

# Attenuation Model for Error Correction of Ultrasonic Positioning System

|                                 |             |                                 |            |
|---------------------------------|-------------|---------------------------------|------------|
| Naoki Fujieda <sup>*a)</sup>    | Non-member, | Takumi Shinohara <sup>*</sup>   | Non-member |
| Shuichi Ichikawa <sup>*</sup>   | Member,     | Yuhki Sakaguchi <sup>*</sup>    | Non-member |
| Shunsuke Matsuoka <sup>**</sup> | Non-member, | Hideki Kawaguchi <sup>***</sup> | Member     |

(Manuscript received June 5, 2017, revised Sep. 20, 2017)

Previously, the authors had proposed an indoor positioning system based on multiple frequencies of ultrasonic waves. This system had a systematic positioning error caused by the attenuation of both ultrasonic waves in air and signals in the detection circuit. This study proposes an error correction method based on an attenuation model, along with directivity models of ultrasonic waves. According to our evaluation, the average error in positioning was reduced to 2.52 cm, which is 18% smaller than the simple correction obtained in the previous study. The additional calculation time required for the proposed model was 0.7–3.8 ms on Arduino Uno. This is an extended version of a work presented at an IEEJ technical meeting in March 2017.

**Keywords:** positioning system, ultrasonic wave, embedded system, microcontroller

## 1. Introduction

Positioning systems such as GPS<sup>(1)</sup> are widely used in various systems and services. GPS measures the propagation time of radio waves to estimate the distances between transmitters (satellites) and a receiver and calculate its position. However, very high precision is required in time measurement for accurate positioning because radio waves travel at the speed of light. Moreover, since radio waves from satellites are blocked by buildings, GPS is not suitable for indoor use.

To achieve highly precise positioning indoors, various systems and algorithms have been proposed<sup>(2)</sup>. Some systems still rely on radio waves but using signal strength<sup>(3)(4)</sup> instead of trilateration that is used by GPS. They do not require precise time measurement, while the accuracy is limited to a few meters. Ultra-wideband (UWB) signals can be used for highly accurate positioning based on trilateration<sup>(5)</sup>; however, the systems get complicated because precise synchronization is required. Positioning systems using light have also been studied. Some earlier systems rely on infrared signals<sup>(6)</sup>, while some recent systems utilize visible LED lighting infrastructure<sup>(7)</sup>. Also, computer vision technologies enabled positioning from images taken by cameras fixed in the room<sup>(8)</sup> or installed in the positioning targets<sup>(9)</sup>. The availability of these

systems, of course, heavily depends on the lighting environment. Hybrid systems adopt multiple technologies at the same time. Some of them use interpolation techniques, which are called pedestrian dead reckoning<sup>(10)</sup>, based on sensors in the positioning target.

We have proposed a low-cost, real-time indoor positioning system based on multiple frequencies of ultrasonic waves<sup>(11)(12)</sup>. Since ultrasonic waves travel much slower than radio waves, an accuracy of a microsecond order is sufficient for precise positioning and the effect of multi-path on the accuracy becomes smaller<sup>(13)(14)</sup>. To share the positioning space among transmitters previous studies use time division<sup>(13)-(15)</sup> or frequency diffusion<sup>(16)(17)</sup>. However, time division makes the positioning interval longer and frequency diffusion requires modulation operation, which compromise the real time and the simplicity of the system. The proposed system solves these problems by measuring the propagation times from different transmitters at the same time by different frequencies of ultrasonic waves.

According to our positioning experiments with prototyped systems, systematic positioning errors were observed<sup>(11)</sup> and one of the reasons for these was interference among multiple frequencies of ultrasonic waves<sup>(12)</sup>. After reducing interference, when a simple correction of measured distance or calculated position was applied, the average positioning error was reduced to within 10 cm<sup>(11)(18)</sup>. However, the correction methods used there were affine transform of the position and polynomial approximation of the distance, which were empirical and did not consider the mechanism that inserts such systematic errors.

In this paper, we propose an attenuation model for positioning systems with multiple frequencies of ultrasonic waves to achieve more precise positioning. It models attenuation of ultrasonic waves in the air and signals in a receiver circuit. In addition, it has some optional models for the directivity of ultrasonic transmitters and receivers, which may also cause

a) Correspondence to: Naoki Fujieda. E-mail: fujieda@ee.tut.ac.jp

\* Department of Electrical and Electronic Information Engineering, Toyohashi University of Technology  
1-1, Hibarigaoka, Tempakuchō, Toyohashi, Aichi 441-8580, Japan

\*\* Department of Mechanical Systems Engineering, National Institute of Technology, Asahikawa College  
2-1-6, Shunkodai 2 Jo, Asahikawa, Hokkaido 071-8142, Japan

\*\*\* College of Information and Systems, Muroran Institute of Technology  
21-1 Mizumotocho, Muroran, Hokkaido 050-8585, Japan

attenuation. We focus on the time lag between the arrival of a wave in the receiver and its detection by a microcontroller of the receiver. This lag can be estimated by parameter fitting with the proposed model. Corrected propagation time can be obtained by subtracting the estimated lag from the measured propagation time. We evaluated correction methods with the proposed model to show that the proposed methods reduce the average positioning error. We also evaluated their implementations to a microcontroller to confirm that they can be processed in real-time even with a low-cost microcontroller.

We presented a preliminary version of this work at an IEEJ technical meeting in March 2017<sup>(19)</sup>. The differences from the preliminary version include a directivity model using a Bessel function and optimizations of correction program in the prototyped system.

## 2. Positioning System with Multiple Frequencies of Ultrasonic Waves

**2.1 Summary of the System** Figure 1 depicts a positioning system using ultrasonic waves. Three or more ultrasonic transmitters are placed at the corners of the positioning space. In Fig. 1 and the following explanation, the number of transmitters is set to three. A transmitter  $n$  ( $n = 1, 2, 3$ ) emits ultrasonic waves of frequency  $f_n$ . They are placed at known positions: the position of the transmitter  $n$  is defined as  $(q_{nx}, q_{ny}, q_{nz})$ . A positioning target, located at  $(x, y, z)$ , has a receiver unit equipped with ultrasonic receivers corresponding to the transmitters. All transmitters and receivers are synchronized with each other via wireless communication.

