore, as long as the input voltage frequency f_{sw} is larger than the cut-off frequency $1/\tau_{const}$, the DC motor will exhibit a steady rotation speed.

The advantage of PWM is that by merely changing the duty cycle α, the appliance can easily control its motor's rotation speed. In Figure 3(b), the duty cycle is 50%, and the switching frequency is 50x larger than that in Figure 3(a). After a short transient period (0-40ms), the motor exhibits a steady speed, 50% of its maximum speed. In Figure 3(c), the duty cycle increases to 75%, and thus the steady speed correspondingly increases to 75% of its maximum speed.[5]

3 SIGNAL MODEL

3.1 Acoustic Characteristics of DC Motors

During rotation, DC motors generate two acoustic signals: mechanical signal and electromagnetic signal [18, 23].

- **Mechanical Signal** is mainly caused by the rotor unbalance, and the brush friction. On the one hand, the misalignment between the rotor's inertia axis and its geometric axis will make the supporting structure vibrate and generate sound. On the other hand, the friction between the brushes and the rotor also makes a sound.
- **Electromagnetic Signal** is mainly caused by the change of magnetic fields. The magnets on the stator generate a static magnetic field, and the coils on the rotor generate a dynamic magnetic field, controlled by the voltage. These two magnetic fields will make the stator and the rotor attract or repel each other, further causing the slight deformation of the motor and the vibration.

3.2 Identification of Signal and Noise

We select the electromagnetic (EM) signal to transmit messages, while the mechanical signal is the noise to our system. The considerations of our selection are following:

- Given the constraint that the motor's rotation should not be affected (**Compatible**), we can only modulate the EM

[3]Here, we take brushed motors as an example to introduce DC motors.
[4]Time constant is formally defined as the time the motor takes to reach its 63% ($\approx 1 - 1/e$) maximum speed.
[5]In practice, the rotation speed is not a linear function of the duty cycle.

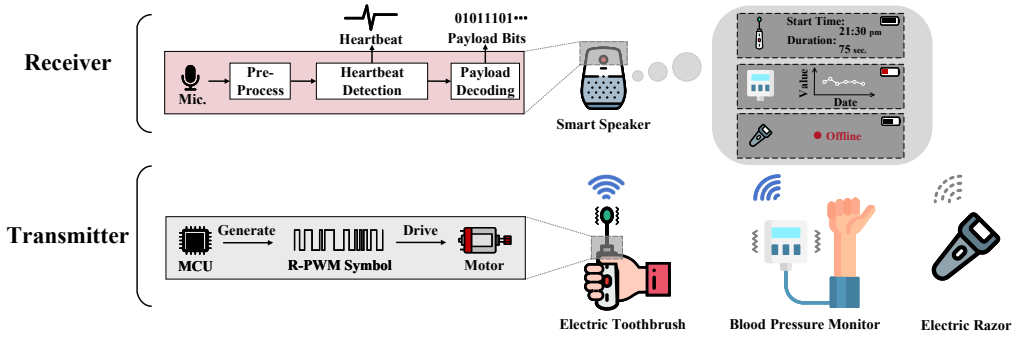


Figure 5: Overview of MotorBeat.

signal, rather than the mechanical signal. Compared with the mechanical signal, which is highly determined by the rotation, the EM signal is independent of the rotation. This property provides us an opportunity for modulation without disturbing the rotation.

- Even if we can modulate the mechanical signal, the mechanical signal is less predictable and controllable than the EM signal. This is because the rotation speed will change with the load (or force) onto the motor, and the load is unpredictable and dynamic (e.g., the force applied to the toothbrush). As a result, the rotation and the resulting mechanical signal are unpredictable. On the other hand, the EM signal is only determined by the voltage. As long as the voltage is deterministic, the EM signal is deterministic.

In MotorBeat, we modulate the voltage with an R-PWM symbol (introduced later in Section 5) to make the motor generate a specific sound. In this sound, only the EM signal S_{EM} can be modulated by the voltage and thus S_{EM} is our desired signal, while the mechanical signal S_{mech} will not be modulated and thus S_{mech} is a noise.

To further understand the relation between the EM signal and the voltage, we especially illustrate the voltage and the corresponding audio in Figure 4(a) and (b), respectively[6]. As we can see, the edges of the voltage will introduce spikes in the audio. Let us first take the positive edge (from OFF to ON) as an example to explain this: At the moment when the voltage is switched ON, the EM force will be generated and introduce a deformation of the motor. The sudden appearance of this deformation behaves like a physical knock onto the motor, thus making the sound (see Figure 4(c)). Similarly, when the voltage is switched OFF, the deformation will suddenly disappear and also behave like a physical knock.

On the other hand, the mechanical signal has no relation with the modulated voltage. The mechanical signal is mainly determined by the rotation. As introduced in Section 2.2, if $f_{sw} \gg 1/\tau_{const}$, the rotation speed is constant[7], which is uncorrelated with the modulated voltage.

Finally, we want to point out that the EM signal is buried by the noise in practice. Firstly, for many small appliances, their DC motors are low-power devices, and can only generate weak sounds. Secondly, the sound of a DC motor is dominated by the mechanical signal [11], which means the EM signal is typically overwhelmed by the mechanical signal. From our measurements, the signal-to-noise ratio can be lower than -15 dB when a microphone is placed only 3 cm away from a DC motor to receive the signal.

4 MOTORBEAT OVERVIEW

Figure 5 illustrates the overview of MotorBeat.

The MotorBeat transmitters are various small appliances that contain DC motors. Each appliance is assigned a predefined R-PWM symbol. According to its symbol, each appliance drives its motor to transmit acoustic messages.

The MotorBeat receiver is the smart speaker. The R-PWM symbols of small appliances are also known to the receiver (Section 8). The receiver then takes these symbols as templates to correlate with the audio samples to separately check whether these symbols exist in the audio (Section 6).

By doing so, the receiver then knows which appliances are operating (Section 7.1), and further obtains these appliances' heartbeats. Such ability can open up some interesting applications. For example, simply by knowing the heartbeats of your electric toothbrush, the smart speaker can know when and how long you brush your teeth each time. What's more, MotorBeat also exploits the time intervals between R-PWM symbols to encode data bits (Section 7.2), which allows small

[6]To clearly display the EM signal, we record the EM signal in very controlled conditions: (1) The rotor is deliberately fixed to avoid the mechanical signal introduced by the rotation. (2) The switching period of the voltage is enlarged to 1s to avoid the overlaps between the spikes. (3) To record this weak signal S_{EM}, the microphone is placed close to the motor.

[7]Even though the speed is constant, the mechanical sound is not a narrow-band signal, but a wide-band signal. This indicates that it is infeasible to separate the EM signal and the mechanical signal in frequency domain.

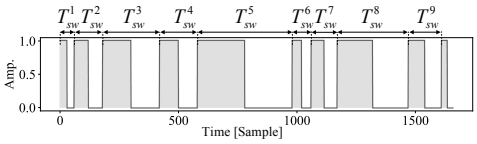


Figure 6: Illustration of R-PWM signal.

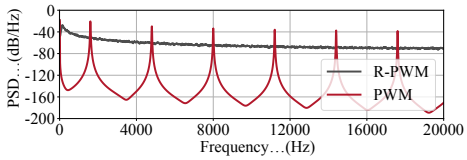


Figure 7: Power spectral density of R-PWM and PWM.

appliances to upload some information, such as remaining battery life or readings of blood pressure.

