A Linear Programming Approach to Sequence-Based Localization in Acoustic Sensor Networks

Naigao Jin, Xin Zhou, Jialin Liu, Qing Tan, Zida Wen, Zihan Wang, Lei Wang
School of Software, Dalian University of Technology, China
Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province
Email: ngjin@dlut.edu.cn, lei.wang@ieee.org

***Abstract*—Acoustic source localization in sensor network is a challenging task because of severe constraints on cost, energy, and effective range of sensor devices. To overcome these limitations in existing solutions, this paper formally describes, designs, implements, and evaluates a Linear Programming method to Sequence-Based Localization, i.e., LPSBL, in distributed smartphone networks. The localization space can be divided into distinct regions, and each region can be uniquely identified by the node sequence that represents the ranking of distances from the reference nodes to that region. The key idea behind LPSBL is to turn the localization problem into linear feasibility problem by processing the node sequence. The proposed design is evaluated through theoretical analysis, extensive simulations, and physical experiments (an indoor test-bed with 30 smartphone nodes). Evaluation results demonstrate that LPSBL can effectively localize the acoustic source with good robustness.**

Keywords—sequence-based localization, linear programming, wireless sensor networks

I. Introduction

Acoustic source localization (ASL) plays an important role in a wide range of application scenarios, such as speaker-location-aware audio capturing in videoconferencing [1], shooter localization in a battle field [2], and wild biological acoustic studies [3]. The traditional centralized microphone array-based solution to ASL exploited multiple synchronized microphones to simultaneously acquire multiple signals, which have some limitations with regard to the distances between the microphones, and sensing range for the large-scale applications. Wireless acoustic sensor networks (WASNs) can overcome these limitations. A WASN consists of a set of wireless microphone nodes that are spatially distributed over the environment, usually in an ad-hoc fashion. Due to the wireless communication capabilities, the array-size limitations disappear and the microphone nodes can physically cover a much larger area.

Acoustic source localization problem in sensor networks has been widely studied in the literature. ASL in WASN is becoming feasible due to recent advances in personal portable computing devices with the rapid deployment ability. Most of the distributed acoustic source localization systems are range-based localization, which are built on top of distance or angle measurements among sensor nodes. These approaches can provide good localization performance, however, generally require costly hardware and have limited effective range due to energy constraints. The requirement of low cost and power prohibits many range-based methods for sensor node localization, especially for large-scale deployments. Yedavalli, *et al.* [4] proposed a Sequence-Based Localization (SBL) in wireless sensor networks. The heart of SBL is the division of a 2D localization space into distinct regions by the perpendicular bisectors of lines joining pairs of reference nodes (nodes with known locations). Each distinct region formed in this manner can be uniquely identified by a location sequence that represents the distance ranks of reference nodes to that region. The unknown node first determines its own location sequence based on the measurement between itself and the reference nodes, then searches through the location sequence table to determine its location.

In this paper, we present a Linear Programming method to Sequence-Based Localization (LPSBL) by processing the node sequence. As a range-free approach, this design applies node sequences instead of direct distance or TOA measurements for localization, and brings in the following two advantages: (i) node sequences feature better robustness to the measurement noise; (ii) node sequences significantly alleviate the accuracy requirement of network level time synchronization. Compared with earlier works on sequence-based localization in sensor networks (e.g. SBL [4]), the primary contribution of this article is providing an effective and optimal approach to solve the sequence-based localization problem for sensor networks. Without brute-force searching in SBL, the proposed LPSBL system formulates the sensor node localization as an convex optimization problem of finding a feasible solution to a system of multiple linear inequalities, which is produced by the node sequences. Then, linear programming (LP) can be applied to reliably and efficiently deal with the convex optimization problem, even in large-scale sensor networks. The proposed design is evaluated with both test-bed experiments and extensive simulations. Evaluation results show that the proposed LPSBL system can provide improved node localization accuracy.