A transmitter unit may be installed in the positioning target and receivers may be distributed to the positioning space in some systems<sup>(14)(16)</sup>. Although no generality is lost by swapping them in the following explanation, it should be noted that the positioning space must be shared by the transmitters. Such systems may be unsuitable for positioning a larger number of targets.

Transmitters start to emit ultrasonic waves at the same time. When the receiver unit detects the waves, it calculates the distance  $r_n$  between the corresponding transmitter and the receiver from the propagation time. After the distances from all of the transmitters are calculated, the position of the target can be obtained from the following simultaneous equations.

$$r_1^2 = (q_{1x} - x)^2 + (q_{1y} - y)^2 + (q_{1z} - z)^2 \dots \dots \dots (1)$$

$$r_2^2 = (q_{2x} - x)^2 + (q_{2y} - y)^2 + (q_{2z} - z)^2 \dots \dots \dots (2)$$

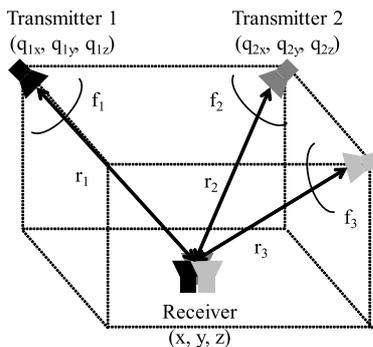


Fig. 1. Ultrasonic positioning model with three transmitters

$$r_3^2 = (q_{3x} - x)^2 + (q_{3y} - y)^2 + (q_{3z} - z)^2 \dots \dots \dots (3)$$

One of the characteristics of our positioning system<sup>(11)</sup> is that transmitters utilize different frequencies of ultrasonic waves from each other. This keeps our system simple and real-time, compared to using single frequency of time-divided<sup>(13)-(15)</sup> or frequency-diffused<sup>(16)(17)</sup> ultrasonic waves.

Though the simultaneous equations (1)–(3) can be numerically solved with an iterative method such as Newton’s method, alternative methods were proposed to calculate the position more quickly<sup>(11)</sup>. They add constraints on the position or the number of transmitters. One of the constraints is to align the height of the three transmitters such that  $q_{1z} = q_{2z} = q_{3z}$ . In this case, the equations can be simplified as follows:

$$2(q_{2x} - q_{1x})x + 2(q_{2y} - q_{1y})y = -r_2^2 + r_1^2 + q_{2x}^2 - q_{1x}^2 + q_{2y}^2 - q_{1y}^2 \dots \dots \dots (4)$$

$$2(q_{3x} - q_{2x})x + 2(q_{3y} - q_{2y})y = -r_3^2 + r_2^2 + q_{3x}^2 - q_{2x}^2 + q_{3y}^2 - q_{2y}^2 \dots \dots \dots (5)$$

$$z = q_{1z} - \sqrt{r_1^2 - (q_{1x} - x)^2 - (q_{1y} - y)^2} \dots \dots \dots (6)$$

In this paper, a program that solves the equations (1)–(3) with Newton’s method is called `newton`, while a program that uses the simplified equations (4)–(6) is called `3sender`.

We have proposed two more programs that used four transmitters to reduce the complexity of calculation or ease the requirement of the receiver<sup>(11)</sup>. Also, more than four transmitters may be used. In this case, similarly to GPS, any combination of three (or four) receivers that detect ultrasonic can be used to calculate the position<sup>(11)</sup>. These enhanced programs are out of scope of this paper.

## 2.2 Detection of Ultrasonic Waves

Figure 2 outlines the flow of ultrasonic detection in the proposed positioning system. A transmitter starts to emit an ultrasonic square wave at time zero. The head of the wave arrives at a receiver at the time of flight (TOF)  $t_f$ . The received signal from the ultrasonic receiver is amplified and then shaped by a low pass filter (LPF). The shaped signal (called compared signal in the figure) is transformed into a sharp square pulse (TOF detecting pulse) by a comparator. The receiver’s microcontroller calculates the distance from the transmitter using the time  $t$ ,

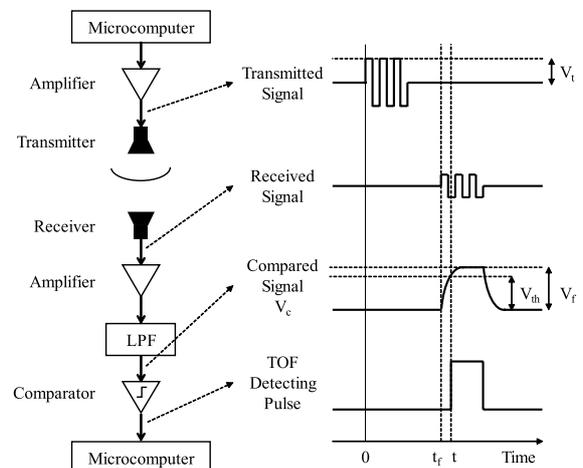


Fig. 2. Ultrasonic detection in our positioning system

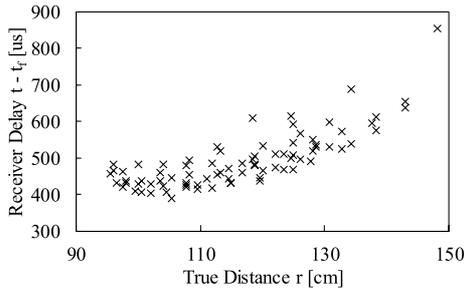


Fig. 3. Receiver delay in 25.0-kHz transmitter/receiver

or the time between the transmission of the wave and the detection of the TOF detecting pulse.