In the next (Section 5), we first introduce the R-PWM symbol, and explain how to achieve the **3C** goals by using R-PWM symbols.

5 TRANSMITTER DESIGN

5.1 Random PWM

Random PWM (R-PWM) is an extension of PWM, but randomizes the parameter(s) of PWM. Figure 6 shows an example of an R-PWM symbol. An R-PWM symbol is composed of a sequence of pulses. Different from PWM whose pulses are identical, the pulses of R-PWM are instead different from each other. In Figure 6, the switching period (T_{sw}^i) of each pulse is random, but the duty cycle is fixed to 50%.

Mathematically, an R-PWM symbol is defined by a tuple $< \mathcal{A}, \mathcal{T}_{sw} >$, where $\mathcal{A} = \{\alpha^1, \alpha^2, \alpha^3, \ldots\}$ denotes the duty cycles of each pulse, and $\mathcal{T}_{sw} = \{T_{sw}^1, T_{sw}^2, T_{sw}^3, \ldots\}$ denotes the switching periods of each pulse. In the rest of this paper, we use R-PWM to refer to PWM with a fixed duty cycle but with a random switching period. Namely, for each $\alpha^i \in \mathcal{A}$ and $T_{sw}^i \in \mathcal{T}_{sw}$, α^i is a constant value, while T_{sw}^i is random.

To generate a predefined R-PWM symbol for each appliance, we assign each appliance with a unique random seed s. The seed is used to initialize the random state to generate a sequence of pseudo-random switching periods $\{T_{sw}^1, T_{sw}^2, T_{sw}^3, \ldots\}$. The variable T_{sw}^i is uniformly distributed between T_{sw}^{min} and T_{sw}^{max}, i.e., $T_{sw}^i \sim U(T_{sw}^{min}, T_{sw}^{max})$.

Note that the R-PWM voltage used here will not damage DC motors. This is because the duty cycle in R-PWM is fixed, the resulting effective voltage (a.k.a. root-mean-square voltage) is thus statistically stable [21]. To make sure that motors can rotate steadily, T_{sw}^{max} should be less than motors' time constant τ_{const} (see Equation 1).

Algorithm 1: Online Transmit an R-PWM symbol

```
Initialize T_sw with random seed s, and initialize α.
Listen for button interrupt with a handler CallBack.
foreach T_sw^i in T_sw do
    Set voltage ON, and sleep α · T_sw^i seconds
    Set voltage OFF, and sleep (1 − α) · T_sw^i seconds
def CallBack(mode m)
    Update duty cycle α to a new value that corresponds to
      mode m
```

5.2 Comfortable to Hear

Small appliances are dedicated to providing good experiences for users, so Quality of Experience (QoE) is a critical factor of small appliances. This requires that MotorBeat should not disturb the users while driving the motor, and the modulated EM signal S_{EM} should be imperceptible to the users.

One solution to modulate S_{EM} is Frequency-Shift Keying (FSK). Specifically, by selecting different switching frequencies of PWM voltage, the motor can emit the EM signal embedded with specific information. For example, Bleep [6] is a recent work that enables the UAV motors to communicate. To convey a symbol, Bleep increases f_{sw} every 50 ms to transmit a chirp-like signal (a series of sounds with increasing but discrete frequencies). We observe that the modulated signal in Bleep will sound like "Do-Re-Mi-Fa" syllables, which is too conspicuous to neglect even though the EM signal is weak (we will evaluate this in Section 9.5).

This is because, from a physiological perspective, the human ear is especially sensitive to a sound whose energy is concentrated at discrete frequencies [33]. Figure 7 shows the Power Spectral Density (PSD) of a PWM signal. As we can see, the energy of this signal is mainly distributed in several discrete harmonics. Therefore, the sound in Bleep is annoying to the users. Of course, such sound is acceptable for UAVs designed to work in noisy industrial environments, but is unacceptable in quiet home environments.

We adopt the R-PWM method to modulate the EM signal, which is imperceptible to the users in MotorBeat. The key idea is that R-PWM randomly changes each switching frequency (i.e., $1/T_{sw}^i$), and distributes the energy over a wide frequency range (see Figure 7). As a result, the discrete harmonics of the corresponding EM sound are significantly reduced. The EM signal is then converted to a white-noise-like signal, which is friendly to the human ear.

5.3 Compatible with Multiple Modes

Many small appliances have different working modes. For example, the electric fan may have multiple operating speeds; The electric toothbrush usually supports multiple modes, such as `sensitive` and `clean`, to provide different intensities

[34]. Moreover, the users can change these modes at will by pressing the button, which means that the motor mode can change unpredictably. These facts require that MotorBeat should not only support multiple working modes, but also not be affected by users.

Our solution is simple. MotorBeat generates and transmits the R-PWM symbol on the air. Algorithm 1 demonstrates the simplified procedures. When a user pushes the button to select a new mode, the handler function `CallBack` will be triggered to update the duty cycle α. Therefore, the motor speed will change accordingly. Typically, a larger duty cycle corresponds to a higher rotation speed.

One may wonder, since the duty cycle has changed when an R-PWM symbol is being transmitted, will the actually transmitted symbol mismatch with the symbol previously known to the receiver, resulting in the failure of symbol detection?

The answer is no. The reason is the following. The receiver detects the transmitted R-PWM symbol of each appliance by correlating the audio with its previously-known symbol. This means that, provided that the transmitted symbol is highly correlated with this previously-known symbol, the receiver can detect the transmitted symbols. Besides, we observe that as long as two symbols have the same switching periods, they are highly correlated even if their duty cycles are different.

Figure 8 explains our observation. Two R-PWM symbols have different duty cycles, 75% and 50%. We highlight the matched and the mismatched parts in their pulses i with the green blocks and the red block, respectively. We can see that the total area of the green blocks is larger than that of the red block. This means that the correlation of the i-th pulse pair (denoted as Cor^i_{pulse}) is positive. In addition, we can check that the correlations of all other pulse pairs are also positive. The correlation of two symbols (denoted as Cor_{sym}) can be calculated as the sum of all pulse pairs' correlations:

$$Cor_{sym} = \sum_i Cor^i_{pulse} \tag{2}$$

Given that all the signs of Cor^i_{pulse} are the same, the absolute value of Cor_{sym} will not be canceled, but will accumulate with the increase of pulses.

In summary, our design ensures that even if the duty cycle of the symbol used by the receiver is different from that of the symbols actually transmitted, the receiver can still detect these transmitted symbols. In our implementation, the receiver only needs to know the switching periods of R-PWM symbols (i.e., the random seeds), and their duty cycles are fixed to 50% by default.

5.4 Concurrent Transmission

Although small appliances can now talk to the smart speaker by using their motors, these appliances cannot listen to each other. Therefore, it's infeasible for them to adopt a channel sensing based mechanism (e.g., CSMA) to avoid transmission collision.

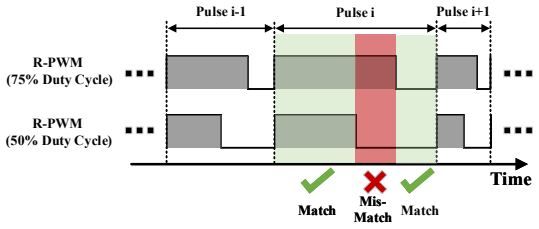


Figure 8: Two R-PWM symbols with the same switching periods but with different duty cycles.