The rest of the article is organized as follows. Section II presents an overview of the localization system. Then, the LPSBL is introduced in section III. Section IV discusses several practical issues. Section V presents simulation results and an empirical evaluation. Section VI briefly surveys related work. Section VII concludes the whole article.

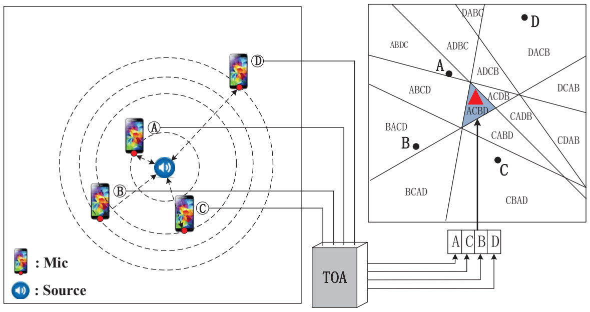

Fig. 1: Overview of LPSBL

II. System Overview

In this section, we focus mainly on the system overview of our LPSBL system, which aims at locating an unknown node. Fig. 1 shows a layout of a sensor network with anchor nodes and the acoustic source. We use circles to denote anchor nodes with known locations and the triangle to denote the acoustic source. Assume that a 2D localization space consists of N reference nodes. Consider any two reference nodes and draw a perpendicular bisector to the line joining their locations. This perpendicular bisector divides the localization space into two different regions that are distinguished by their proximity to either reference nodes, as illustrated in Fig. 1. Similarly, if perpendicular bisectors are drawn for all pairs of reference nodes, they divide the localization space into many regions. All locations inside a region have the same location sequence. If each region in the arrangement is represented by its centroid, then there is a one-to-one mapping between a location sequence and the centroid of the region that it represents.

Briefly, sequence-based localization system works as follows. After the acoustic source generates a sound, anchor nodes detect the event sequentially at different time instances that naturally gives an ordering of related nodes, called a node sequence. For instance, in Fig. 1, when the acoustic source generates a wave, the node sequence $NodeSeq(ACBD)$ is obtained along the sound propagation. The location information of acoustic source is embedded within the node sequence. The node sequence of a given region is unique to that region. By collecting all sensing results, locations of the acoustic source can be estimated by processing the node sequence.

Specifically, as shown in Fig. 2, $NodeSeq(ACBD)$ can be obtained by time of arrival (TOA) information of the acoustic event. Moreover, the TOA sequence $t_A < t_C < t_B < t_D$ is determined by the distance sequence $d_A < d_C < d_B < d_D$ from each node to the acoustic source S.

Problem 1: Assume that the location of acoustic source is known, the node sequence can be obtained by the distances from nodes to the acoustic source.

Sequence-based localization can be considered as an inverse problem of Problem 1, and described as:

Problem 2: Assume that the node sequence obtained from the measurement is known, estimate the location of an acoustic source.

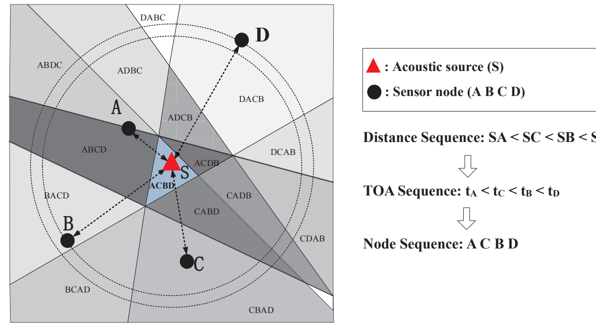

Fig. 2: The basic idea of LPSBL

In practice, solving the inverse problem is extremely difficult. In prior research, SBL estimated the location of nodes roughly by using a searching method with heavy computation. While in this paper, we carefully formulate Problem 2 as a convex optimization issue and propose an efficient solution which can provide the optimal localization results. To the best of our knowledge, this is the first work to leverage convex optimization for solving sequence-based localization problems in sensor networks.