The point is that the propagation time  $t$  obtained by the receiver is not the same as the actual time of flight  $t_f$  and it includes the delay caused by the receiver  $t - t_f$ . Our early prototype<sup>(11)</sup> estimated  $t_f$  from a preliminary experiment. We measured  $t_z$ , the propagation time when placing the transmitter and the receiver as close as possible, and used it as the receiver’s delay. Using this correction, the distance  $r$  can be calculated as  $r = v(t - t_z)$ , where  $v$  is the speed of sound. When we conducted a positioning experiment using this prototype, large systematic errors were confirmed. In particular, the error tended to be large when the distance between the transmitter and the receiver was long.

In the second prototype<sup>(18)</sup>, the distance was corrected by assuming this tendency of errors as virtual reduction of the speed of sound. There, the distance  $r$  was assumed as a linear function of the propagation time  $t$  i.e.  $r = v't + c$ . The constants  $v'$  and  $c$  were estimated by the least-square method. According to a positioning experiment on this prototype in the range of 80 cm in width, 80 cm in length, and 117 cm in height, the average error in positioning was approximately 3.1 cm. However, this correction utilized only the tendency of measured data and the proposal of more accurate correction methods remained as a challenge. The error may be further reduced by considering the attenuation of ultrasonic waves and the circuit characteristics of the system.

Figure 3 plots the delay by the 25.0-kHz receiver  $t - t_f$  in a positioning experiment using the second prototype, which is also used in the evaluation in Section 5. Since the true position of the receiver is known in the experiment, the receiver delay is calculated from the true distance between the transmitter and the receiver. In this plot, the correction in the first prototype means they are approximated by a line parallel to the x-axis. The correction in the second prototype is expressed as applying a linear approximation to them. However, the relationship between the distance and the receiver delay is obviously not linear. A new delay model should explain both the behavior of the system and the tendency of the receiver delay.

### 3. Attenuation Model

#### 3.1 Overview

An attenuation model proposed in this paper focuses on the compared signal, the output of the LPF. The voltage of the compared signal  $V_c$  can be considered as a circuit transient in a first-order LPF. There,  $V_c$  is expressed as the following function of time:

$$V_c = V_f(1 - e^{-t/\tau}), \dots \dots \dots (7)$$

where  $V_f$  is the input voltage of the LPC and  $\tau$  is a time constant, which is determined by the resistance and capacitance of the filter. The delay caused by the filter  $t_d$  is defined as the time for  $V_f$  to reach the threshold voltage of the comparator  $V_{th} (< V_f)$ . Solving (7) for  $t$ , we obtain the following equation:

$$t_d = -\tau \log_e \left( 1 - \frac{V_{th}}{V_f} \right), \dots \dots \dots (8)$$

The equation (8) implies that a large filter delay occurs when the distance between the transmitter and the receiver is close to the maximum distance where ultrasonic waves can be detected (i.e.  $V_f \approx V_{th}$ ). Considering delay caused by other sources  $t_o$ , such as processing in the microcontroller, the delay caused by receiver becomes  $t_d + t_o$ , and thus the time of flight  $t_f$  is estimated as

$$t_f = t - t_d - t_o, \dots \dots \dots (9)$$

The corrected distance is obtained from  $vt_f$ .

This model also implies two limitations of our system. First, if the distance between the transmitter and the receiver is too long, the ultrasonic cannot be detected because it is too much attenuated and  $V_f$  gets smaller than  $V_{th}$ . Second, when an obstacle is placed between the transmitter and the receiver, the apparent distance between them might get longer. While we are developing another prototype to deal with these limitations, this paper only focuses on the correction of distance.

#### 3.2 Modeling of Filter Delay

This section discusses the effect of attenuation and directivity to the input voltage of the LPF  $V_f$ . Though attenuation of ultrasonic waves is caused by both diffusion and absorption<sup>(20)</sup>, the proposed model considers diffusion only. By diffusion, the intensity of sound in the receiver becomes proportional to  $1/r^2$  and the receiver voltage  $V_f$  is proportional to  $1/r$  (if the gain of the amplifier is constant). Since the threshold voltage of the comparator  $V_{th}$  will be fixed in adjusting the receiver circuit,  $V_{th}/V_f \propto r$  is obtained.

Ultrasonic transmitters and receivers have directivity; the voltage that receivers obtain may be reduced when a transmitter-receiver pair do not properly face each other. Our optional models for directivity, which will be described in the following section, estimate the relative voltage considering the directivity of the transmitter and the receiver as  $k_t$  and  $k_r$ , respectively. If no directivity models are applied, both  $k_t$  and  $k_r$  are set to 1. Since it is expressed as  $V_f \propto k_t k_r$ , we obtain  $V_{th}/V_f \propto r/(k_t k_r)$ . Along with equations (8) and (9),  $t_f$  can be estimated as the following equation:

$$t_f = t + \tau \log_e \left( 1 - \frac{Kr}{k_t k_r} \right) - t_o, \dots \dots \dots (10)$$

where  $K$  is a proportional constant.

However, the distance  $r$  in the equation (10) is exactly what we are going to calculate. It must be either approximately given or iteratively calculated. In the proposed model, this  $r$  is approximated as  $v(t - t_o)$  i.e. the distance obtained when the filter delay  $t_d$  is set to zero. Therefore, the final model for  $t_f$  becomes:

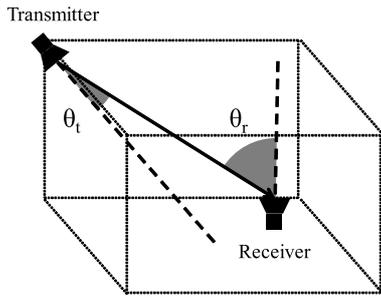


Fig. 4. Angle of transmitter  $\theta_t$  and receiver  $\theta_r$

$$t_f = t + \tau \log_e \left\{ 1 - \frac{Kv(t - t_o)}{k_t k_r} \right\} - t_o \dots \dots \dots (11)$$

The two constants of the equation,  $K$  and  $t_o$ , are given from parameter fitting with measured propagation times at known points. To be more specific, the true value of  $t_f$  can be obtained at known points because the distance between the transmitter and the receiver  $r$  is also known. Thus, the difference from  $t_f$  estimated by (11) can be calculated. The constants are determined so that the sum of the square of the differences can be minimized.

