

Whistle: Synchronization-Free TDOA for Localization

Bin Xu*, Ran Yu*, Guodong Sun*, Zheng Yang†

**Department of Computer Science and Technology
Tsinghua University, Beijing, China*

† *TNLIST, School of Software, Tsinghua University*

Email: {xubin,sgdcs}@tsinghua.edu.cn; {yuran00,hmilyyz}@gmail.com

Abstract

Localization is of great importance in mobile and wireless network applications. TDOA is one of the widely used localization schemes, in which a to-be-located object emits a signal and a number of receivers record the arriving time of the signal. By calculating the time difference of different receivers, the location of the object is estimated. In such a scheme, receivers must be precisely synchronized; even slight noises are completely unacceptable for centimeter-level localization. Previous studies have shown that existing time synchronization approaches for low-cost devices are insufficiently accurate and basically infeasible for high accuracy localization. In our scheme (called Whistle), several asynchronous receivers record a target signal and a successive signal that is generated artificially. By two-signal sensing and sample counting techniques, high time resolution can be achieved. This design fundamentally changes TDOA in the sense of releasing the synchronization requirement and avoiding many sources of inaccuracy. We implement Whistle on commercial off-the-shelf (COTS) cell phones. Through extensive real-world experiments in indoor and outdoor, quiet and noisy environments, the mean error is 10~20 centimeters in a $9 \times 9 \times 4m^3$ 3D space.

1. Introduction

The proliferation of wireless and mobile devices has fostered the demand for context-aware applications, in which location is often viewed as one of the most significant contexts. To know the location of a wireless device, a number of ranging techniques are developed, including Received Signal Strength (RSS), Time of Arrival (TOA), Time Difference of Arrival (TDOA), and so on. Among them, high accuracy localization can be achieved by measuring TDOA information of acoustic or radio signals. In existing TDOA systems,

an object to be located emits a signal and a number of receivers at fixed positions record the arriving time of the signal. By calculating the time difference of different receivers, the location of the object can be further determined.

Actually there is another category of localization methods in the literature also named “TDOA”, which uses a combination of signals of different propagation speeds. Taking Cricket [24] as a typical example, a sender emits ultrasound and RF signals simultaneously; while a receiver records the arrival time of them. The difference of two timestamps can be further used to calculate the sender-receiver distance. Different from multi-modal signal TDOA, in this study, we consider time difference as the difference of the arrival times of a same signal at different receivers [7], [8], [10], [31], [33]. In such a scheme, time synchronization among receivers is needed in previous approaches.

Many synchronization solutions for low-cost wireless devices are designed, such as RBS [6], TPSN [11], FTSP [20], GTSP [29], providing a common timescale for local clocks of nodes in a network for many applications. Generally, synchronization is obtained based on a series of beacon message exchanges. When a node generates a timestamp and sends it to another node for synchronization, unfortunately, the packet carrying the time-stamp often suffers from several kinds of delay before it reaches the receiver and is decoded. The delay prevents the receiver from correctly setting its local clock. Fikret et al. [27] decomposes the time uncertainties into four components:

- Send time: Time spent to generate a beacon message at the sender.
- Access time: Delay at the medium access control (MAC) layer before actual transmission.
- Propagation time: Time spent for a message to be transmitted in the air from a sender’s network interface to a receiver’s.
- Receive time: Time needed for a receiver to

receive and decode the message, and report it to upper layer applications.

Some synchronization solutions can remove one or two components above, but not all of them [11] [29]. What's worse, even a solution that can handle the four components properly is still not perfect due to clock skew and drifting. Thus, synchronization is often not a one-time task and needed to be carried out frequently during the lifetime of nodes, which introduces substantial costs in many aspects. Under this circumstance, the insufficiency of existing synchronization solutions makes high accuracy localization extremely challenging, especially for low-cost or COTS devices.

In this research, we propose Whistle, a new design of TDOA localization without time synchronization. High time resolution is achieved through the following key techniques: two-signal sensing and sample counting. The similar idea has been proposed by BeepBeep [23] for accurate ranging. First, receivers record both the source signal and the successive signal (called two-signal sensing). After the recording stage, each receiver counts the number of samples between these two sounds and derives the elapsed time between the two sounds (called sample counting). Then each receiver reports its results to an Access Point (AP), which uses the results to calculate the time difference of the target signal's arrival at different receivers. Finally, Whistle can locate the sound source according to the TDOA measurements.

Compared with other TDOA systems using specialized hardware, Whistle achieves low cost, rapid deployment, and widespread use simultaneously. Concretely, the main contributions of this study are as follows.

- Synchronization-free. Whistle changes the scheme of TDOA fundamentally, releasing the synchronization requirement and avoiding many sources of inaccuracy founded in other TDOA approaches.
- High accuracy. Time resolution relies on the sampling rate of receivers. Using 44.1 kHz sampling rate, Whistle has a time measuring accuracy of 0.023 ms. Such a sound recording rate is supported by most of microphones of COTS devices, like cell phones, MP3 players, sensor motes, etc.
- We implement Whistle on six COTS smart phones. Through extensive real-world experiments in indoor and outdoor, quiet and noisy environments, the mean location error is 10~20 centimeters in a $9 \times 9 \times 4m^3$ 3D space, which satisfies the requirement of a large number of applications.

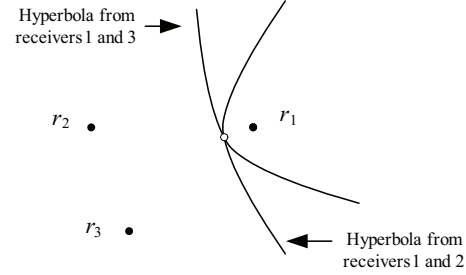


Figure 1: Location computation by TDOA

The rest of the paper is organized as follows. Section 2 presents an overview of basic TDOA schemes and the Whistle. The Whistle design is discussed in Section 3. We build up a testbed to evaluate the performance of Whistle in Section 4. In Section 5, we review the related work. Finally, we conclude this study and discuss future work in Section 6.