III. Design

In this section, we firstly introduce the basic LPSBL method. After the basic LPSBL method is proposed, we describe the robust LPSBL method in the next subsection. Finally, the computational complexity analysis of LPSBL is given.

A. Basic LPSBL

In this section, we introduce the basic sequence-based localization technique based on linear programming method.

Considering a sensor network in the 2D space with N nodes, all nodes is $\boldsymbol{X} = \{\boldsymbol{node_1}, \cdots, \boldsymbol{node_i}, \cdots, \boldsymbol{node_N}\}$, where any node $\boldsymbol{node_i}$ has its location coordinates denoted as $[x_i, y_i]$. As showed in Fig. 2, an acoustic event occurs at $\boldsymbol{X_s}$= $[x_s; y_s]$, d_i is the distance from $node_i$ to the acoustic source $\boldsymbol{X_s}$. The node sequence is determined by the distances from nodes to the acoustic source $\boldsymbol{X_s}$. Given the following node sequence $NodeSeq(\cdots, i, j, \cdots)$, $d_i < d_j$ can be inferred. We have the following inequality:

$$(x_i - x_s)^2 + (y_i - y_s)^2 < (x_j - x_s)^2 + (y_j - y_s)^2 \quad (1)$$

Then, we get

$$2(x_j - x_i)x_s + 2(y_j - y_i)y_s < x_j^2 - x_i^2 + y_j^2 - y_i^2 \quad (2)$$

Given the node sequences with N node, we can get $N(N-1)/2$ linear constraints. The locations of nodes can be computed by solving the following linear feasibility problem:

$$\boldsymbol{A}\boldsymbol{X_s} < \boldsymbol{b} \quad (3)$$

where

$$
\boldsymbol{A} = \begin{bmatrix} 2x_2 - 2x_1 & 2y_2 - 2y_1 \\ 2x_3 - 2x_2 & 2y_3 - 2y_2 \\ \vdots & \vdots \\ 2x_n - 2x_{n-1} & 2y_n - 2y_{n-1} \end{bmatrix}
$$

$$
\boldsymbol{b} = \begin{bmatrix} x_2^2 - x_1^2 + y_2^2 - y_1^2 \\ x_3^2 - x_2^2 + y_3^2 - y_2^2 \\ \vdots \\ x_n^2 - x_{n-1}^2 + y_n^2 - y_{n-1}^2 \end{bmatrix}
$$

The problem of finding a feasible solution to a system of linear inequalities can be described as a linear programming problem in which the objective function is zero. We can find a solution to this feasibility program only if there is an embedding satisfying all of the constraints. We utilize the nodes sequences to set the inequality constraint. The bound constraint in the linear programming problem is set based on the size of the network deployment area. Describing the problem of sequence-based localization in the standard form of linear programming as

$$
\hat{\boldsymbol{X}}_{\boldsymbol{s}} = \min \boldsymbol{c}^{\boldsymbol{T}} \boldsymbol{X}_{\boldsymbol{s}} \tag{4}
$$

$$
s.t.\ \boldsymbol{A}\boldsymbol{X}_{\boldsymbol{s}} \leq \boldsymbol{b}
$$

where $\boldsymbol{c}$ is zero vector.

To summarize, the LPSBL is presented in Algorithm 1. The input is the node sequences and locations of anchor; the output is the position of the acoustic source. Step 1 sets the objective function of the optimization problem. Step 2 uses the node sequence to construct the inequality constraint. Step 3 solves the LP problem to achieve the position of the acoustic source.