**3.3 Modeling of Directivity** This section describes the directivity models to estimate  $k_t$  and  $k_r$ . In our system, a transmitter and a receiver do not always face each other because their mounting angles are fixed. Figure 4 shows the mounting angles in our positioning experiment. Transmitters are pointed toward the center of the floor, while receivers face upwards. We define  $\theta_t$  ( $\theta_r$ ) as the angle between the direction of the receiver (transmitter) as seen from transmitter (receiver) and the direction that the transmitter (receiver) heads. If these angles are large, the received voltage will become smaller. The directivity models describe the relationship between the angle  $\theta_t$  ( $\theta_r$ ) and the relative voltage  $k_t$  ( $k_r$ ), which is collectively represented by a function  $k(\theta)$ .

We attempted a preliminary experiment to measure directivity. A transmitter-receiver pair was placed 100 cm apart, facing each other in the beginning. To measure the transmitter's directivity, the transmitter was rotated by a certain angle at that position. After transmitting ultrasonic waves for a sufficiently long time, the rotation angle and the voltage of the output of the LPF were recorded. The receiver's directivity was also measured in a similar way. The rotation angle was set from  $-90$  to  $90$  degrees in increments of 10 degrees. We undertook this experiment twice; the transmitter or the receiver is rotated horizontally in the first experiment and vertically in the second. The relative voltage is calculated by averaging the results of two experiments. Finally, a model formula for each transmitter or receiver is produced with parameter fitting of a base function.

We evaluate two functions for the directivity models. The first function is a fourth degree polynomial. It is defined with five constants as follows:

$$k(\theta) = a\theta^4 + b\theta^3 + c\theta^2 + d\theta + e \dots \dots \dots (12)$$

The second function is based on the directivity index for a circular piston source<sup>(20)</sup>, which is calculated from the following equation:

$$R_c(\theta) = \left| \frac{2J_1(\kappa \sin \theta)}{\kappa \sin \theta} \right| \dots \dots \dots (13)$$

where  $\kappa$  is a constant determined from the radius of the source and the wavelength and  $J_1$  is the Bessel function of the first kind with an order of 1. Since the received voltage is sometimes saturated,  $k$  is estimated with the following equation:

$$k(\theta) = \min(AR_c(\theta), 1) \dots \dots \dots (14)$$

where  $A$  is another parameter regarding the amplification, determined by parameter fitting along with  $\kappa$ . In this paper, a directivity model using equation 12 and 14 are called *Poly* and *Bessel*, respectively.

**4. Correction Program and its Optimization**

The microcontroller in the receiver calculates its position with the following steps.

- Step 1** measure the propagation time  $t$  for every transmitter-receiver pair.
- Step 2** estimate the distance from the transmitter. Equation 11 is used when the proposed model is applied. Directivity is not considered at this time: both  $k_t$  and  $k_r$  are set to 1.
- Step 3** calculate the receiver's position with a positioning program such as *newton* and *3sender*.  
If a directivity model is applied, the position calculated in Step 3 is provisional to obtain the angles. The following additional steps are required in this case.
- Step 4** calculate  $\theta$  and  $k(\theta)$  for each transmitter and receiver from the provisional position.
- Step 5** estimate the distance again using  $k_t$  and  $k_r$  obtained through Step 4.
- Step 6** calculate the receiver's position again. A different positioning program from Step 3 may be used.

These steps can be repeated multiple times. Figure 5 shows the flowchart of the positioning program. Since Step 3 and Step 6 are basically identical, the actual program repeats the calculation of the provisional position and correction of the distance from it for a predefined number of times.

A straightforward way to obtain the value of  $k(\theta)$  (Step 4) is divided into (1) finding  $\cos \theta$  using the cosine theorem, (2) calculating  $\theta$  with the arccosine function, and (3) assigning  $\theta$  to the model formula. However, an inverse trigonometric function sometimes takes too much time for a microcontroller. For example, in an Arduino Uno R3 microcontroller board (used in our evaluation of Section 6), it took 176 microseconds on average, which meant it would take about 1 ms to simply calculate  $\theta$  of three transmitters and three receivers.

We consider two optimization techniques to avoid the use of the arccosine. The first optimization is applicable for the *Bessel* model, where  $\sin \theta$  is actually required to obtain  $k$ . We can simply calculate  $\sin \theta$  from  $\sin \theta = \sqrt{1 - \cos^2 \theta}$ . We call this optimization *sin*. The second optimization can be used for both directivity models. It directly approximates  $k$  from  $\cos \theta$  with a table lookup and a linear interpolation. According to our trial calculations, a table of 65 elements for each formula (i.e.  $\cos \theta = 0, 1/64, 2/64, \dots$ ) is sufficient to obtain the accuracy of decimal three digits. If each element is a 32-bit floating point, it requires 1,560 bytes of data to store the approximation of formulae for three pairs of transmitters and receivers. We call this optimization *tbl*.