MotorBeat supports concurrent transmission. This ability stems from the fact that the R-PWM symbols generated by different random seeds are statistically uncorrelated (or orthogonal) to each other. We can refer to Equation 2 again to understand this. If two R-PWM symbols' switching periods are randomly different, the correlations of their pulse pairs Cor^i_{pulse} will not have the same signs. Therefore, the correlation of these two symbols Cor_{sym} will be canceled with the increase of pulses. Such property allows the receiver to detect symbols from the collided signals separately.

In practice, a unique random seed is assigned to each appliance to generate the switching periods $\mathcal{T}_{sw}$ of its R-PWM symbol. This design not only ensures the orthogonality of different appliances' symbols, but also can be used to identify appliances because their R-PWM symbols are unique.

6 RECEIVER DESIGN

Here, we introduce the general approach to detecting R-PWM symbols. The specific methods may vary in specific applications, which are described in detail in Section 7.

The receiver, i.e., smart speaker, gets audio samples from its microphone, and then detects R-PWM symbols from the samples in the following ways: The receiver first normalizes previously-known R-PWM symbols to $[-1, 1]$. Next, it takes each normalized symbol as a template, and correlates it with the audio samples. The receiver then measures the mean (μ) and the standard deviation (σ) of the noise of the correlation, and sets a detection threshold as five standard deviations above the mean (i.e., $\mu + 5\sigma$). If the correlation is larger than the threshold, the receiver assumes that an R-PWM symbol is detected.

In practice, there are multiple small appliances' symbols that need to be detected. The receiver should compute correlations for each symbol. This means that the computational cost is directly proportional to the number of small appliances. Given the fact that there may be dozens of small appliances in a modern family, the smart speaker may not

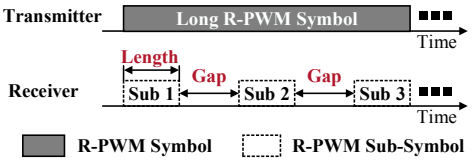


Figure 9: The detection of heartbeats

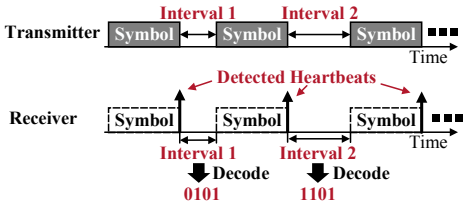


Figure 10: Data bits are embedded in symbol intervals.

afford such a computational cost. In the next section, we will discuss how to tackle this problem.

7 APPLICATIONS

7.1 Heartbeat Detection

Each appliance can repeatedly transmit its R-PWM symbol during operation. By doing so, the smart speaker can know the time when this appliance is ON or OFF (e.g., when you brush teeth), and each operation duration (e.g., how long you brush teeth each time). We call this 'heartbeat detection'.

In traditional RF communication, the transmitters intermittently broadcast their staying alive messages to reduce energy consumption and channel occupation. However, in MotorBeat, the transmitters are continuously transmitting long R-PWM symbols to broadcast their aliveness. This long R-PWM symbol is composed of multiple identical short R-PWM symbols. Before further introduction, we explain that the design of continuous transmission is reasonable:

- The continuous transmission will not annoy the users, because the transmission is friendly to the ears (**Comfortable**).
- The transmission will not disturb the motor's operation, or the change in its mode (**Compatible**).
- The transmission of one appliance hardly interferes with others' transmission (**Concurrent**).
- Additionally, the transmission is piggybacked on the ongoing operation of a motor. This means that the long transmission will not introduce additional power consumption.

To check each transmitter's aliveness, the receiver only needs to detect the parts of this symbol, instead of the whole symbol. Figure 9 further illustrates our idea. The receiver generates a series of fragments from the known long symbol at equal intervals. We define these fragments as **sub-symbols**. By doing so, the receiver can virtually view that the transmitter is broadcasting sub-symbols in series and is keeping notifying the receiver of its aliveness. The receiver first waits for the arrival of sub-symbol *Sub 1*. If the receiver asserts that sub-symbol *Sub 1* is detected, then it outputs a heartbeat. After that, the receiver begins waiting for the next sub-symbol, *Sub 2*. By repeating the above procedures, the receiver outputs the heartbeats sequentially.

Our design provides the receiver with much flexibility in detecting heartbeats, because the receiver can online tune both the length of sub-symbol L_{sub} and the gap length L_{gap}. (see illustration in Figure 9). Specifically, the computational complexity of detecting a sub-symbol is $O(L_{sub} \log L_{sub})$. If the signal quality of an appliance is good enough, the receiver can use a shorter sub-symbol to detect its aliveness and thus reduce the computational cost. What's more, the receiver can go to sleep during the gaps between sub-symbols. If the computing resources are insufficient, the receiver can reduce the detection frequency by choosing a longer gap length.

In practice, the receiver sets sub-symbol length L_{sub} and gap length L_{gap} for each transmitter in the following way. (1) The receiver first sets the default sub-symbol length as 2s to ensure a nearly perfect detection accuracy (evaluated in Section 9.3). The receiver then on demand reduces the length of sub-symbols which have good signal quality. (2) After that, the receiver sets the gap length for each appliance based on its typical operating duration. For example, for the electric fan, which may run for hours, we choose a low heartbeat rate, e.g., one heartbeat every 30s. For the electric toothbrush, which only runs for several minutes each time, we choose a fine-grained heartbeat rate, e.g., one heartbeat every 5s.

7.2 Payload Upload

MotorBeat also enables small appliances to upload some payload bits to the smart speaker. Such ability allows them to upload more information (e.g., remaining battery life).

The idea is that time intervals of transmitted R-PWM symbols can be used to encode payload bits. As shown in Figure 10, the transmitter deliberately sends the same R-PWM symbols after different time intervals to encode N bits. The intervals $T_{interval}$ are represented as

$$T_{interval} = T_{min} + K \cdot \delta \tag{3}$$

where $K = 0, 1, 2, \ldots 2^N - 1$, δ denotes the time granularity, and T_{min} denotes the minimum symbol interval.

The receiver detects the heartbeats, and records their arrival timestamps. Time intervals are obtained by subtracting the symbol length[8] from heartbeat intervals. After that, payload bits are decoded from time intervals. In MotorBeat, to minimize the Bit Error Rate (BER) introduced by the error in time intervals, gray code is used to encode payload bits.

[8]In this application, the symbol length is a fixed value previously-known to both the transmitter and the receiver.

Next, we analyze the average data rate R_b of a MotorBeat link. We assume M symbols are sent during each round of uploading. The data rate can be calculated as

$$R_b = \frac{(M-1)\cdot N}{M \cdot L_{sym} + (M-1)\cdot \mathbf{E}[T_{interval}]} \tag{4}$$

where L_{sym} denotes the symbol length, and $\mathbf{E}[T_{interval}]$ denotes the expectation of $T_{interval}$, equaling $T_{min}+(2^N-1)\cdot\delta/2$. As we can see, $\mathbf{E}[T_{interval}]$ increases with N exponentially. With the increase of N, R_b is gradually dominated by N, and decreases exponentially. This indicates that the joint maximization of data rate should be conducted.

In practice, due to the low budget, small appliances only integrate a low-end MCU, which introduces a certain clock skew error. To compromise this problem, we resort to set a low time granularity, i.e., δ = 20 ms. Meanwhile, symbol length L_{sym} is 1 second to ensure a high detection accuracy. T_{min} is set to $L_{sym}/4$, and M is set to 4. Given those settings, N should be 5, yielding an optimal data rate of 3.21 bit/s.