Algorithm 1: LPSBL Method

Input: The position of N nodes
The node sequence of the acoustic source for N reference nodes

Output: Position of the acoustic source

```
Step 1: Setting objective function: setting c ← 0;
Step 2: Constructing inequality constraint:
        setting A and b by processing the node sequence;
for i ← 1 to N − 1 do
    for j ← i + 1 to N do
        According to inequality (2) to get inequality
         constraint;
    end
end
Step 3: Solve LP problem to get the target's position.
```

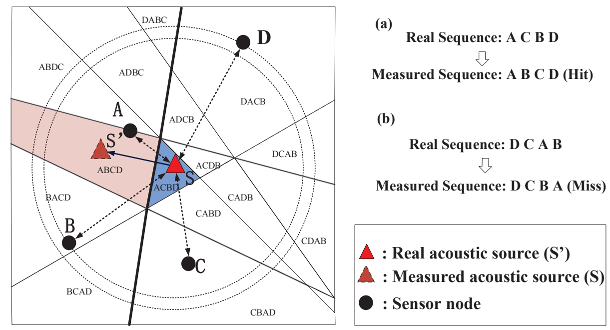

Fig. 3: Flip Tolerance

B. LPSBL with Flip Tolerance

For the sake of presentation, until now we have described LPSBL in an ideal case where a complete and perfect node sequence can be obtained. In this section, we describe how to make LPSBL work well under more realistic conditions.

In the practical application, if two nodes are located too close to each other along the direction of event propagation, they detect the event almost simultaneously. In this case, the node ordering in the sequence may occur flip. For instance, the true sequence is $NodeSeq(\cdots, i, j, \cdots)$, but the detected sequence is $NodeSeq(\cdots, j, i, \cdots)$. Algorithm 1 can find a solution to this feasibility program only if there is an embedding satisfying all of the constraints. However, it is impossible to find a feasible solution that satisfies all of the constraints when sequence flip occurs. For example, as showed in Fig. 2, the right middle region identifies by the node sequence $NodeSeq(DCAB)$. Once the order of node A and B occur flip in $NodeSeq(DCAB)$, the region corresponding to $NodeSeq(DCBA)$ does not exist, Algorithm 1 can not give the accurate estimation. In this section, we propose the solution to address the problem of sequence flip using traditional convex relaxation techniques.

We thus introduce a slack variable ξ_{ij} for each inequality constraint to allow for inequality violations. Rewrite the inequality constraint (2) as

$$
d_i - d_j \leq 0;\ i, j \in S \tag{5}
$$

Relax the inequality constraint (5) to

$$
d_i - d_j \leq \xi_{ij};\ i, j \in S \tag{6}
$$

where $\xi_{ij} > 0$.

As a result, LPSBL with error tolerance can be formulated as a convex optimization problem with linear inequalities constraints as follows:

$$
\min_{x_i, y_i} \sum_{(i,j) \in X} \xi_{ij};\ \xi_{ij} \geq 0 \tag{7}
$$

$$
s.t.\ d_i - d_j \leq \xi_{ij}
$$

where the objective function of optimization problem is the total amount of all slacks.

In the optimization problem expressed by the equation (7), the problem of inequality violations can be solved by introducing a slack variable for each inequality [5]. Therefore, the proposed LPSBL can provide the robust estimation when the problem of sequence flip occurs.

C. Incremental LPSBL

The data transmitted from the sensor node reach the the server in sequence in WSN. The basic idea of the proposed incremental LPSBL is to ultilize the available constraints to solve the linear program. Once the new data come, the proposed incremental LPSBL add the new constraints to refresh the location of the acoustic source by solving a new linear program.

Algorithm 2: Incremental LPSBL Method

Input: the first two data package $pack_i$ $pack_j$
newest data package $pack_k$
Output: Location of the acoustic source

Step 1: Initial LP problem using the first two data package $pack_i$ $pack_j$
while (the newest data package $pack_k$ reach)
$pack_k$ (ID_k,$Location_k$,TOA_k)
Step 2: Constructing inequality constraint according to the TOA of the newest data package :
Step 3: Solve LP problem to get the target's location.