2. Preliminaries

2.1. TDOA Localization

TDOA measurement gives the difference of the time a signal arriving at different reference nodes. A TDOA measurement Δt_{ij} and the coordinates of reference nodes i and j define one branch of a hyperbola whose foci are at the locations of reference nodes i and j . Hence, the unknown node must lie on the hyperbola. Thus, localization based on TDOA measurement is also called hyperbolic positioning. In two-dimensional space, measurements from a minimum of three reference nodes are required to uniquely determine the location of an unknown node, as illustrated in Fig. 1.

Due to measurement noises and redundant receivers, we seldom solve the non-linear TDOA equations directly. Instead, researchers have developed several estimators [7], [8], [10], [13], [28], [31]. Comparing these methods, in this study, we adopt Chan's method [33] because it produces high accuracy with proper computational complexity.

2.2. System Overview

In the design of Whistle, the object to be located is required to be able to emit some kind of signals, which demonstrate the existence of the object. Sound signal is suitable for a large range of devices, such as cell phones, PDAs, sensor motes, or all kinds of events that make sounds. In this study, we choose a sound source as the to-be-located object. In fact, the techniques developed here can be applied for other scenarios.

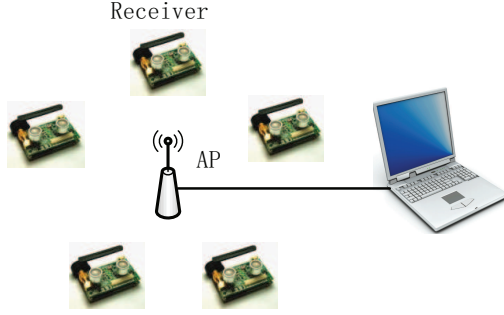


Figure 2: A star-like topology of receivers

Definition 1: S is an acoustic source locating at a fixed position in a limited 3D space and can generate an omni-directional sound signal that lasts for a specified period. For convenience, we also use S to denote the emitted signal.

In Whistle, We assume that the signal S can be mathematically described in advance. Technically, Whistle does not rely on any specific form of acoustic signals and can be accordingly used for general purposes.

In the view of hardware, Whistle involves an acoustic source and several receivers. Receivers are deployed at known locations, serving as the basic infrastructure of a TDOA system. Each receiver has a basic set of hardware, including a speaker, a microphone, and wireless connectors, such as Bluetooth or WiFi. Unless explicitly pointing out, the words node and receiver are exchangeable henceforth. Fig. 2 shows a star-like network topology of Whistle. Receivers, in recording mode, capture sound signals and transmit the time-related information to an access point. The AP relays that information to a laptop for further processing.

Using a laptop in Whistle is optional. If we use a node to replace the AP and the laptop, the network model described above is changed into a purely star-like model.

3. The Design of Whistle

3.1. Measuring TDOA by TD2S

As TDOA is important, we describe how to accurately measure it in Whistle. Fig. 4 shows a typical time sequence of two Whistle nodes A and B . Let t_{A1} and t_{B1} denote the arriving time of S at the microphones of A and B , respectively. The TDOA measurement is exactly $t_{B1} - t_{A1}$, if nodes are synchronous. But in Whistle, nodes maintain clocks independently. Even using a perfect synchronization scheme, however, we still cannot obtain the exact value of TDOA. Due to

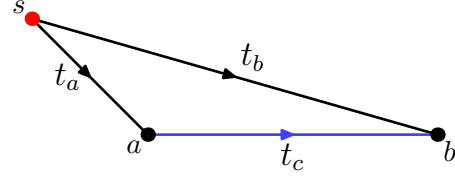


Figure 3: An illustration of transmissions of two signals

the latency of software and hardware, node A detects S at t_{A2} instead of t_{A1} . The same latency occurs at node B . Many experimental studies demonstrate that such latencies are inevitable and unpredictable [23]. Hence, $t_{A2} - t_{A1}$ and $t_{B2} - t_{B1}$ are not necessarily the same (basically different from each other), which prevents using $t_{B2} - t_{A2}$ as the estimate of $t_{B1} - t_{A1}$.

To eliminate the uncertainties and errors mentioned above, we introduce a system sound S' to avoid time synchronization.

Definition 2: After receiving the source sound signal S , one of the receivers will emit another sound signal S' , which is called the system sound signal. The node emitting S' is called base node.

With the help of the system sound S' , we develop a method to measure $t_{B1} - t_{A1}$. As shown in Fig. 4, after a specified time interval τ , node A , as the base node, emits S' from its speaker at time t_{A3} according to its own clock.

Theorem 1: For all non-base nodes, they always detect the system sound signal S' after the source sound signal S .

Proof: Let S be the sound source, a the base node, and b a non-base node. Assume that t_a and t_b are the propagation time of S from s to a and from s to b , respectively. In addition, the propagation time of S' from a to b is denoted by t_c . Finally, we use t_{ab} as the time span lasting from emitting S at s and receiving S' at b .