8 PRACTICAL ISSUES

8.1 Configuration of Smart Speaker

MotorBeat requires the users to associate their small appliances with the smart speaker manually. Specifically, we envision a typical scenario as follows: Manufacturers print a QR code in the packing box of each small appliance. Each QR code contains the necessary configuration information, such as the R-PWM symbol's random seed, the parameters of the modulation and the encoding. When a user wants to associate a new appliance with the smart speaker, the user uses the phone to scan the QR code and then synchronize the configuration information to the smart speaker via WiFi or Bluetooth.

8.2 Multipath Effect

Here, we discuss the effect of multipath on MotorBeat. In Section 9.8, we will further evaluate it quantitatively.

For the application of heartbeat detection, the impact of the multipath effect is negligible. When the correlation is larger than a threshold, we assert a heartbeat is detected. In other words, we only care about the energy level of the signals. As long as the energy of one path is strong enough, we can detect heartbeats.

For the application of payload upload, multipath has a negative effect. The payload bits are embedded into the time interval between two symbols. It is possible that the receiver detects path A in the first symbol, but detects path B in the second symbol. In this case, the detected time interval is polluted by the time-difference-of-arrival (TDOA) between path A and B. MotorBeat adopts a relatively large time granularity of time interval (δ = 20 ms) to alleviate this problem.

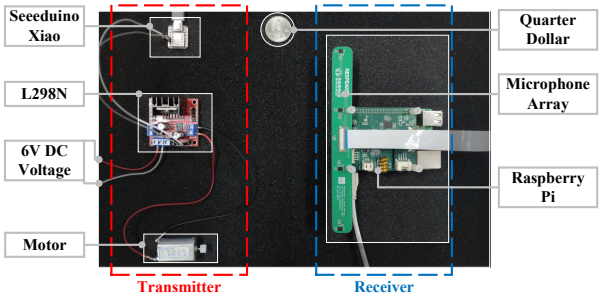


Figure 11: The prototype of MotorBeat.

The intuition is that the large time interval provides a wide guard interval that can tolerate the interference introduced by multipath propagation.

9 EVALUATION

9.1 Implementation

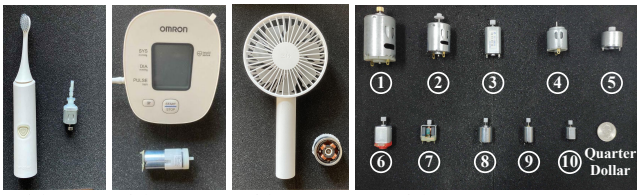
Hardware. Figure 11 shows the prototype of MotorBeat. We want to point out that without additional hardware, most small appliances can easily upgrade from PWM modulation to R-PWM modulation. This is because these two modulation methods have the same modulation mechanism: switching motors ON and OFF. The only difference is that PWM keeps a fixed switching period, while R-PWM randomizes the switching period. In other words, small appliances can reuse their original hardware to modulate their motors with R-PWM.

Since we have no access to small appliances' embedded software, we then directly drive their motors by connecting them to our transmitter prototype. We use a Seeeduino Xiao ($4.9) [41] and an L298N module ($1.6) to drive the motors. Specifically, the motors are powered by the L298N module. The voltage is 6V. The Seeeduino Xiao has two pins connected to the L298N module. The Seeeduino can switch the motors on and off by toggling the state of the pins. The R-PWM symbol's switching periods are uniformly distributed between 0.5ms and 2ms, smaller than the time constant of DC motors. The corresponding switching frequencies are between 500Hz and 2000Hz. To drive the motor to transmit messages, the Seeeduino toggles the pins according to its R-PWM symbol.

For the receiver, we use a ReSpeaker 4-mic linear microphone array [40] to act as a smart speaker to receive the audio[9], with a sampling rate of 24 kHz. The array is sitting on top of a Raspberry Pi 4 Model B, on which we run the software to detect the symbols and decode the payload bits.

Software. The signal processing is written in Python. We use the sliding-window method to process the audio stream chunk by chunk. To ensure that a sliding window can cover

[9]Currently, only one microphone is actually used in our implementation.



(a) TB (b) Fan (c) BPM (d) 10 DC motors

Figure 12: Three appliances and ten DC motors

the whole R-PWM symbol (or sub-symbol), we set the window length to $1.25L_{sym}$ (or $1.25L_{sub}$). The sliding windows are overlapped because the symbol (or sub-symbol) boundary is unknown. The sliding size is $0.25L_{sym}$ (or $0.25L_{sub}$). To detect all prior-known R-PWM symbols (or sub-symbols), we correlate each of them with each window of audio samples.

9.2 Experimental Methodology

Figure 12(a) shows three appliances used for evaluation: LangTian LT-G8 electric toothbrush ($10.4), ZMI AF215 mini fan ($9.1), and OMRON U10L blood pressure monitor ($30.6). For convenience, these three appliances are abbreviated as TB, Fan, and BPM, respectively. Their motors are also displayed. During evaluation, these motors are kept in their original positions. To further show that MotorBeat can be widely applied to DC motors, we use another ten different motors for evaluation, shown in Figure 12(d). The average price of these motors is $0.39. Our experiments are conducted in a classroom. The background noise is around 40-48 dB.

In the following, we first evaluate the overall performance in terms of the accuracy of detecting heartbeats, and bit error rate (BER) of decoding payload bits (Section 9.3). Then, we use ten motors to evaluate MotorBeat (Section 9.4). To present **3C** features, we conduct head-to-head experiments and deliver the results of each goal (Section 9.5). Furthermore, we evaluate the computing cost of the receiver on the Raspberry Pi (Section 9.6). We then evaluate MotorBeat in the mobile scenario (Section 9.7). Finally, the performance in different multipath cases are evaluated (Section 9.8).

9.3 Overall Performance

This subsection evaluates the performance of detecting heartbeats and decoding payload bits. Three appliances, TB, Fan, and BPM, are used for evaluation.

9.3.1 Heartbeat detection accuracy. Here, we study the impact of the distance and the symbol length on the detection accuracy. We place the receiver 0.5m, 2m, 6m, and 10m away from the transmitter. For each distance setting, we further vary the symbol length from 0.0625s to 2s to measure the accuracy of detecting heartbeats. Figure 13 shows the results.

By comparing Figure 13(a)-(d), we can find that the accuracy decreases as the distance increases. This is expected since the longer distance means more serious signal attenuation. To combat the problem of attenuation, we can increase the symbol length to extend the communication range. For example, when the symbol length increases to 2s, the detection performance is nearly perfect, and no error occurs in all distance settings. On the other hand, when the distance is no more than 2m, we can use symbols with shorter lengths, such as 0.5s or 1s, which also maintain a reasonable accuracy ($\geq$0.973). These results motivate the flexible design introduced in Section 7.1, which allows the receiver to tune the symbol length to trade off between the detection accuracy and the computational cost.