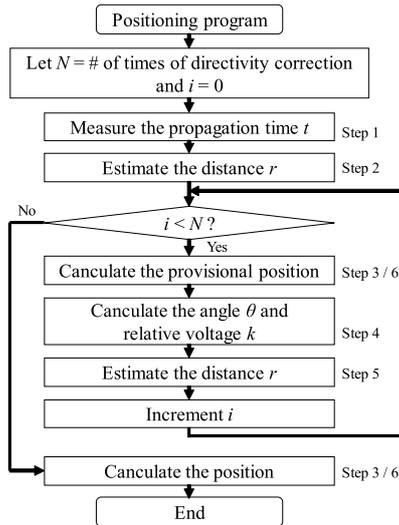


Fig. 5. Flowchart of the positioning program of the system

```

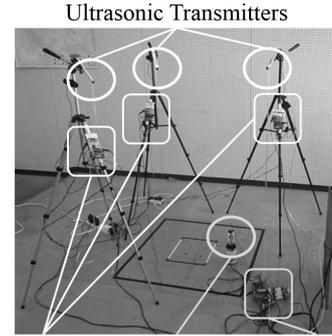
1  const float k_table[6][65] PROGMEM = {
2      // 25 kHz transmitter
3      {0.73290300, 0.74727691, 0.76114752, ...},
4      // 32 kHz transmitter
5      {0.58134780, 0.60099066, 0.61999675, ...},
6      ...};
7
8  void calc_directivity(...)
9  {
10     float costheta[6], k[6];
11     float fract, ki0, ki1;
12     int i, index;
13
14     ... // costheta is calculated here
15
16     for (i = 0; i < 6; i++) {
17         index = (int) (costheta[i] * 64);
18         fract = costheta[i] * 64 - index;
19         ki0 = pgm_read_float(&k_table[i][index]);
20         ki1 = pgm_read_float(&k_table[i][index + 1]);
21         k[i] = ki0 * (1 - fract) + ki1 * fract;
22     }
23 }
    
```

Fig. 6. Abstract of the program of Step 4 with the *tbl* optimization

Figure 6 shows a part of the C program of the *tbl* optimization for an AVR microcontroller. The table *k\_table* stores pre-calculated values of  $k(\cos^{-1}(0))$ ,  $k(\cos^{-1}(1/64))$ , ..., and  $k(\cos^{-1}(1))$  for all transmitters and receivers (in the lines 1–6). Since it is too large for an AVR microcontroller with very limited RAM space, it is stored in the program memory (PROGMEM). In the calculation,  $\cos\theta$  is multiplied by 64 and separated into the integer and the fractional parts (in the lines 17–18). The integer part is used as an index of table lookup (in the lines 19–20). The fractional part is used for a linear interpolation (in the line 21).

## 5. Evaluation of Positioning Accuracy

**5.1 Methodology** The evaluation of positioning accuracy of the proposed model used the data of the positioning experiment on the prototype in a paper<sup>(18)</sup>. Figure 7 shows a photograph of the prototype. The range of this experiment was 80 cm in width, 80 cm in length, and 117 cm in height. Each transmitter was mounted at a corner of the top, pointed to the center of the floor. The frequencies of ultrasonic



Transmitter Modules    Ultrasonic Receiver    Receiver Module

Fig. 7. Prototype of the positioning system used in the evaluation<sup>(18)</sup>

waves were 25.0 kHz, 32.7 kHz, and 40.0 kHz. Ultrasonic transducers by Nippon Ceramic Co., Ltd. were used. The model numbers of the transmitters were T2516A1, T3216A1, and T4016A2, and those of the receivers were R2516A1, R3216A1, and R4016A2, respectively. A receiver was placed at each of 9×9 grid points at 10 cm intervals with a height of 21.5 cm, facing upwards. The propagation time  $t$  was measured for each point. The position of the receiver was estimated from these data with the following correction methods:

**Fixed** a method using  $t_z$  as the receiver's delay<sup>(11)</sup>,

**Linear** a method that express the distance as a linear function of  $t$ , which was originally used in this positioning experiment<sup>(18)</sup>,

**DelayModel** the proposed method that estimates the receiver's delay using the attenuation model, where a directivity model was not applied,

**DelayModel+Poly** the DelayModel method with the *Poly* directivity model, and

**DelayModel+Bessel** the DelayModel method with the *Bessel* directivity model.

If a directivity model was applied, recalculation of the position (Steps 4–6) was repeated twice. The 3sender program with double-precision floating point numbers was always used to calculate position. The time constant of the LPF  $\tau$ , which is used in the DelayModel method, was estimated as 200 microseconds from a measurement with an oscilloscope, as its element values were not known.

**5.2 Fitting Results for Directivity** At first, parameters of the approximation formulae were extracted, which were shown in Tables 1 and 2 for the *Poly* and *Bessel* directivity models, respectively. In the leftmost column, a transmitter and a receiver are represented by their initial letters. All values in the tables were rounded off to three decimal places. Figure 8 plots the normalized receiver voltage measured by rotating a 25.0-kHz receiver, along with approximate curves with the directivity models. In the approximate curve with the *Poly* model,  $k_r$  got slightly larger than its baseline or 1.0. The curve of the *Bessel* model sometimes missed a plotted point for a large  $|\theta_r|$ . As seen above, although they were not complete, both basically gave a good approximation.

In addition, from the constant  $\kappa$  of the *Bessel* model, we obtained 3.4–6.8 mm of the radius of the circular piston source. Considering that the outside diameter of ultrasonic transducer used in our experiment was about 16 mm (8 mm in radius), it

Table 1. Parameters in the *Poly* directivity model

| Type | Freq.    | <i>a</i> | <i>b</i> | <i>c</i> | <i>d</i> | <i>e</i> |
|------|----------|----------|----------|----------|----------|----------|
| T    | 25.0 kHz | -0.076   | -0.002   | 0.080    | 0.010    | 0.991    |
| T    | 32.7 kHz | -0.094   | -0.002   | 0.054    | 0.018    | 0.997    |
| T    | 40.0 kHz | 0.153    | -0.052   | -0.706   | 0.086    | 1.110    |
| R    | 25.0 kHz | -0.057   | -0.004   | -0.158   | -0.011   | 1.036    |
| R    | 32.7 kHz | -0.162   | 0.007    | 0.167    | -0.020   | 0.982    |
| R    | 40.0 kHz | 0.054    | -0.072   | -0.316   | 0.073    | 1.058    |