D. Robust LPSBL

Given the $NodeSeq(ACBD)$, the constraint is tight for AC, CB and BD, can contribute more to the localization accuracy. However, compared with the constraint of AD, the the constraint of AC is subject to flip. We add the constraint of AD firstly, then process the constraints of AB and CD, finally, the constraints of AC, CB and BD is added. After add the new constraint, once the optimal solution is not given, we just discard the constraint and keep the optimum before.

Algorithm 3: Robust LPSBL

Input: the first two data package $pack_i$ $pack_j$
newest data package $pack_k$
Output: Location of the acoustic source

Step 1: Initial LP problem using the first two data package $pack_i$ $pack_j$
while (the newest data package $pack_k$ reach)
$pack_k$ (ID_k,$Location_k$,TOA_k)
Step 2: Constructing inequality constraint according to the TOA of the newest data package :
Step 3: Solve LP problem to get the target's location.

E. Complexity Analysis

This section provides the complexity analysis for the proposed LPSBL design. It needs to be emphasized that LPSBL itself adopts an asymmetric design in which sensor nodes need only to detect and report the events. Therefore, we only analyze the computational cost on the node sequence processing side, where resources are plentiful.

SBL: Calculating the Kendalls Tau between two sequences is $O(N^2)$ operations. Since the location sequence table is of size $O(N^4)$, searching through it takes $O(N^6)$ operations [4].

LPSBL: The complexity of low dimensional linear programming with L constraints is $O(L)$ [6]. The number of linear constraints is $N(N-1)/2$ in LPSBL, where N is the number of nodes. Thus, the overall computation complexity of LPSBL can be written as $O(N^2)$.

IV. Discussion

A. multiple measurement

Sequence flip may provide more constraints to sequence-based localization. When the position of two nodes in the node sequence is often chanced for multiple measurement, it means that the distances from the acoustic source to the two node are nearly common:

$$\|d_i - d_j\| \leq \epsilon_{ij} \tag{8}$$

where ρ is given threshold parameter. Inequality (8) is equivalent to the following two linear constraints

$$\begin{aligned} d_i - d_j \leq \epsilon_{ij} \\ d_j - d_i \leq \epsilon_{ij} \end{aligned} \tag{9}$$

B. Time Synchronization

Like the effect of TOA detection error, time synchronization error maybe also occur flip problem when computing the node sequence in LPSBL system, then lead to localization error. Traditional time synchronization protocol, such as RBS, TPSN, and FTSP, can achieve synchronization less than 100ns. Compared with the the measurement error of TOA, time synchronization error has little effect on the LPSBL system.

V. EVALUATION

This section presents a thorough evaluation of our sequence-based localization based on linear programming (LPSBL). In the first half of this section, we evaluate the parameters that affect the localization performance of LPSBL. In the second half, test-bed experiment is carried out in indoor environments to vrified the system performance.

A. Simulation

We develop a Monte Carlo simulator to evaluate the localization performance under different conditions. In the simulation, the nodes are randomly depolyed in a field of $10m \times 10m$. Considering the impact of the uncertainty of node position and TOA detection, we add a certain amount of node location error and TOA measure error in all the simulations. All the

TABLE I: **Default configuration parameter**

Parameter	Description
Field Area	10m × 10m
Number of Anchors	50 (Default)
Node Location Error	0.10m (Default)
TOA Detection Error	0.10ms (Default)
Random-Seed Loop	500 times (Default)
Error Statistics	RMSE

statistics are running more than 100 times for high confidence, and reported by RMSE figure. Table 1 illustrates the default simulation setup parameters.