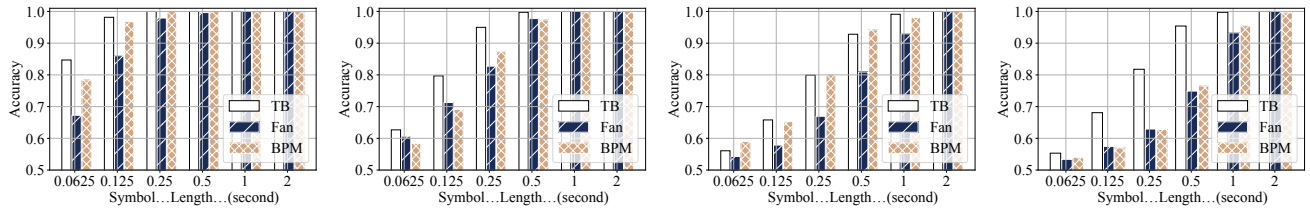
Apparently, the performance also varies across appliances. The main reasons are two-fold: (1) As we can see in Figure 12(a)-(c), the motors of these appliances are quite different. Naturally, the sounds generated by these motors and their signal quality are also different. (2) These appliances attach different gearboxes to their motors for different purposes, such as the toothbrush head for the TB, the blades for the Fan, and the air pump for the BPM. These gearboxes introduce varying degrees of noise. For example, the Fan's blades will generate significant aerodynamic noise. This explains why the Fan performs worst among these appliances.

9.3.2 Bit error rate. Here, we evaluate the performance of decoding payload bits (i.e., BER). We place the receiver 2m and 10m away from the transmitter, and then separately study how time resolution and symbol length affect BER.

Figure 14 shows the result with respect to time resolution δ. As mentioned in Section 7.2 and Section 8.2, a larger time resolution provides a wider guard interval and mitigates the timing error and multipath effect. We then vary the time resolution from 2.5ms to 40ms to evaluate BER. The symbol length is fixed to 1s. As expected, the BER decreases with the increase of the time resolution. When the time resolution is 40 ms, the payload of the TB can be accurately decoded. The BERs are only 0.00011 at 2m, and 0.0085 at 10m. Again, because different appliances generate different levels of noise, their decoding performance is also different.

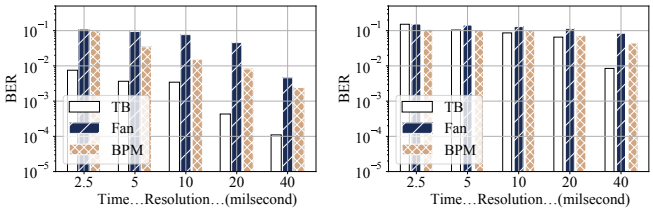
Figure 15 shows the result with respect to symbol length. With the increase of symbol length, the correlation peaks tend to be sharper. Therefore, the receiver can more precisely determine the arrival time of each symbol. We vary the symbol length from 0.0625 s to 2 s to measure BER. Meanwhile, the time resolution is fixed to 20 ms. We observe that the BER decreases as the symbol length increases. When the symbol length is less than 0.5s, the BERs of some appliances are higher than 0.1, which are unacceptable in practice.

When the symbol length increases to 2s, the problem is significantly alleviated: We can check that these BERs are not more than 0.054. As an example, the BER of the TB is 0.00024 and 0.023 at 2m and 10m, respectively.



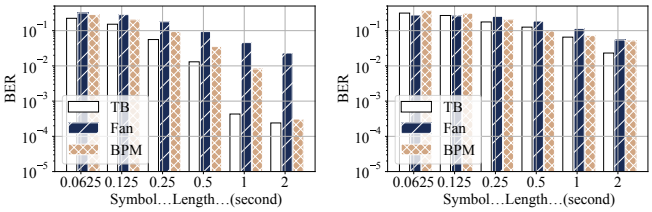
(a) Distance = 0.5 m (b) Distance = 2 m (c) Distance = 6 m (d) Distance = 10 m

Figure 13: Heartbeat detection accuracy vs. sub-symbol length in different distance settings.



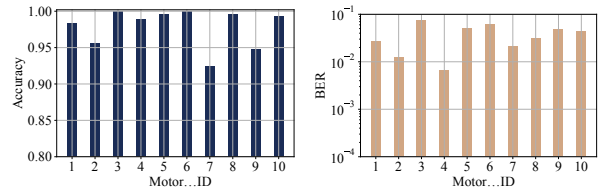
(a) Distance = 2 m (b) Distance = 10 m

Figure 14: BER vs. time resolution when symbol length is fixed to 1 second.



(a) Distance = 2 m (b) Distance = 10 m

Figure 15: BER vs. symbol length when time resolution is fixed to 20 milliseconds.



(a) Heartbeat accuracy (b) BER

Figure 16: Performance across 10 motors (Dist. = 6m)

Modulation method	Motor A	Motor B	Motor C
R-PWM (MotorBeat)	**4.2**	3.8	**4.5**
PWM (Standard)	4.1	**4.0**	**4.5**
Chirp (Bleep)	3.0	2.2	2.8

Table 2: The average accepted scores of the sounds generated by three modulation methods.

9.4 Performance across 10 DC Motors

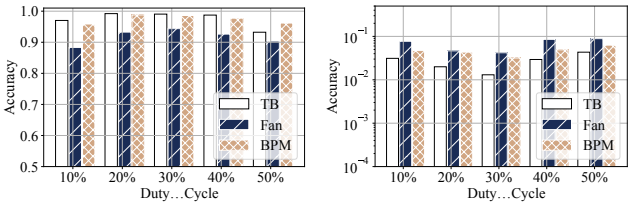
Here, we want to show the feasibility of applying MotorBeat to various DC motors. We evaluate MotorBeat's performance on 10 different motors shown in Figure 11(b). We place each motor 6m away from the receiver. Figure 16(a) shows the result of the heartbeat detection accuracy. In this experiment, the sub-symbol length is fixed to 1s. The heartbeats of these motors can be accurately detected. The worst accuracy (Motor 7) can still reach 0.924. Figure 16(b) shows the result of the BER. In this experiment, the symbol length and the time resolution are fixed to 1s and 20ms, respectively. All the BERs are less than 0.1, and the average BER across all motors is only 0.038.

9.5 3C Goals

We obtained the IRB approval to conduct the following experiments.

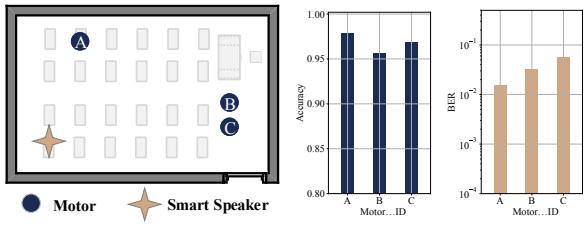
9.5.1 Comfortable. Here, we evaluate the acceptance of the sounds produced by MotorBeat. We compare MotorBeat with two other methods: standard PWM adopted by most appliances, and chirp-like modulation adopted by Bleep [6]. We invite 10 volunteers to join this experiment. We place each motor 0.5m away from volunteers, and drive the motor with three modulation methods separately. After hearing each sound, each volunteer is then asked to rate the acceptance: 5=acceptable, 4=slightly acceptable, 3=mild, 2=slightly unacceptable, and 1=unacceptable. Table 2 shows the average accepted scores of these sounds. The results show that Bleep seriously disturbs the users, and has the lowest acceptance among these three methods. Meanwhile, MotorBeat introduces no additional annoying sounds, and almost has the same acceptance as the standard PWM.

9.5.2 Compatible. The user may change the working mode of the appliances at will. The duty cycle of transmitted R-PWM symbols is unknown to the receiver. This experiment intends to show that even if the duty cycle of the symbol used



(a) Heartbeat accuracy (b) BER

Figure 17: Performance with varying duty cycles



(a) Deployment (8m x 5m) (b) Heartbeat (c) BER

Figure 18: Performance when 3 motors transmit concurrently.