Table 2. Parameters in the *Bessel* directivity model

| Type | Freq.    | $\kappa$ | <i>A</i> |
|------|----------|----------|----------|
| T    | 25.0 kHz | 3.076    | 3.714    |
| T    | 32.7 kHz | 2.860    | 2.398    |
| T    | 40.0 kHz | 3.276    | 1.804    |
| R    | 25.0 kHz | 3.127    | 2.267    |
| R    | 32.7 kHz | 3.594    | 8.466    |
| R    | 40.0 kHz | 2.521    | 1.517    |

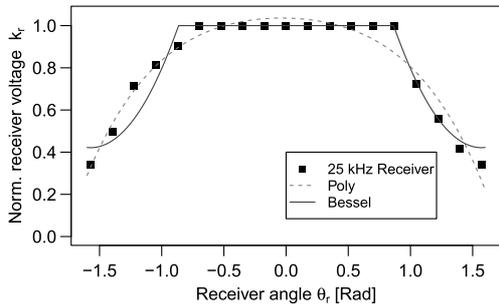


Fig. 8. Approximation results of the directivity of a 25.0-kHz receiver

Table 3. Fitting results of the proposed model

| Freq. [kHz] | DelayModel |                            | DelayModel+Poly |                            | DelayModel+Bessel |                            |
|-------------|------------|----------------------------|-----------------|----------------------------|-------------------|----------------------------|
|             | <i>K</i>   | <i>t</i> <sub>0</sub> [μs] | <i>K</i>        | <i>t</i> <sub>0</sub> [μs] | <i>K</i>          | <i>t</i> <sub>0</sub> [μs] |
| 25.0        | 0.00561    | 241.7                      | 0.00489         | 284.7                      | 0.00558           | 245.0                      |
| 32.7        | 0.00554    | 152.5                      | 0.00566         | 148.3                      | 0.00554           | 152.5                      |
| 40.0        | 0.00347    | 77.9                       | 0.00279         | 108.1                      | 0.00327           | 85.5                       |

also looked a reasonable approximation.

**5.3 Fitting Results for Delay** Table 3 summarizes the fitting results of the proposed attenuation model. In the table, the constant *K* is rounded off to five decimal places and the time *t*<sub>0</sub> is rounded to the nearest 0.1 microseconds. Equation (10) implies that, if  $Kr/k_t k_r \geq 1$ , *t*<sub>f</sub> has no solutions; that is, the attenuation is too large for the receiver to detect ultrasonic waves. As *K* is 0.00561 in *DelayModel*, the 25.0-kHz receiver will not detect waves when it leaves 178 cm or more from the transmitter without considering directivity. The range of the positioning experiment was determined in order that the receiver could detect waves everywhere in the range. The longest possible distance was about 163 cm, where the receiver was at the opposing corner to the transmitter. As the difference between them was not so large, the proposed attenuation model is generally reasonable.

When *K* of *DelayModel* is compared with the other models, *K* got slightly smaller by considering directivity except for the 32.7-kHz transmitter-receiver pair. This can be explained by the apparent increase of distance caused by the decrease of the received voltage by directivity.

**5.4 Positioning Accuracy** Table 4 shows the average, the standard deviation (SD), and the worst value of the positioning error using correction methods. The average error

Table 4. Evaluation results of positioning error

| Correction        | Avg. [cm] | SD [cm] | Worst [cm] |
|-------------------|-----------|---------|------------|
| Fixed             | 19.53     | 4.55    | 38.78      |
| Linear            | 3.07      | 1.77    | 12.54      |
| DelayModel        | 2.54      | 1.28    | 7.88       |
| DelayModel+Poly   | 2.57      | 1.40    | 8.01       |
| DelayModel+Bessel | 2.52      | 1.29    | 7.83       |

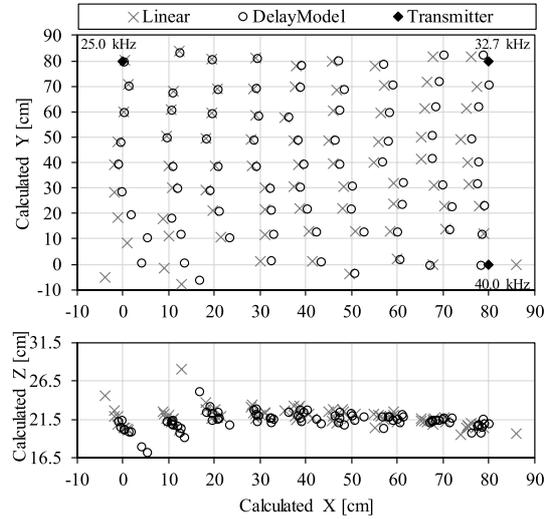


Fig. 9. Estimation results of positions after correction

was reduced to 2.52 cm in *DelayModel+Bessel*, which was 18% smaller than *Linear*, a simple correction utilized in the previous study.

Figure 9 plots the estimated position of the receiver for each placement after *Linear* or *DelayModel* correction. The x-axis represents the X coordinate of the estimated position, while the y-axes of the upper and the lower graphs corresponds to the Y and Z coordinates, respectively. The transmitters are placed on diamond points. From the placement of the receiver (shown in Section 5.1), if the position is correctly estimated, points will be plotted on the grids in the upper graph and on the line of 21.5 cm in the lower graph.

The positions estimated by *Linear* showed large errors near the bottom-left and bottom-right corners. This is because it underestimated the detection time of the 25.0-kHz or 32.7-kHz receiver when the receiver was far from the transmitter. Judging from the *K* in Table 3, the 25.0-kHz and 32.7-kHz transmitter-receiver pairs are more sensitive to attenuation than the 40.0-kHz pair. The proposed model mostly corrected the errors at these points, though errors in the opposite direction were observed at some of them, which might be due to an excessive correction.

On the other hand, the difference among directivity models in the positioning accuracy looked within the margin of error. The difference of corrected distance was also quite small (less than 1 cm at a maximum). This fact is interpreted as follows. In this positioning experiment, since the receiver was placed near the floor,  $\theta$  (especially  $\theta_t$ ) did not get so large. When it was placed near the top,  $\theta$  would be large but the distance between the transmitter and the receiver would be small enough. The effect of directivity would be emphasized in such cases. Further experiments to test this hypothesis are left for future work.