The results of simulation evaluation are as following:

1) Impact of the number of anchors: In this experiment, we investigate the localization error and number of anchors with a different number of anchors from 10 to 40 in steps of 2. We run the simulation with the TOA error is 0.1ms, and other simulation parameters are default. Since the two localization methods being compared are aiming to locate the target by processing the anchors, but the SBL method can cause a big error, while the LPSBL method is aiming to avoid the appearance and get better result. We can speculate that with more anchors, the whole area will be divided into smaller parts, thus more accurate localization estimation should be achieved in the LPSBL method. Fig. 4 confirms our expectation. As shown in Fig. 4, with the number of anchors increases, localization error for both methods are down slowly. Fig. 4 also shows that the localization error of the LPSBL method is approximate to the SBL method when the number of anchor node is larger.

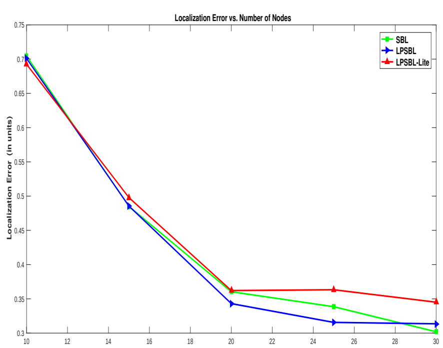

Fig. 4: Localization Error vs. Number of Anchors

2) Impact of the position error: In this experiment, we compare the two methods for different position errors of anchors. In Fig. 5, we choose the position error with the range from 0 to 0.4m in step of 0.02m for the two methods. Fig. 5 indicates the position error of anchors has an effect on the localization results. For the both methods, the localization error increases as position error increases in Fig. 5.

3) Impact of the TOA measurement error: In this experiment, we discusses the impact of the TOA error of anchors for the two methods being compared with the range from 0 to 2ms in steps of 0.1ms. Other simulation parameters

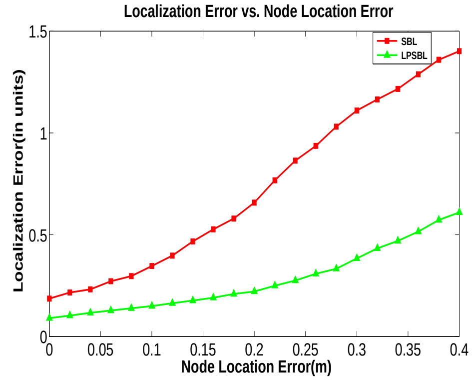

Fig. 5: Localization Error vs. Node position Error

keep default. Figure 6 reports the average localization error of both localization methods. Results clearly show a significant increases of the localization error as increases of TOA measurement error. In conclusion one can observe that the error of TOA measurement has significant influence on the localization accuracy.

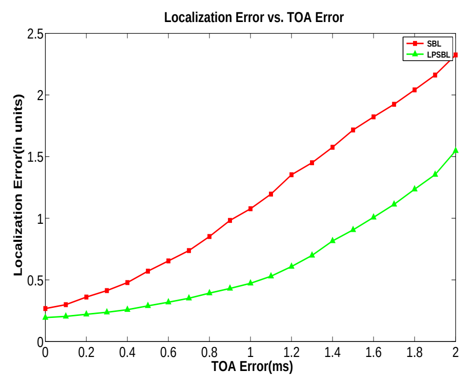

Fig. 6: Localization Error vs. TOA Error

B. *Emulation*

In this section, we report system implementation of our design based on smartphones. we use 30 Samsung smartphones as anchors and connect them through CISCO CVR328W-K9-CN wireless router. TPSN protocol is adapted in the proposed LPSBL system to realize time synchronization. The 30 smartphones are deployed in a size of 16m×10m space and there just one target during an experiment. In the experiment, smartphones are random deployed in the space, and 100 times localization results are showed in Fig. 7. In the figure, blue squares stand for anchor nodes, red circle squares are the real position of acoustic sources, and black dot are the estimated location by LPSBL. An arrow origins from the estimated location of each acoustic source and points to its real position. As the results showed in Fig.7, most of estimated locations are close to the ground truth and the errors between them are very small, which means that LPSBL can effectively localize the acoustic source with good robustness.