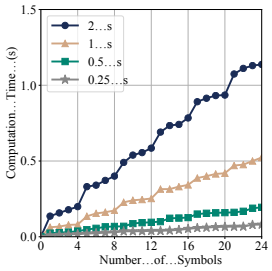
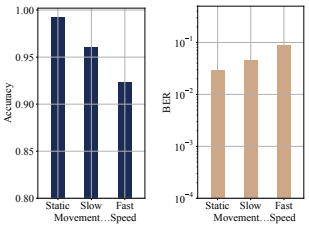
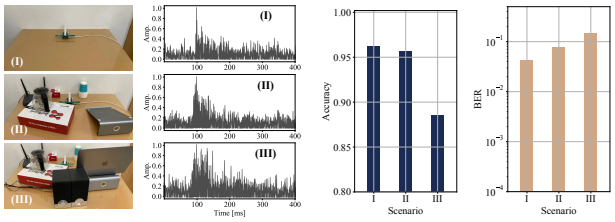


Figure 19: Computing time cost.



(a) Heartbeat (b) BER

Figure 20: Performance vs. speed.



(a) Three Scenarios (b) Heartbeat (c) BER

Figure 21: Performance vs. multipath severities

by the receiver is different from that of the symbols actually transmitted, the receiver can still detect these transmitted symbols. Each appliance is 6m away from the receiver, and repeatedly transmits its R-PWM symbols or sub-symbols (length = 1s) with a fixed duty cycle of 30%. The receiver detects these R-PWM symbols or sub-symbols with varying duty cycles: 10%, 20%, 30%, 40%, and 50%. Figure 17 (a) and (b) show the results. As expected, the receiver achieves the best accuracy when the duty cycle is 30%. This is because the symbol with duty cycle 30% perfectly matches the transmitted symbol. When the receiver uses symbols with other duty cycles, the performance only suffers a slight degradation.

9.5.3 Concurrent. Here, we evaluate the performance in the multi-source scenario. We place three motors (Motor 3, 6, and 8 in Figure 12) in a room with a size of 8m x 5m, as shown in Figure 18(a), and let them transmit simultaneously. The other settings are the same as those in Section 9.4. Figure 18 (b) and (c) show the results. In general, MotorBeat achieves a satisfactory performance. All heartbeat detection accuracies are above 0.95, and the worst BER is still less than 0.056.

9.6 Computational Efficiency

Here, we evaluate the computation efficiency of MotorBeat. We run the MotorBeat receiver on a Raspberry Pi 4B whose CPU has four cores. We then use four processes to compute correlations simultaneously. Figure 19 shows the computational time as a function of the number of symbols. We use a sliding window method to process the audio samples. In order to process the audio in real time, MotorBeat should detect all R-PWM symbols within the sliding size (0.25 L_{sym}). For example, for the 2s symbols, MotorBeat needs to detect them all within 0.5s. By referring to Figure 19, we find that MotorBeat takes 0.49s to detect 9 symbols. In other words, MotorBeat detects 9 symbols with a length of 2s in real time.

9.7 Impact of Mobility

Some small appliances, such as toothbrush, might not be stationary during usage, so this section studies how the movement of the transmitter affects MotorBeat. A volunteer is asked to stand 6m away from the receiver, and hold and move the toothbrush with three speeds: static, slow (about 0-0.2m/s), and fast (about 0.5-1m/s). In this experiment, the symbol (or sub-symbol) length is 1s, and the time resolution is 20 ms. Figure 20 shows that the performance decreases with the increase of speed. This is because the movement introduces an additional Doppler effect, which undermines the correlation between the symbol and the audio samples. The faster the transmitter moves, the more serious the Doppler effect will be. From our measurements, when the speed increases to 2m/s, the receiver will fail to detect the symbols. Nevertheless, the typical moving speed of small appliances are no more than 1 m/s at home.

9.8 Impact of Multipath

Here, we evaluate how multipath affects MotorBeat. We place the receiver in three multipath scenarios I, II, and III. Figure 21(a) displays these scenarios as well as their corresponding Channel Impulse Responses (CIRs)[10]. Apparently, Scenario I's channel is the cleanest, while in Scenario III, the line-of-sight path is completely blocked and its channel experiences the most severe multipath effect. In this experiment, the motor is 6m away from the receiver, the symbol (or sub-symbol) length is 1s and the time resolution is 20 ms.

Figure 21 (b) and (c) show the performances in terms of two applications, heartbeat detection and payload decoding, respectively. As expected, the performance degrades with the severity of multipath. However, the two applications have different sensitivities to the multipath effect. By comparing the results of Scenarios I and II, we can find that the heartbeat detection accuracy only suffers a slight degradation, from 0.963 to 0.957, while the BER experiences a relatively considerable increase, from 0.042 to 0.076. In the NLOS case (Scenario III), the performance degrades further with the accuracy of 0.885 and the BER of 0.147.

10 RELATED WORK

Motor-Based Communication. Bleep [6] is the work closest to ours. Bleep modulates the sounds of UAV motors to enable UAVs to communicate in the acoustic channel. Bleep increases or decreases the switching frequency of PWM voltage every 50ms to transmit acoustic up-chirp or down-chirp signals. To capture and decode the acoustic signals, Bleep equips each UAV with an additional microphone.

Ripple [37, 38] exploits vibration motors in smartphones to achieve motor-accelerometer communication. Different from MotorBeat which is based on DC motors, Ripple uses AC motors, Linear Resonant Actuators (LRAs). Ripple is thus allowed to modulate both the magnitude and frequency of vibration, and has much more modulation space. What's more, MotorBeat has more constraints than Ripple. This is because we need to maintain the original function of small appliances, while Ripple is free to drive the motor at will.

Acoustic or Vibration Side-Channel. Deaf-Aid [14] utilizes ultrasonic signals to make the gyroscope resonate and then to convey information. These side-channels are also used for device authentication [10, 29, 32, 53, 56], object identification [19], drone identification [35], and inter-vehicular communications[36]. Meanwhile, numerous prior works demonstrate that these side-channels can be used to recover information from input/output devices such as keyboards[4, 20, 30], printers [5], and screens [16, 24].

Timing-Based Encoding. ONPC [31] uses the timing of 802.11 frames to convey information. WiChronos [39] encodes information in the time interval between two narrowband symbols to achieve low-power communication. FreeBee [26] achieves cross-technology communication via embedding bits into beacons by shifting transmission timing. Encoding information within the beacon timing is also studied in optical and UWB communications[12, 15, 17]. Inspired by these works, we embed bits into the intervals between R-PWM symbols to upload bits.

Random Pulse Width Modulation. R-PWM techniques are originally designed for voltage-controlled power electronic converters [7, 28, 46]. Because R-PWM can disperse the concentrated energy of harmonics over a wide frequency range, many works utilize it to mitigate acoustic noise [8, 27], electromagnetic interference [45, 47], or mechanical resonance frequency [9].

11 LIMITATIONS AND FUTURE WORK

One-Way Communication. MotorBeat enables small appliances only to speak, not to listen. Therefore, the receiver cannot reply ACKs to these small appliances. Nevertheless, building this up-link from small appliances to the smart speaker is meaningful, because this up-link can enable many new applications. For example, small appliances are now able to broadcast their heartbeats and report their status.

Microphone Array and Beamforming. Most smart speakers contain multiple microphones. In our implementation, only one microphone is used. It is possible to further improve our performance by using the beamforming technique.