The locations of s , a and b construct a triangle in Fig. 3, or lie in a straight line. In both cases, we have $t_b \leq t_a + t_c$ due to triangle inequality. Since the base node encounters a delay between receiving S and emitting S' , we have $t_{ab} > t_a + t_c$. Therefore, $t_{ab} > t_b$ can be obtained, meaning that any non-base node b always firstly senses S , then S' . \square

As shown in Fig. 4, S' is emitted from A 's speaker at t_{A3} , and arrives at the microphones of A and B at t_{A4} and t_{B3} , respectively. A and B 's CPUs detect S' at t_{A5} and t_{B4} , respectively. According to Theorem 1, t_{B3} is always later than t_{B1} . We define the TD2S value of one node as the elapsed time between the arrival of S and the arrival of S' at the node. Obviously,

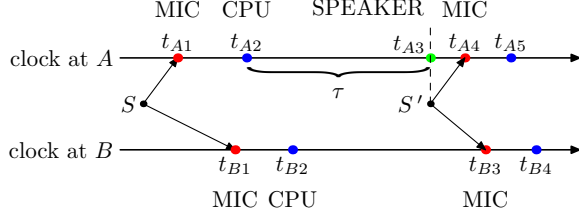


Figure 4: Time lines of node A and B

$T_{A2S} = t_{A4} - t_{A1}$, $T_{B2S} = t_{B3} - t_{B1}$. The distance of A 's speaker to B 's microphone (denoted by d_{AB}) is a constant since both A and B fix at known locations. In addition, the distance between A 's speaker to its own microphone (denoted by d_{AA}) is also a constant. Let T_{AB} denote the $TD2S$ value of A and B , i.e., $T_{AB} = t_{B1} - t_{A1}$. T_{AB} can be expressed using T_{A2S} and T_{B2S} , as illustrated in Eq. (1).

$$\begin{aligned}
 T_{AB} &= t_{B1} - t_{A1} \\
 &= (t_{B3} - t_{A3}) - (t_{B3} - t_{B1}) + (t_{A3} - t_{A1}) \\
 &= (t_{B3} - t_{A3}) - (t_{B3} - t_{B1}) \\
 &\quad + (t_{A4} - t_{A1}) - (t_{A4} - t_{A3}) \\
 &= \frac{d_{AB}}{v} - (t_{B3} - t_{B1}) + (t_{A4} - t_{A1}) - \frac{d_{AA}}{v} \\
 &= k_1 - T_{B2S} + T_{A2S} - k_2
 \end{aligned} \tag{1}$$

In Eq. (1), k_1 and k_2 are defined as $\frac{d_{AB}}{v}$ and $\frac{d_{AA}}{v}$, respectively. Since both k_1 and k_2 are constant, only T_{A2S} and T_{B2S} are needed to calculate T_{AB} . The significance of Eq. (1) is that $TD2S$ can be calculated by $TD2S$ values which can be measured independently by asynchronous nodes. The next challenge is how to calculate the $TD2S$ accurately.

3.2. Calculating $TD2S$ Accurately

In Fig. 4, since latencies are unpredictable, $t_{A4} - t_{A1}$ is not necessarily equal to $t_{A5} - t_{A2}$. Therefore, for a node, using two CPU timestamps to calculate a $TD2S$ (traditional synchronization methods often do in this way) is not promising.

Recall that nodes are in the recording state and samples at a fixed frequency (f Hz). In other words, every $\frac{1}{f}$ second, a node senses and translates signals into a real or complex number by its A/D converter. For example, when the signal S arrives at the microphone of a node X , X is recording the i -th sample data. After some latency, X 's microphone detects the signal S' when its sample counting goes to j ($j > i$ according to Theorem 1). Now, we can obtain T_{X2S} ,

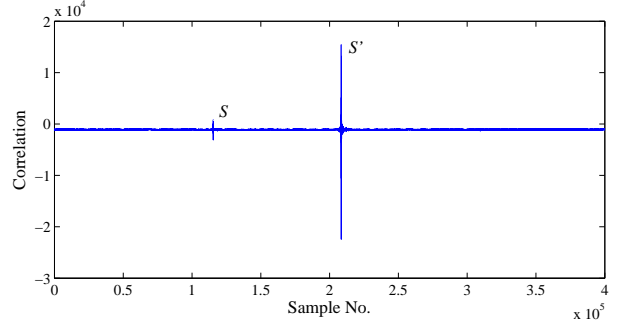


Figure 5: One example case of S and S'

the $TD2S$ of node X and have $T_{X2S} = \frac{j-i}{f}$. Thus, we eliminate the uncertainties that are inherent in the time synchronization and CPU-timestamp based methods. Basically, a higher sampling rate results in a higher accuracy. In this paper, the sampling rate is 44.1kHz, which is supported by most COTS devices.

3.3. Detecting Sound Signals

Another critical challenge in Whistle's design is: how to detect both arrivals of the sound signals S and S' ?

We assume that both S and S' use the same form of sound signal that is known as a priori. If the signal has a good auto-correlation property, it is easy and accurate to determine the peak relevant to the arrival of S by correlating it with the known signal. Specifically, after cross-correlation, the two maximal peaks represent the time-of-arrival of S and S' in theory, and the latter one is S' according to Theorem 1 (as shown in Fig. 5).

In practice, the highest peak can not always represent the arrival of signal because of multi-path effects and other uncertainties; we choose the earliest sharp peak in a shadow window of ω_0 points right before the maximal peak instead. We use two parameters: height and average slope. Average slope is calculated as

$$P = \frac{Y_{peak} - Y_{valley}}{X_{peak} - X_{valley}} \tag{2}$$

X_{valley} is the nearest valley point before the peak. Only the peak whose height is larger than $Y_{maxpeak} \times TH_Y$ and slope is larger than $P_{maxpeak} \times TH_P$ would be chosen. In our implementation, we empirically set $\omega_0 = 5000$, $TH_Y = 0.5$, $TH_P = 0.5$.