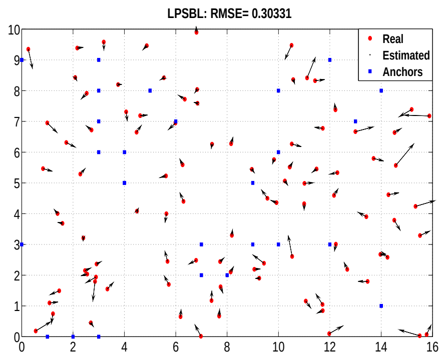

Fig. 7: Test-bed localization result of the proposed LPSBL

VI. Related Work

Acoustic source localization in sensor network is a widely-studied problem. In the past few years, there has been a growing interest for spatial distributions of independent (unsynchronized) acoustic sensors, each made of two or more synchronized microphones. Due to space constraints, we can only mention a few directly related works here. Wang, *et al.* [7] described a system having static cluster architecture, the system experienced a problem in that the accuracy decreased when an acoustic source occurred between the clusters. Chen, *et al.* [8] showed that nodes in the system did not need to recognize their cluster head, reducing the constraints on deployment of the localization system. Hu, *et al.* [9] design the system based on 2-tier architecture, which experienced cost and deployment problems especially in the very large target area. Rabbat, *et al.* [10] proposed a decentralized algorithm based on the distributed ML estimation technique using token ring architecture. Kim, *et al.* [11] proposed to identify the node closest to the acoustic source, based on TOA comparisons between all nodes, thus incurring high communication cost and requiring global synchronization between all sensor nodes. Lightning is a method proposed in [12] to identify the sensor closest to the acoustic source, also based on expensive broadcasting/flooding. In previous research on distributed acoustic source localization system, generally each node just has a single element. In the past few years, there has been a growing interest for acoustic nodes made of two or more synchronized microphones. Aarabi, *et al.* [13] used 10 dual-microphone arrays distributed in a room and used their data to locate three speakers. Wu, *et al.* [14] used three dual-microphone arrays to locate two sound sources in a distributed way in which only the local DOA estimates are communicated among arrays. Canclini, *et al.* [15], [16] proposed a method for localizing an acoustic source with distributed microphone networks based on TDOA between microphones of the same sensor.

Most of the existing acoustic source localization methods in sensor networks are based on range-based measurement. In contrast, our work is a range-free method and shown to be robust to the errors of node locations and the errors of measurements. There have existed some research on range-free localization method. Yedavalli, *et al.* [4] proposed a Sequence-Based Localization (SBL) method in WSN. The heart of SBL is the division of a 2D localization space into distinct regions by the perpendicular bisectors of lines joining pairs of reference nodes (nodes with known locations). In their earlier work [17], Ecolocation used location constraints for robust localization. Chakrabarty, *et al.* [18] and Ray, *et al.* [19] use identity codes to determine the location of sensor nodes in grid and nongrid sensor fields, respectively. e, *et al.* [20] propose an RF-based localization technique in which the unknown node location is determined by the intersection of all triangles, formed by reference nodes, that are likely to bound it. The unknown node determines its existence inside a triangle by comparing its measured RSS values to that of its neighbors to detect a trend in RSS values in any particular direction This technique depends on the weak assumption that signal strength decreases monotonically with distance, which is not true in real-world scenarios. Zhong, *et al.* [21] convert the original tracking problem to the problem of finding the shortest path in a graph, which is equivalent to optimal matching of a series of node sequences. Zhong, *et al.* [22] introduce a proximity metric called RSD to capture the distance relationships among 1-hop neighboring nodes in a range-free manner. Guo, *et al.* [23] proposed a novel method to detect nodes with data faults by ranking the nodes based on their sensing readings from the event. Shu, *et al.* [24] proposed a novel localization design that utilizes the unique Time of Charge (TOC) sequences among wireless rechargeable sensors. Zhong, *et al.* [25] presented a Multi-Sequence Positioning (MSP) method for sensor node localization by processing multiple one-dimensional node sequences. Yang, *et al.* [26] use the relative relationship information between RSS values as the fingerprint data in the Wi-Fi indoor positioning system.