Localization and Motion Sensing. Recent works [1, 43, 51] demonstrate the feasibility of localizing sounds with a smart speaker. In our case, small appliances are also acoustic sources, and thus can be localized. What's more, the receiver can measure the CIR of each small appliance from the correlation function between the previously-known symbol and the audio samples, such as Figure 21(a). Therefore, a wide range of wireless sensing techniques [22, 48–50, 52, 55, 57] might be adopted to sense small appliances' motions.

12 CONCLUSION

A critical problem that impedes the development of smart home is that most home appliances are cut off from the Internet. To fill this gap, we propose MotorBeat, a novel motor-based communication approach. MotorBeat exploits DC motors, which widely exist in small appliances, to connect small appliances to the Internet. We hope that MotorBeat can pave the way to the vision of smart home, and open up a wide range of applications.

[10]Here, CIR is actually obtained from the correlation between the symbol and the audio samples.

REFERENCES

[1] 2021. MAVL: Multiresolution Analysis of Voice Localization. In *Proceedings of USENIX NSDI, Virtual Event, April 12–14, 2021.*

[2] Amazon. 2021. Amazon. https://www.amazon.com/.

[3] Amazon. 2021. Default categories of small appliances. https://www.amazon.cn/s/ref=nb_sb_noss?url=search-alias%3Dhome-appliances. Accessed: 2021-03-21.

[4] Dmitri Asonov and Rakesh Agrawal. 2004. Keyboard Acoustic Emanations. In *2004 IEEE Symposium on Security and Privacy, Berkeley, CA, USA, May 9-12, 2004,.*

[5] Michael Backes, Markus Dürmuth, Sebastian Gerling, Manfred Pinkal, and Caroline Sporleder. 2010. Acoustic Side-Channel Attacks on Printers. In *19th USENIX Security Symposium, Washington, DC, USA, August 11-13, 2010.*

[6] Adeola Bannis, Hae Young Noh, and Pei Zhang. 2020. Bleep: motor-enabled audio side-channel for constrained UAVs. In *Proceedings of ACM MobiCom, London, United Kingdom, September 21-25, 2020.*

[7] M. M. Bech, J. K. Pedersen, and F. Blaabjerg. 2001. Field-oriented control of an induction motor using random pulsewidth modulation. *IEEE Transactions on Industry Applications* 37, 6 (2001), 1777–1785.

[8] A. C. Binojkumar, J. S. S. Prasad, and G. Narayanan. 2013. Experimental Investigation on the Effect of Advanced Bus-Clamping Pulsewidth Modulation on Motor Acoustic Noise. *IEEE Transactions on Industrial Electronics* 60, 2 (2013), 433–439.

[9] Frede Blaabjerg, John K Pedersen, Ewen Ritchie, and Peter Nielsen. 1995. Determination of mechanical resonances in induction motors by random modulation and acoustic measurement. *IEEE transactions on industry applications* 31, 4 (1995), 823–829.

[10] Huangxun Chen, Wei Wang, Jin Zhang, and Qian Zhang. 2019. EchoFace: acoustic sensor-based media attack detection for face authentication. *IEEE Internet of Things Journal* 7, 3 (2019), 2152–2159.

[11] Yong Thung Cho. 2018. Characterizing sources of small DC motor noise and vibration. *MDPI Micromachines* 9, 2 (2018), 84.

[12] A. A. D'Amico, U. Mengali, and E. Arias-de-Reyna. 2007. Energy-Detection UWB Receivers with Multiple Energy Measurements. *IEEE Transactions on Wireless Communications* 6, 7 (2007), 2652–2659.

[13] Colin Dixon, Ratul Mahajan, Sharad Agarwal, A. J. Bernheim Brush, Bongshin Lee, Stefan Saroiu, and Paramvir Bahl. 2012. An Operating System for the Home. In *Proceedings of USENIX NSDI, San Jose, CA, USA, April 25-27, 2012.*

[14] Ming Gao, Feng Lin, Weiye Xu, Muertikepu Nuermaimaiti, Jinsong Han, Wenyao Xu, and Kui Ren. 2020. Deaf-aid: mobile IoT communication exploiting stealthy speaker-to-gyroscope channel. In *Proceedings of ACM MobiCom, London, United Kingdom, September 21-25, 2020.*

[15] Lijia Ge, Guangrong Yue, and Sofiene Affes. 2002. On the BER performance of pulse-position-modulation UWB radio in multipath channels. In *2002 IEEE Conference on Ultra Wideband Systems and Technologies, Baltimore, MD, USA, May 21-23, 2002.*

[16] Daniel Genkin, Mihir Pattani, Roei Schuster, and Eran Tromer. 2019. Synesthesia: Detecting screen content via remote acoustic side channels. In *2019 IEEE Symposium on Security and Privacy, San Francisco, CA, USA, May 19-23, 2019.*

[17] Zabih Ghassemlooy, AR Hayes, NL Seed, and ED Kaluarachchi. 1998. Digital pulse interval modulation for optical communications. *IEEE Communications Magazine* 36, 12 (1998), 95–99.

[18] Jacek F Gieras, Chong Wang, and Joseph Cho Lai. 2018. *Noise of polyphase electric motors.* CRC press.

[19] Taesik Gong, Hyunsung Cho, Bowon Lee, and Sung-Ju Lee. 2019. Knocker: Vibroacoustic-based object recognition with smartphones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–21.

[20] Tzipora Halevi and Nitesh Saxena. 2015. Keyboard acoustic side channel attacks: exploring realistic and security-sensitive scenarios. *International Journal of Information Security* 14, 5 (2015), 443–456.

[21] Austin Hughes and Bill Drury. 2013. *Electric Motors and Drives Fundamentals, Types and Applications (Fourth Edition).* Newnes.

[22] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of ACM MobiCom, New Delhi, India, October 29 - November 02, 2018.*

[23] Hongling Kang. 1995. The study of DC motor noise and vibration. *SAE transactions* (1995), 2461–2467.

[24] Hyosu Kim, Anish Byanjankar, Yunxin Liu, Yuanchao Shu, and Insik Shin. 2018. UbiTap: Leveraging Acoustic Dispersion for Ubiquitous Touch Interface on Solid Surfaces. In *ACM SenSys, Shenzhen, China, November 4-7, 2018.*

[25] Ji Eun Kim, George Boulos, John Yackovich, Tassilo Barth, Christian Beckel, and Daniel Mossé. 2012. Seamless Integration of Heterogeneous Devices and Access Control in Smart Homes. In *2012 Eighth International Conference on Intelligent Environments, Guanajuato, Mexico, June 26-29, 2012.*

[26] Song Min Kim and Tian He. 2015. Freebee: Cross-technology communication via free side-channel. In *Proceedings of ACM MobiCom, Paris, France, September 7-11, 2015.*

[27] R. L. Kirlin, S. F. Legowski, and A. M. Trzynadlowski. 1995. An optimal approach to random pulse width modulation in power inverters. In *Proceedings of Power Electronics Specialist Conference, Atlanta, GA, USA, June 18-22, 1995.*

[28] S. Legowski and A. M. Trzynadlowski. 1991. Advanced random pulse width modulation technique for voltage-controlled inverter drive systems. In *Proceedings of Sixth Annual Applied Power Electronics Conference and Exhibition, Dallas, TX, USA, March 10-15, 1991.*

[29] Jian Liu, Chen Wang, Yingying Chen, and Nitesh Saxena. 2017. VibWrite: Towards Finger-input Authentication on Ubiquitous Surfaces via Physical Vibration. In *Proceedings of ACM Conference on Computer and Communications Security, Dallas, TX, USA, October 30-November 03, 2017.*

[30] Jian Liu, Yan Wang, Gorkem Kar, Chen Yingying, Jie Yang, and Marco Gruteser. 2015. Snooping Keystrokes with mm-level Audio Ranging on a Single Phone. In *Proceedings of ACM MobiCom, Paris, France, September 7-11, 2015.*

[31] Philip Lundrigan, Neal Patwari, and Sneha Kumar Kasera. 2019. On-Off Noise Power Communication. In *Proceedings of ACM MobiCom , Los Cabos, Mexico, October 21-25, 2019.*

[32] Anil Madhavapeddy, Richard Sharp, David Scott, and Alastair Tse. 2005. Audio networking: the forgotten wireless technology. *IEEE Pervasive Computing* 4, 3 (2005), 55–60.