Generally, we can use any signal with good autocorrelation property as the reference signal. Considering the properties of the built-in speaker and microphone of a cell phone, in this paper, we select the linear chirp signal (sounds like a whistle) with the frequency changing from 2kHz to 6kHz as S and S' , the duration of the chirp signal is set to 50ms.

3.4. Filtering $TD2S$ Outliers

We have shown a method using $TD2S$ to calculate $TDOA$. But sometimes, because of the instability of sound recording, we get wrong $TD2S$ from defective recording which can be attributed to obstacles between S and receivers or background noises. Since when and where S being emitted are unknown, it is difficult to give a range which the real value of $TD2S$ should locate in. But still we can use a method to tell wrong $TD2S$ s from the other: majority decision.

Suppose that the sound source M emits S at t , while the base node N emits S' at t' . Receivers A and B both record the two signals. Thus, the $TD2S$ s of both receivers have the following relation:

$$\begin{aligned}
 |\Delta_{TD}| &= |T_{A2S} - T_{B2S}| \\
 &= |((t' + \frac{d_{AN}}{v}) - (t + \frac{d_{AM}}{v})) - ((t' + \frac{d_{BN}}{v}) - (t + \frac{d_{BM}}{v}))| \\
 &= \frac{|(d_{AN} - d_{BN}) - (d_{AM} - d_{BM})|}{v} \quad (3) \\
 &\leq \frac{|d_{AN} - d_{BN}| + |d_{AM} - d_{BM}|}{v} \\
 &\leq \frac{2d_{AB}}{v}
 \end{aligned}$$

Providing D is the maximum distance among all receivers in localization (because we have positions of N_i , D can be easily got), for an arbitrary pair of receivers, we have $|\Delta_{TD}| \leq \frac{2D}{v}$. So when we get correct $TD2S$ s of all receivers to form a set D and sort it, we have $TD2S_{max} - TD2S_{min} \leq \frac{2D}{v}$.

If most of the $TD2S$ s are correct and a few have significant error, elements in D can be divided into three categories: (1) a majority of elements which are in an interval whose length is shorter than $\frac{2D}{v}$; (2) a few elements which are outside of this interval; and (3) the other elements. For arbitrary elements a in (1) and b in (2), $|a - b| > \frac{2D}{v}$.

In this case we can judge category (2) as error $TD2S$ s, and only use category (1) to localize S . Obviously, category (1) must have sufficient nodes to calculate, or we have to use the whole set to get a result which usually has large errors. This kind of division bases on the assumption that most of the $TD2S$ s are correct. Such a method works well in practice.

Obviously, this method needs redundant nodes. More nodes bring extra localization or measuring cost, but enlarge the set of category (1) and enhance the accuracy of localization. In fact, there is a trade-off between cost and performance. We empirically find

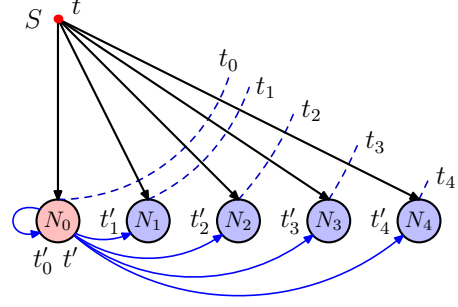


Figure 6: An illustration of the emissions and receptions of both source sound signal S and system sound signal S' . N_0 is the base node.

that 2 ~ 3 redundant nodes can increase location accuracy significantly.

3.5. Framework Design

After discussing the details of Whistle, we present the Whistle framework design in this section. As mentioned in Section 2.1, we need M ($M \geq 5$) nodes for 3D localization. Without loss of generality, let N_0 be the base node and N_i ($1 \leq i \leq M-1$) be a non-base node (N_0 is not always the nearest to S). Whistle has three steps:

In step one, nodes start recording, and receive S by their own microphones. As illustrated in Fig. 6, the signal S is emitted at time t , which will arrive at N_0, N_1, \dots , and N_{M-1} at time t_0, t_1, \dots and t_{M-1} , respectively. Note that the time point t or t_i ($0 \leq i \leq M-1$) in Fig. 6 is regarding to the absolute time line, instead of the time line maintained by any individual node N_i ($0 \leq i \leq M-1$).

In step two, the base node N_0 will emit a system sound signal S' at time t' (as shown in Fig. 6) after it detects S . The signal S' , of course, will arrive at each node after some latency. A node N_i ($0 \leq i \leq M-1$) records the time t'_i of S' arriving at N_i . Every node maintains a recording R_i which includes both S and S' . Correlating R_i with the reference signal, using the method mentioned in Section 3.3, we have $t'_i - t_i$. For node N_i , the $TD2S$ value ($t'_i - t_i$) depends only on N_i 's own clock, provided that the clock of N_i is accurate in a term longer than the period of the sound occurrence.

In the last step, the value of $TD2S$ will be delivered to a node or an AP for deriving accurate $TDOA$ values, based on which the location of the sound source can be finally determined.

4. Experimental Evaluation

In this section, we evaluate the performance by a set of experiments on a real testbed.

4.1. Prototype Configuration

To build a prototype, we use eight cell phones consisting of six Dopod P800, one O2 XDA Atom and one SonyEricsson XPeria. Each of them has a pair of built-in speaker and microphone, and a WiFi module. We use the O2 cell phone as the base node, the Sony cell phone as the sound source, and all other Dopod cell phones as receivers. The signals S and S' are designated to the linear chirp sound.

We organize these eight cell phones and an AP (TP-LINK) into a star-like network where all cell phones connect directly to an AP (seen Fig. 2). The AP relays received data to a laptop(Lenovo T60) for location computation. A high performance cell phone can replace the laptop and AP and serve as a hub connecting the other cell phones.