Table 5. Evaluation results of calculation time on Arduino UNO

| Correction              | newton    |            | 3sender   |            |
|-------------------------|-----------|------------|-----------|------------|
|                         | Time [ms] | Error [cm] | Time [ms] | Error [cm] |
| Linear <sup>(18)</sup>  | 4.42      | 2.98       | 0.30      | 3.07       |
| DelayModel              | 5.16      | 2.48       | 1.02      | 2.54       |
| DelayModel+Poly         | 9.27      | 2.52       | 5.13      | 2.57       |
| DelayModel+Bessel       | 10.01     | 2.47       | 5.89      | 2.52       |
| DelayModel+Bessel (sin) | 8.75      | 2.47       | 4.62      | 2.52       |
| DelayModel+Poly (tbl)   | 8.24      | 2.52       | 4.10      | 2.57       |
| DelayModel+Bessel (tbl) | 8.25      | 2.48       | 4.11      | 2.53       |

## 6. Calculation Time on Microcontroller

The evaluation of the calculation time is also important to keep the proposed positioning system real-time. This section evaluates the calculation time of positioning programs on a microcontroller. As well as previous prototypes<sup>(11)(18)</sup>, we used an Arduino Uno R3 microcontroller board, which includes Atmel ATmega328P that operates at 16 MHz. The capacity of the program memory and SRAM is 32 KiB and 2 KiB, respectively. Programs were written in C and compiled with Arduino IDE 1.8.1 (which includes GCC 4.9.2). Both *newton* and *3sender* programs were evaluated; however, when a directivity model was applied, a provisional position was calculated with *3sender*. The recalculation of the position was conducted only once. All floating point operations were done in single precision. In the *newton* program, the initial position was fixed at the center of the positioning space. Its calculation was repeated until the difference of the current position from the previous became less than 1 cm or the number of repetition reached a predefined limit (100).

Table 5 summarizes the evaluation results of calculation time. The columns “Time” mean the average of calculation time, while the columns “Error” show the average error in positioning. The name of the optimization technique is shown after the name of the model in parentheses, if applied. Compared *DelayModel* with *Linear*, about 0.7 ms of increase of the time was seen. An additional process for *DelayModel* was an estimation of the distance using Eq. (11). Most of the additional time might take in the calculation of logarithm. However, the average time taken by *DelayModel* with the *3sender* program was only 1.02 ms, which was small enough to keep the system real-time.

In the *3sender* program with a directivity model, Step 5 and 6 are almost identical to Step 2 and 3 (note that both Step 3 and Step 6 uses *3sender*). The time taken in Step 4 can be estimated by subtracting the time twice as long as *DelayModel* (i.e.  $1.02 \times 2 = 2.04$  ms). Without optimizations, Step 4 of the *Poly* and *Bessel* models took about 3.09 (=  $5.13 - 2.04$ ) ms and 3.85 (=  $5.89 - 2.04$ ) ms, respectively, which was almost as long as the *newton* program.

The *sin* optimization reduced it to 2.58 (=  $4.62 - 2.04$ ) ms in *Bessel*, which corresponded to 33% reduction. Simply eliminating the arccosine function was quite effective. This method also has an advantage of requiring little additional memory. The *tbl* optimization further reduced it to 2.06 (=  $4.10 - 2.04$ ) ms. This made the attenuation model with a directivity model possible to be argued that it was somewhat real-time. Also, the calculation time became easier to be predicted because floating-point operations in the optimized

correction only contained basic arithmetic and square root operations. This result almost matched the expected calculation time if addition, subtraction, and multiplication took 10 microseconds, and division and square root took 30 microseconds. However, if more complex calculation, such as iterative recalculation of the position (Step 4–6), was required, it would easily break the property of real-time. The use of a microcontroller faster than Arduino should be considered, though a concrete examination is left as future work.

## 7. Conclusion

We have proposed a real-time indoor positioning system using multiple frequencies of ultrasonic waves. In this paper, we proposed an attenuation model to correct the delay caused by the detection circuit of receiver, along with optional directivity models. We evaluated the model by both positioning accuracy and calculation time. According to the results, the average positioning error was reduced by 17%. The increase of the calculation time on an Arduino Uno board was 0.7–3.8 ms.

Although we proposed two directivity models, none of them showed significant effect on the positioning accuracy in the range of current experimental data. Our future work includes a further experiment, where the effect of directivity will be emphasized, to develop a practical indoor positioning system. The correction of error due to attenuation becomes more important there. If the positioning space gets large, there will be more places where ultrasonic waves must be detected under a low signal-to-noise ratio. Comparison of the characteristics of the system with other ultrasonic positioning system (e.g. using frequency diffusion, which is usually robust over noise) should be made. In addition, we are going to examine the use of a faster microcontroller in the same price range to keep the property of real time.

### Acknowledgment

This study was partially supported by JSPS Grants-in-Aid for Scientific Research (KAKENHI), Grant Number 16K00072.