[33] Matthew A Nobile. 1985. Identifying prominent discrete tones in noise emissions. *The Journal of the Acoustical Society of America* 78, S1 (1985), S33–S33.

[34] Philips. 2021. what are the differences in modes of my philips sonicare toothbrush. https://www.usa.philips.com/c-f/XC000005380/what-are-the-differences-in-modes-of-my-philips-sonicare-toothbrush. Accessed: 2021-03-21.

[35] Soundarya Ramesh, Thomas Pathier, and Jun Han. 2019. SoundUAV: Fingerprinting Acoustic Emanations for Delivery Drone Authentication. In *Proceedings of ACM MobiSys, Seoul, Republic of Korea, June 17-21, 2019.*

[36] Sean Rowan, Michael Clear, Mario Gerla, Meriel Huggard, and Ciarán Mc Goldrick. 2017. Securing Vehicle to Vehicle Communications using Blockchain through Visible Light and Acoustic Side-Channels. *CoRR* abs/1704.02553 (2017).

[37] Nirupam Roy and Romit Roy Choudhury. 2016. Ripple II: Faster Communication through Physical Vibration. In *Proceedings of USENIX NSDI, Santa Clara, CA, USA, March 16-18, 2016.*

[38] Nirupam Roy, Mahanth Gowda, and Romit Roy Choudhury. 2015. Ripple: Communicating through Physical Vibration. In *Proceedings of USENIX NSDI, Oakland, CA, USA, May 4-6, 2015.*

[39] Yaman Sangar and Bhuvana Krishnaswamy. 2020. WiChronos: energy-efficient modulation for long-range, large-scale wireless networks. In *Proceedings of ACM MobiCom, London, United Kingdom, September 21-25, 2020.*

[40] Seeed. 2020. ReSpeaker 4-Mic Linear Array Kit for Raspberry Pi. https://wiki.seeedstudio.com/ReSpeaker_4-Mic_Linear_Array_Kit_for_Raspberry_Pi/. Accessed: 2021-03-21.

[41] Seeed. 2021. Seeeduino XIAO - Arduino Microcontroller - SAMD21 Cortex M0+. https://www.seeedstudio.com/Seeeduino-XIAO-Arduino-Microcontroller-SAMD21-Cortex-M0+-p-4426.html. Accessed: 2021-03-21.

[42] Chenguang Shen, Rayman Preet Singh, Amar Phanishayee, Aman Kansal, and Ratul Mahajan. 2016. Beam: Ending monolithic applications for connected devices. In *Proceedings of 2016 USENIX Annual Technical Conference,Denver, CO, USA, June 22-24, 2016.*

[43] Sheng Shen, Daguan Chen, Yu-Lin Wei, Zhijian Yang, and Romit Roy Choudhury. 2020. Voice localization using nearby wall reflections. In *Proceedings of ACM MobiCom, London, United Kingdom, September 21-25, 2020.*

[44] Statista. 2021. Household appliances unit sales in the United States from 2012 to 2025. https://www.statista.com/statistics/1173384/household-appliances-unit-sales-united-states/. Accessed: 2021-03-21.

[45] Andrzej Trzynadlowski, Mauro Zigliotto, Silverio Bolognani, and Michael Bech. 1998. Reduction of the electromagnetic interference conducted to mains in inverter-fed AC drives using random pulse width modulation. In *Conference Record of 1998 IEEE Industry Applications Conference, St. Louis, MO, USA, Oct 12-15, 1998.*

[46] A. M. Trzynadlowski, F. Blaabjerg, J. K. Pedersen, R. L. Kirlin, and S. Legowski. 1994. Random pulse width modulation techniques for converter-fed drive systems-a review. *IEEE Transactions on Industry Applications* 30, 5 (1994), 1166–1175.

[47] A. M. Trzynadlowski, K. Borisov, Yuan Li, Ling Qin, and Zhiqiang Wang. 2004. Mitigation of electromagnetic interference and acoustic noise in vehicular drives by random pulse width modulation. In *Power Electronics in Transportation.* 67–71.

[48] Anran Wang and Shyamnath Gollakota. [n.d.]. MilliSonic: Pushing the Limits of Acoustic Motion Tracking. In *Proceedings of ACMConference on Human Factors in Computing Systems, Glasgow, Scotland, UK, May 04-09, 2019.*

[49] Anran Wang, Jacob E Sunshine, and Shyamnath Gollakota. 2019. Contactless infant monitoring using white noise. In *Proceedings of ACM MobiCom, Los Cabos, Mexico, October 21-25, 2019.*

[50] Ju Wang, Hongbo Jiang, Jie Xiong, Kyle Jamieson, Xiaojiang Chen, Dingyi Fang, and Binbin Xie. 2016. LiFS: low human-effort, device-free localization with fine-grained subcarrier information. In *Proceedings of ACM MobiCom, New York City, NY, USA, October 3-7, 2016.*

[51] Weiguo Wang, Jinming Li, Yuan He, and Yunhao Liu. 2020. Symphony: localizing multiple acoustic sources with a single microphone array. In *Proceedings of ACM SenSys, Virtual Event, Japan, November 16-19, 2020.*

[52] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of ACM MobiCom,, New York City, NY, USA, October 3-7, 2016.*

[53] Wei Wang, Lin Yang, and Qian Zhang. 2016. Touch-and-guard: secure pairing through hand resonance. In *Proceedings of ACM UbiComp 2016, Heidelberg, Germany, September 12-16, 2016.*

[54] Wikipedia. 2021. Home Appliance. https://en.wikipedia.org/wiki/Home_appliance. Accessed: 2021-03-21.

[55] Yaxiong Xie, Zhenjiang Li, and Mo Li. 2015. Precise power delay profiling with commodity WiFi. In *Proceedings of ACM MobiCom, Paris, France, September 7-11, 2015.*

[56] Xiangyu Xu, Jiadi Yu, Yingying Chen, Qin Hua, Yanmin Zhu, Yi-Chao Chen, and Minglu Li. 2020. TouchPass: Towards Behavior-Irrelevant on-Touch User Authentication on Smartphones Leveraging Vibrations. In *Proceedings of ACM MobiCom, London, United Kingdom, September 21-25, 2020.*

[57] Fusang Zhang, Zhi Wang, Beihong Jin, Jie Xiong, and Daqing Zhang. 2020. Your Smart Speaker Can "Hear" Your Heartbeat! *Proceedings of ACM Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 161:1–161:24.