Using the Windows Mobile 6.0 SDK, we develop the software level of Whistle. For convenience, we implement a recorder in software level that can enter and quit the recording status periodically and save automatically the recorded sound data into a file with the *wav* format.

4.2. Source of Errors

We summarize possible sources of errors in this section. In Whistle, we get $TDOA$ from $TD2S$ first, and solve the equations according to $TDOA$. Errors may come from both steps. The main factors are as below:

- Signal to noise ratio (SNR) - Environmental noises will always be recorded. If the energy of the transmit signals is limited or the frequency of noises is close to the signals, it is difficult to detect the arrival time of the signals.
- Multi-path effects - Because of reflection, an acoustic signal may reach a receiver via different paths. Though we take the earliest peak instead of the highest peak to deal with multipath effects, sometimes the right time point cannot be caught.
- Equation solving - Chan's method performs well when the TDOA measurement errors are small. However, as the errors increase, the performance degrades. In the experiments, we find that although a majority of TDOA errors are below 0.4ms, a small portion of relatively large errors contribute a lot to the location error.

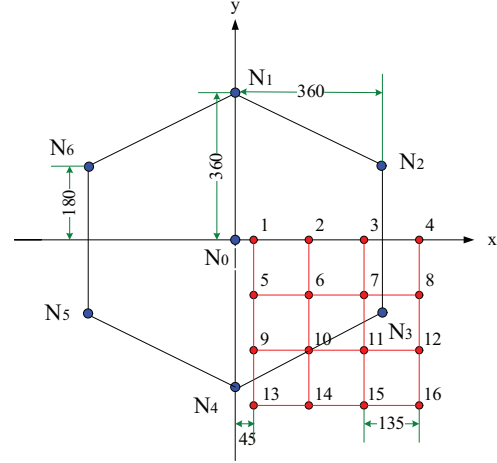


Figure 7: Deployment of receivers and acoustic sources in 2D case. Unit:cm

Except for some negligible factors, like the propagation speed variation of sound, there are still other factors that may influence localization accuracy. We plan to use the schemes like [35] to infer hidden failure reasons and detect possible faulty components in the future.

4.3. Settings of Test Environment

Whistle was evaluated in both 2D and 3D scenarios. In each scenario, according to the error sources mentioned above, we intentionally conduct the experiments in three environments:

- Case 1-Outdoor, quiet: Whistle is deployed outside a large gymnasium.
- Case 2-Outdoor, noisy: The location is same to Case 1, but loudspeaker plays music nearby.
- Case 3-Indoor, quiet: A hall of size about 9m×9m×4m.

In the following parts, we use “Normal” to denote Case 1, “Noisy” to Case 2, and “Inside” to Case 3. Through the comparison of three environments, we are expected to reveal the effect of a single factor (noise or multi-path effect).

We place the sound source at 16 different points in each environment, and collect 10 repeated samples of measuring data at each point. So totally we have 960 different samples under the 2D and 3D deployments. Since temperature may change significantly from start to end, we record temperature once S changes its position. The model of sound speed in use is formally described as $v_{air} = 331.3 + 0.6 \times \theta$ m/s in [1]. The parameter θ represents the air temperature in Celsius.

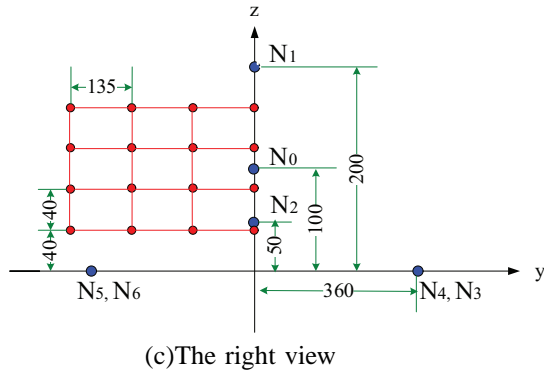
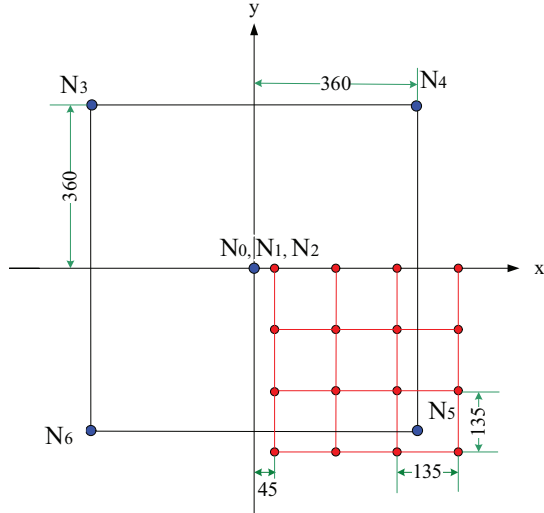
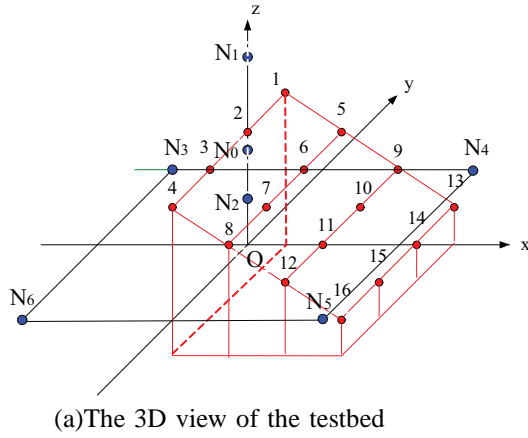


Figure 8: Nodes deployment and sound source distribution in 3D experiments. The red points denote different locations of S . Unit:cm

