# Demo Abstract: Whistle - Synchronization-Free TDOA for Localization


Ran Yu Bin Xu Guodong Sun
Department of Computer Science and Technology
Tsinghua University, Beijing, China
Tsinghua National Laboratory for Information Science and Technology, Beijing, China
{yuran,xubin,sgdcs}@keg.cs.tsinghua.edu.cn

Zheng Yang
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
yangzh@cse.ust.hk


## 1 Introduction

We demonstrate a synchronization-free localization system, called Whistle, for wireless sensor networks. The system is tested in several realistic environments on commercial off-the-shelf(COTS) devices and the mean error is 10∼20 centimeters in a $9 \times 9 \times 4m^3$ 3D space.

TDOA is a widely-used localization method. 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 calculated. During the process, receivers have to be synchronized. However, existing synchronization approaches often produce errors which cannot be efficiently eliminated and accordingly lower localization accuracy of TDOA systems. In addition, synchronization is resource-consuming, since it often needs to be carried out regularly due to clock skew and drifting.

Priyantha [2] got rid of synchronization in the Cricket system which uses a combination of ultrasound and RF hardware. But the method in Cricket cannot be widely implemented, as it requires two kinds of signals emitted simultaneously and its hardware is expensive. In our research, we propose a new design of TDOA localization (called Whistle) using COTS devices without time synchronization. In our scheme, several asynchronous receivers record a target signal and a successive signal that is generated artificially. And we calculate TDOA of two different nodes by the elapsed time between the two sounds.

Whistle changes the scheme of TDOA fundamentally, totally releasing the synchronization requirement and avoiding many sources of inaccuracy found in other TDOA schemes. The 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 COTS devices, such as cell phones, PDAs, MP3 players, sensor motes, etc. We implement Whistle on COTS cell phones. Compared with other TDOA systems using specialized hardware, Whistle achieves low cost, rapid deployment, high accuracy, and widespread use simultaneously.



*SenSys'10*, November 3–5, 2010, Zurich, Switzerland.
ACM 978-1-4503-0344-6


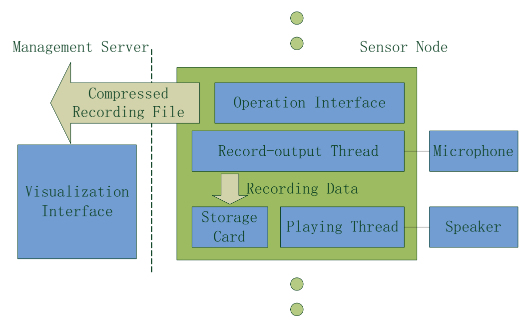


**Figure 1. System overview.**

Our contributions are: (1)designing a novel scheme of TDOA localization in which time synchronization is not needed; (2)developing several techniques to increase time resolution and avoid a number of time uncertainties founded in other TDOA systems.

In the following, we present the three major components of Whistle: controlling mechanism of sensors, real-time data transmission, and location computation.

## 2 System Design

### 2.1 System Architecture

Fig. 1 shows the system architecture of Whistle. The system is composed of a management server on which a localization GUI is running, and sensor nodes whose microphones can record sound with 44.1kHz sample rate. The server displays sensor's information on its visualization interface, by which users can issue orders to sensors. A sensor node consists of an operation interface, a record-output thread and a playing thread. A sensor's recording data are written to its storage card as a .wav file. There is a sound source $S$ in the sensors. One specific sensor $S'$ called base node emits sound after $S$. For simplicity, we also use $S$ and $S'$ to denote the sound emitted by them. The sound source and the base node don't record, only emitting sound in due time. The other sensors which don't emit sound are called anchors. The base node and the anchors all know their positions.

**Figure 2. Localization equipments.**

## 2.2 Intercommunications

A complete localization course is realized through intercommunications between the server and sensors. Firstly, all the sensors must connect to the server. We implement it by connecting the server and sensors to a WLAN, and inputting the server IP into sensors' operation interfaces manually. After connecting successfully, sensors send their basic information, including IP, location($S$ doesn't have it), name, etc. to the server. These information would be shown in the server's clients list. Secondly, with adequate sensors, a user can start a round of localization by sending commands of starting recording to anchors. Having ensured all the anchors are in recording state by ACK from them, the server automatically orders $S$ and $S'$ to emit sound successively. After receiving ACK, which shows the completion of sound playing at $S'$, the server demands of all the nodes to finish recording and send its recording data to the server.

## 2.3 Real-time Data transmission

From correlating the recording data with sample data, with a known sample rate, we can calculate the elapsed time between the two sounds, by which the arrival-time difference of two different nodes can be got through our method. But the problem is: Should we perform this calculation on individual sensor, or on the server? In practice, we find that in the former plan, it takes about one and a half minutes for a sensor to analyze a 5-second recording file which contains about 220 thousand samples with a linear-time correlation algorithm; in the latter plan, a compressed 5-second file's size is about 25Kb and can be sent to the server through a 54Mbps WLAN immediately. The calculation on sensor is a bottleneck, so we choose the latter scheme. If the sensor is more capable and the network bandwidth is narrower, maybe we adapt the former.

## 2.4 Calculations on the Server

As mentioned above, the server analyzes the recording files to get the elapsed time between the two sounds, and calculates TDOA of two different nodes by the node positions and elapsed time. Having all the TDOAs, We choose the algorithm proposed by Y.T.Chan and K.C.Ho [3] to get the result. Finally we show the sound source's position on the server's visualization interface.

# 3 Implementation

To build a prototype, we use eight cell phones consisting of six Dopod P800, one O2 XDA Atom and one SonyEricsson XPeria. Part of them can be seen in Fig. 2. 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 in Fig. 3). The AP connects to the server, which is a laptop or a PC where the solution is got. Using the Windows Mobile 6.0 SDK, we develop the software level of Whistle, and deploy them to cell phones.

# 4 Demonstration

In the demonstration, we would show the following capabilities of our system:

1. Localization accuracy: We test our system in a $10\times10$ $m^2$ area. After emission, intercommunication, and calculation, we get the location of $S$ and compare it with the measurement value.

2. Real-time feature: After the emission of source sound, Whistle can achieve a high-accuracy result within about 10 seconds, which is short enough for continuous localizing tasks in sensor networks, such as object tracking.

# 5 Acknowledgements

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.

# 6 References

[1] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi. The flooding time synchronization protocol. In *SenSys*, pages 39–49, 2004.

[2] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Mobicom*, pages 32–43, 2000.

[3] 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.

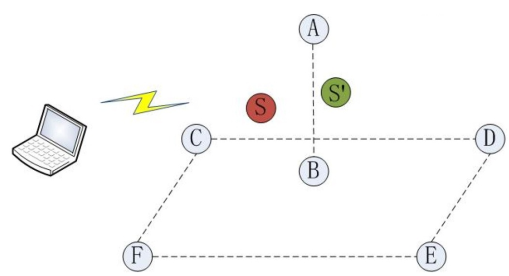


**Figure 3. The demonstration with 8 sensors. A$\sim$F are anchors. All sensors connect directly to the laptop.**