## References

- (1) B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins: “Global Positioning System: Theory and Practice”, 5th ed., Springer-Verlag (2001)
- (2) K. Al Nuaimi and H. Kamel: “A survey of indoor positioning system and algorithms”, Proc. 2011 International Conference on Innovations in Information Technology, pp.185–190 (2011)
- (3) P. Bahl and V.N. Padmanabhan: “RADAR: An In-Building RF-based User Location and Tracking System”, Proc. 19th Annual IEEE Conference on Computer Communications, pp.776–785 (2000)
- (4) A. Goswami, L.E. Ortiz, and S.R. Das: “WiGEM: A Learning-Based Approach for Indoor Localization”, Proc. 7th Conference on Emerging Networking Experiments and Technologies, pp.3:1–3:12 (2011)
- (5) B. Kempke, P. Pannuto, and P. Dutta: “PolyPoint: Guiding Indoor Quadrotors with Ultra-Wideband Localization”, Proc. 2nd International Workshop on Hot Topics in Wireless, pp.16–20 (2015)
- (6) R. Want, A. Hopper, V. Falcao, and J. Gibbons: “The Active Badge Location System”, *ACM Transactions on Information Systems*, Vol.10, No.1, pp.91–102 (1992)
- (7) N. Ul Hassan, A. Naeem, M.A. Pasha, T. Jadoon, and C. Yuen: “Indoor Positioning Using Visible LED Lights: A Survey”, *ACM Computing Surveys*, Vol.48, No.2, pp.20:1–20:32 (2015)
- (8) J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer: “Multi-Camera Multi-Person Tracking for EasyLiving”, Proc. 3rd IEEE International Workshop on Visual Surveillance, pp.3–10 (2000)

- (9) M. Werner, M. Kessel, and C. Marouane: "Indoor Positioning using Smartphone Camera", Proc. 2011 International Conference on Indoor Positioning and Indoor Navigation, pp.1–6 (2011)
- (10) S. Beauregard and H. Haas: "Pedestrian Dead Reckoning: A Basis for Personal Positioning", Proc. 3rd Workshop on Positioning, Navigation, and Communication, pp.27–35 (2006)
- (11) S. Matsuoka, N. Fujieda, S. Ichikawa, and H. Kawaguchi: "Development of Real-time Positioning System by ultrasonic waves", *Journal of the Japan Society of Applied Electromagnetics and Mechanics*, Vol.23, No.2, pp.380–385 (2015) (in Japanese)
- (12) S. Matsuoka, Y. Sakaguchi, T. Shinohara, N. Fujieda, S. Ichikawa, and H. Kawaguchi: "Improved real-time positioning system by ultrasonic waves of different frequencies", 24th MAGDA Conference in Tohoku, No.PS5 (2015) (in Japanese)
- (13) M. Akiyama, H. Sunaga, S. Ioroi, and H. Tanaka: "Composition and Verification Experiment for Indoor Positioning System using Ultrasonic Sensors", *Journal of the Institute of Positioning, Navigation and Timing of Japan*, Vol.3, No.1, pp.1–8 (2012) (in Japanese)
- (14) U. Yayan, H. Yucel, and A. Yazici: "A Low Cost Ultrasonic Based Positioning System for the Indoor Navigation of Mobile Robots", *Journal of Intelligent & Robotic Systems*, Vol.78, No.3, pp.541–552 (2015)
- (15) N.B. Priyantha: "The cricket indoor location system", Ph.D. Thesis, MIT (2005)
- (16) M. Hazas and A. Hopper: "Broadband Ultrasonic Location Systems for Improved Indoor Positioning", *IEEE Transactions on Mobile Computing*, Vol.5, No.5, pp.536–547 (2006)
- (17) A. Suzuki, T. Iyota, S. Usami, and K. Watanabe: "Realtime Correlation Calculation for Distance Measurement Using Spread Spectrum Ultrasonic Waves", *Transaction of the Society of Instrument and Control Engineers*, Vol.46, No.7, pp.357–364 (2010) (in Japanese)
- (18) Y. Sakaguchi, S. Matsuoka, N. Fujieda, S. Ichikawa, and H. Kawaguchi: "Improvement of Positioning System by Ultrasonic Waves", IEICE General Conference 2015, No.D–6–15 (2015) (in Japanese)
- (19) T. Shinohara, N. Fujieda, S. Ichikawa, Y. Sakaguchi, S. Matsuoka, and H. Kawaguchi: "A proposal of the error correction method based on the circuit characteristics of ultrasonic positioning system", IEEJ Technical Meeting Document IIS–17–025 (2017) (in Japanese)
- (20) L.E. Kinsler, A.R. Frey, A.B. Coppens, and J.V. Sanders: "Fundamentals of Acoustics", 4th ed., John Wiley & Sons (1999)

**Naoki Fujieda** (Non-member) received his D.E. degree in 2013 from the Department of Computer Science of Tokyo Institute of Technology. Since 2013, he is an assistant professor of the Department of Electrical and Electronic Information Engineering of Toyohashi University of Technology. His research interests include processor architecture, applied FPGA systems, embedded systems, and secure processors. He is a member of IPSJ, IEICE, and IEEE.



**Takumi Shinohara** (Non-member) received his M.E. degree in 2017 from the Department of Electrical and Electronic Information Engineering of Toyohashi University of Technology. Since April 2017, he is affiliated with Total Technical Solutions K.K.



**Shuichi Ichikawa** (Member) received his D.S. degree in Information Science from the University of Tokyo in 1991. He has been affiliated with Mitsubishi Electric Corporation (1991–1994), Nagoya University (1994–1996), Toyohashi University of Technology (1997–2011), and Numazu College of Technology (2011–2012). Since 2012, he is a professor of the Department of Electrical and Electronic Information Engineering of Toyohashi University of Technology. His research interests include parallel processing, high-performance computing, custom computing machinery, and information security. He is a member of IEEJ, IEEE, ACM, IEICE, and IPSJ.



**Yuhki Sakaguchi** (Non-member) received his B.E. degree in 2016 from the Department of Electrical and Electronic Information Engineering of Toyohashi University of Technology. Since April 2016, he is affiliated with Aisan Industry Co., Ltd.



**Shunsuke Matsuoka** (Non-member) received his Ph.D. degree in engineering from Muroran Institute of Technology in 2006, and is presently an assistant professor at National Institute of Technology, Asahikawa College. He has worked on development of ultrasonic positioning system. IEICE society member.



**Hideki Kawaguchi** (Member) received the B.S., M.S. degrees and Ph.D. in Electrical Engineering from Hokkaido University in 1986, 1988, and 1994, respectively. During 1988–1990, he worked for Nippon Telegraph and Telephone Co. and for Hokkaido University during 1991–1998. He is now with Muroran Institute of Technology.