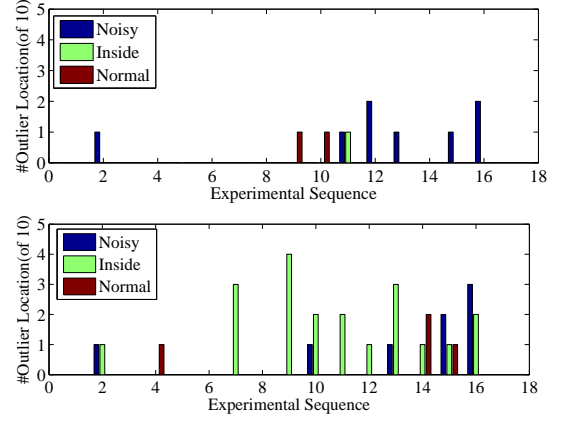


Figure 9: Outlier number. Figures at top and bottom represent the outlier number of 10 samples in 2D and 3D cases, respectively.

For 2D or 3D, in all environments, we have the same deployment of N_i and S . Since sound attenuates rapidly when transmitting in the air, our method is distance-restricted, so we do our work in an about $9m \times 9m \times 4m$ space. We depict 2D deployment in Fig. 7 and 3D in Fig. 8, the red points represent the different acoustic source locations and have been labeled with different number, the blue points represent different receivers. Critical distances are all labeled in the deployment picture. Our sound source positions only occupy 1/4 of the space because it is symmetrical.

For simplicity, we use the number associated with the sound source point as experimental sequence to denote the corresponding test in the following section.

4.4. Outlier Number

In the experiment, we find some results with extremely large errors; those errors are from obviously failed experiments which can not get enough eligible $TDOAs$. These results significantly lower positional accuracy but are of little value and can be easily separated from other results. So we identify the results with the absolute errors greater than 1m as outlier and exclude them from the results to be used in accuracy evaluation.

For example, in a case where a location S_1 , obtained by Whistle, is (10, 20, 30) and the exact sound source $S = (8, 22, 30)$, $|x_{S_1} - x_S| = 2m > 1m$ shows that the result of S' is an outlier.

Fig. 9 shows the number of outliers of the 10 samples in every point of experiments. It is obvious that there are only very few outliers in most experiments. The outlier number of inside scene in 3D case is the largest, but even in this scene, the percentage of

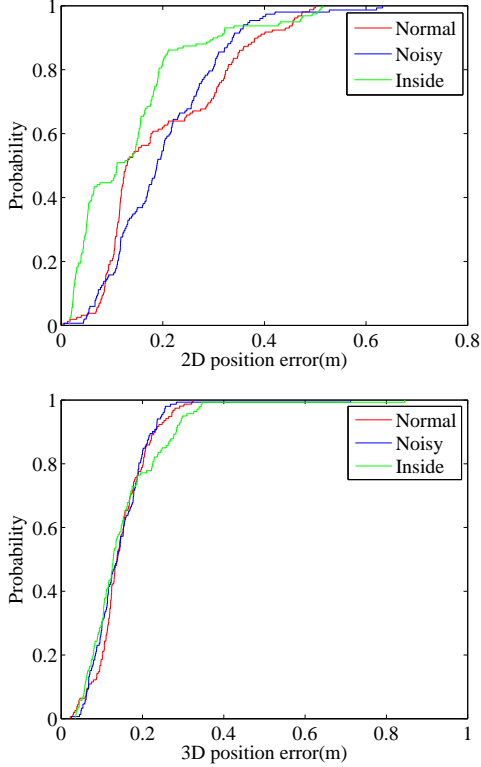


Figure 10: CDF of position errors in 2D and 3D cases

outlier is only 12.5%. The largest outlier number in the 3D inside scene suggests that WSN localization in indoor environment is vulnerable due to multi-path and obstacles, so researchers often resort to other equipments in this case. For example, Lionel M. NI et al. [18] propose a location sensing prototype system named LANDMARC that uses RFID for locating objects inside buildings. The similar idea can be used in our future work to enhance our system's resistance in indoor environments.

In the figure we can see that 2D experiments have much less outliers than 3D. We can explain it with the conclusion which we get from Section 3.4 that more redundant nodes bring higher accuracy. The 2D case has one more redundant node than 3D, so it is easier for 2D to avoid outliers. It is observed that both noise and multi-path cause more outliers. In 3D case multi-path gives stronger interference than 2D.

4.5. Positional Errors

Fig. 10 depicts the cumulative density function (CDF) of the positional errors in 2D and 3D case. The mean position error and some thresholds can be seen in Tab.1 and Tab.2. The pictures and tables reveal a

mean error of 10 ~ 20 cm in 3D case and 10 ~ 21 cm in 2D case. They demonstrate that whistle achieves high accuracy in all environments.

We can see that results in normal environments are not always the best. Though noise and multi-path increase the number of outliers, they do not directly enlarge errors of right results. The consequences of 3 environments are quite similar. We think that our system resists multi-path because of the efficient method for finding the first peak (seen in Section 3.3), and resists noises because of the good autocorrelation property of chirp signal.

Table 1: 2D critical data

Indicator(cm)	Normal	Noisy	Inside
<i>mean error</i>	20.0	20.3	13.7
<i>50% error</i>	12.7	18.8	10.9
<i>90% error</i>	38.1	33.7	29.9
<i>std</i>	12.8	10.9	12.1
<i>outlier percentage</i>	1.25%	5%	0.6%

Table 2: 3D critical data

Indicator(cm)	Normal	Noisy	Inside
<i>mean error</i>	14.7	14.4	15.2
<i>50% error</i>	13.7	13.8	12.8
<i>90% error</i>	22.8	21.9	28.2
<i>std</i>	6.2	7.4	9.9
<i>outlier percentage</i>	2.5%	5%	12.5%

5. Related Work

Many range-based localization algorithms have been proposed within a few past decades. Existing work basically use three types of measurements of acoustic signals: TDOA [24], [19], [32], [9], DOA (direction of arrival) [22], [14], [12], [15] and received signal energy [25], [26], [4], [21], [3].

Cricket [24] is a TDOA based localization system. It uses concurrent wireless communication and ultrasonic signal to determine the *TDOA* value between two nodes referring to their own CPU timestamps. Mahajan et al.'s work [19] uses TDOA measurements to solve a linear equations system similar to ours. Since their TDOAs are measured by CPU timestamps, clock synchronization is needed. Moreover, their work is evaluated only by simulation. The work in [32] employs 8 microphones to obtain the TDOA information, which is further used to estimate the bearings of sound sources.

Motivated by the applications of monitoring animals in field, the work in [2], [12] implements and evaluates a sound source localization system called ENSBox [12] and employs it to localize marmot alarm-calls. ENSBox is a DOA-based platform, integrating

an ARM processor, a sensor array, a wireless radio connector, and precise self-calibration of array position and orientation.

An energy-based localization system is proposed in [4], in which the positions of speakers are estimated using an ad hoc microphone array setting and the maximum likelihood estimation method. In [21], a set of methods based on energy measurement are developed, compared, and contrasted. Particularly, a weighted direct least-squares formulation is presented, which incorporates the dependence of unknown parameters leading to high performance.

BeepBeep [23] is an acoustic-based ranging system with high accuracy. It measures the distance of two cell phones using only their microphones, speakers, and WiFi connection, without leveraging any pre-planned infrastructure. In BeepBeep ranging system, two cell phones emit the same sound signal (Beep) successively, then each phone computes the elapsed time between the two time-of-arrivals (ETOA). With the two ETOAs, the distance between the two phones can be obtained. Both the two nodes of BeepBeep must actively participate the ranging process, so the scheme of BeepBeep can be used for TOA localization, but not for TDOA inherently because the to-be-located object (or event) does not necessarily cooperate with receivers. Whistle extends BeepBeep and develops new techniques for TDOA localization.

Though range-based localization methods can get high accuracy, they often need expensive equipments and have strict connectivity requirements. Some classic range-free algorithms like DV-hop [5] and APIT [30] are proposed to implement localization in more extensive circumstances. DV-hop uses the average distance of each hop and hop counts to calculate the real distance between two nodes, so DV-hop has better performance in isotropic WSNs. APIT makes tests to know whether the unknown node is inside the triangle formed by different anchor nodes, and obtains the intersected region of all the triangles that cover the unknown node, then sets the centroid of that region as the result. APIT has high accuracy but has strict requirements for connectivity. Mo Li et al. [17] reveal that these range-free schemes fail in anisotropic WSNs with possible holes, and propose the Rendered Path (REP) protocol that is the only protocol for locating sensors with constant number of seeds in anisotropic WSNs.

Several techniques have been proposed for solving non-linear TDOA equations. Fang [7] reduced the computation to the solution of a quadratic or a quartic equation, but his method can not make use of extra measurements from extra receivers to improve

position accuracy. More general methods with extra receivers can be found in [10], [28]. They provide closed-form solutions, but their estimators are not unbiased and optimum. Abel [13] proposes a divide and conquer (DAC) method which can achieve optimum performance and unbiased estimator when the data vector is appropriately subdivided, but DAC requires quiet large Fisher information and is difficult to be implemented. The Taylor-series method [31], [8] can get high accuracy at reasonable noise levels, but it is computationally intensive because of its iterative course and finding a proper initial point to avoid the convergence problems is not easy. The method proposed by Chan [33] is noniterative and gives an explicit solution, and attains the Cramer-Rao lower bound near the small error region. This method also has a higher noise threshold than DAC. In short, the method balances computational complexity and accuracy, and is accordingly adopted in our work.

6. Conclusion

In this paper, we propose an acoustic source localization framework, Whistle. As a TDOA based system, Whistle changes the scheme of TDOA fundamentally, by releasing the synchronization. In Whistle, several asynchronous receivers record a target signal and a system signal. High time resolution is achieved through two-signal sensing and sample counting. Real-world experiments are conducted on a testbed system consisting of COTS cell phones. Experiment results show that Whistle has a mean location error of 10~20 centimeters in a $9 \times 9 \times 4m^3$ 3D space. To summarize, Whistle achieves low cost, rapid deployment, and widespread use simultaneously.

Our ongoing work are: (1) extending Whistle for larger wireless sensor networks, in which a large number of nodes are organized in an ad-hoc manner. There may be quite a lot new problems including error control and localizability [36] in this multi-hop WSN. Besides, we may have to construct a self-adaptive WSN topology like [16], for some nodes may run out of energy earlier than the other. Finally, we should pay particular attention to the factors that restrict the WSN's scale and lifetime proposed by Yuan He et al. [34]; (2) trying to use ordinary sound or voice instead of a signal that can be mathematically described, and see if we can find some new method for detecting its arrival accurately. (3) using RF signal instead of acoustic signal. UWB signal is our first choice, as it offers high time resolution. We will implement Whistle on a UWB localization system and see its effects.

Acknowledgement

Special thanks to Bo Liu for help in designing the sound recording module and doing the early experiments. This work is supported in part by the NSF China under Grant No.60803124 and the National Basic Research Program of China(973 Program) under grant No.2006CB303000.

References

- [1] http://en.wikipedia.org/wiki/speed_of_sound.
- [2] A. M. Ali, T. C. Collier, and L. Girod. An empirical study of collaborative acoustic source localization. In *IPSN*, pages 41–51, 2007.
- [3] D. Ampeliotis and K. Berberidis. Linear least squares based acoustic source localization utilizing energy measurement. In *SAMSP*, 2008.
- [4] M. Chen, Z. Liu, L. He, P. Chou, and Z. Zhang. Energy-based position estimation of microphones and speakers for ad hoc microphone arrays. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2007.
- [5] D. Niculescu and B. Nath. Ad hoc positioning system (aps). In *IEEE GlobeCom*, San Antonio, AZ, November 2001.
- [6] J. Elson, L. Girod, and D. Estrin. Fine-grained network time synchronization using reference broadcasts. In *OSDI*, 2002.
- [7] B. T. Fang. Simple solutions for hyperbolic and related position fixes. *IEEE Trans. on Aerospace and Electronic Systems*, 26(5):748–753, September 1990.
- [8] W. H. Foy. Position-location solutions by taylor-series estimation. *IEEE Transactions on Aerospace and Electronic Systems*, AES-12(2):187–194, March 1976.
- [9] K. Frampton. Acoustic self-localization in a distributed sensor network. *IEEE Sensor Journal*, 6(1):166–172, 2006.
- [10] B. Friedlander. A passive localization algorithm and its accuracy analysis. *IEEE Journal of Oceanic Engineering*, OE-12(1):234–245, January 1987.
- [11] S. Ganeriwal, R. Kumar, and M. B. Srivastava. Timing-sync protocol for sensor networks. In *SenSys*, pages 138–149, 2003.
- [12] L. Girod, M. Lukac, V. Trifa, and D. Estrin. The design and implementation of a self-calibrating distributed acoustic sensing platform. In *SenSys*, pages 71–85, 2006.
- [13] J.S. Abel. A divide and conquer approach to least-squares estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 26(2):423–427, March 1990.
- [14] K. Kaplan, Q. Le, and P. Molnar. maximum likelihood methods for bearings-only target localization. In *ICASSP*, 2001.
- [15] A. Ledeczi, G. Kiss, B. Feher, P. Volgyesi, and G. Balogh. Acoustic source localization in fusing sparse direction of arrival estimates. In *ISES*, 2006.
- [16] Mo Li and Yunhao Liu. Underground coal mine monitoring with wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 5, March 2009.
- [17] Mo Li and Yunhao Liu. Rendered path: Range-free localization in anisotropic sensor networks with holes. *IEEE/ACM Transactions on Networking (TON)*, 18(1):320–332, February 2010.
- [18] Lionel M Ni, Yunhao Liu, Yiu Cho Lau and Abhishek Patil. Landmarc: Indoor location sensing using active rfid. *ACM Wireless Networks, (WINET)*, 10:701–710, November 2004.
- [19] A. Mahajan and M. Walworth. 3-d position sensing using the differences in the time-of-flights from a wave source to various receivers. *IEEE Trans. on Robotics and Automation*, 17(1):91–95, 2001.
- [20] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi. The flooding time synchronization protocol. In *SenSys*, pages 39–49, 2004.
- [21] C. Meesookho, U. Mitra, and S. Narayanan. On energy-based acoustic source localization for sensor network. *IEEE Trans. on Signal Processing*, 56(1):365–377, 2008.
- [22] Y. Oshman and P. Davidson. Optimization of observer trajectories for bearings-only target localization. *IEEE Trans. on Aerospace and Electronic Systems*, 35(3):892–901, 1999.
- [23] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan. Beepbeep: a high accuracy acoustic ranging system using cots mobile devices. In *SenSys*, pages 1–12, Sydney, Australia, November 2007.
- [24] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Mobicom*, pages 32–43, 2000.
- [25] X. Sheng and Y. Hu. Energy based acoustic source localization. In *IPSN*, 2003.
- [26] X. Sheng and Y. Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Trans. on Signal and Processing*, 53(1):44–54, 2005.
- [27] F. Sivrikaya and B. Yener. Time synchronization in sensor networks: a survey. *IEEE Network*, pages 45–50, 2004.
- [28] J. O. Smith and J. S. Abel. The spherical interpolation method of source localization. *IEEE Journal of Oceanic Engineering*, OE-12(1):246–252, January 1987.
- [29] P. Sommer and R. Wattenhofer. Gradient clock synchronization in wireless sensor networks. In *IPSN 09: Proceedings of the 2009 International Conference on Information Processing in Sensor Networks*, pages 37–48, Washington, DC, USA, 2009. IEEE Computer Society.
- [30] T. He, C. Huang, B.M. Blum, J.A. Stankovic and T.F. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *MOBICOM*, pages 81–95, 2003.
- [31] D. J. Torrieri. Statistical theory of passive location systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-20(2):183–198, March 1984.
- [32] J.-M. Valin, F. Michaud, J. Rouat, and D. Letoumeau. Robust sound source localization using a microphone array in a mobile robot. In *IROS*, 2003.
- [33] Y.T. Chan and K.C. Ho. A simple and efficient estimator for hyperbolic location. *IEEE Transactions on Signal Processing*, 42(8):1905–1915, August 1994.
- [34] L. M. Yuan He and Y. Liu. Why are long-term large-scale wireless sensor networks difficult: early experience with greenorbs. *Mobile Computing and Communications Review*, 14(2):10–12, 2010.
- [35] K. L. Yunhao Liu and M. Li. Passive diagnosis for wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 18(4):1132–1144, August 2010.
- [36] X. W. Yunhao Liu, Zheng Yang and L. Jian. Location, localization, and localizability. *Journal of Computer Science and Technology*, 25(2):274–297, March 2010.