VII. Conclusions

In this paper, we presented a simple and novel node sequence-based localization technique based on linear programming, LPSBL. In LPSBL, node sequences are used to uniquely identify distinct regions in the localization space. The reference node sequence is computed by using TOA measurements of acoustic signals between the acoustic source and the reference nodes. The inequality constraint is constructed by processing the nodes sequence, then turn the sequence-based localization into linear programming problem. Since our system runs on COTS smartphones and supports spontaneous setup, it has potential to enable a wide range of distributed acoustic source localization systems. Besides the basic design, advanced LPSBL is proposed for further enhancing system robustness. Our system is verified and evaluated through analysis, extensive simulation as well as the test-bed experimentation. The test results have shown that the proposed method can effectively implement aoustic source localization with ad-hoc smartphone array.

References

[1] Y. Guo and M. Hazas, “Localising speech, footsteps and other sounds using resource-constrained devices,” in *IEEE/ACM IPSN*, 2011, pp. 330–341.

[2] J. Sallai, A. Lédeczi, and P. Völgyesi, “Acoustic shooter localization with a minimal number of single-channel wireless sensor nodes,” in *ACM SenSys*, 2011, pp. 96–107.

[3] M. Allen, L. Girod, R. Newton, S. Madden, D. T. Blumstein, and D. Estrin, “Voxnet: An interactive, rapidly-deployable acoustic monitoring platform,” in *IEEE/ACM IPSN*, 2008, pp. 371–382.

[4] K. Yedavalli and B. Krishnamachari, “Sequence-based localization in wireless sensor networks,” *IEEE Transactions on Mobile Computing (TMC)*, vol. 7, no. 1, pp. 81–94, 2008.

[5] S. Boyd and L. Vandenberghe, “Convex optimization,” in *Cambridge University Press*, 2004.

[6] I. Griva, S. Nash, and A. Sofer, “Linear and nonlinear optimization,” in *SIAM Press*, 2009.

[7] Q. Wang, W.-P. Chen, R. Zheng, K. Lee, and L. Sha, “Acoustic target tracking using tiny wireless sensor devices,” in *IEEE/ACM IPSN’03*, 2003, pp. 642–657.

[8] W.-P. Chen, J. C. Hou, and L. Sha, “Dynamic clustering for acoustic target tracking in wireless sensor networks,” *IEEE Transactions on Mobile Computing (TMC)*, vol. 3, no. 3, pp. 258–271, 2004.

[9] W. Hu, N. Bulusu, C. T. Chou, S. Jha, A. Taylor, and V. N. Tran, “Design and evaluation of a hybrid sensor network for cane toad monitoring,” *ACM Transactions on Sensor Networks (TOSN)*, vol. 5, no. 1, p. 4, 2009.

[10] M. Rabbat, R. Nowak, and J. A. Bucklew, “Robust decentralized source localization via averaging,” in *IEEE ICASSP’05*, 2005, pp. 1057–1060.

[11] Y. Kim, J. Ahn, and H. Cha, “Locating acoustic events based on large-scale sensor networks,” *Sensors*, vol. 9, no. 12, pp. 9925–9944, 2009.

[12] Q. Wang, R. Zheng, A. Tirumala, X. Liu, and L. Sha, “Lightning: a hard real-time, fast, and lightweight low-end wireless sensor election protocol for acoustic event localization,” *IEEE Transactions on Mobile Computing (TMC)*, vol. 7, no. 5, pp. 570–584, 2008.

[13] P. Aarabi, “The fusion of distributed microphone arrays for sound localization,” *EURASIP Journal on Advances in Signal Processing*, vol. 2003, no. 4, pp. 338–347, 2003.

[14] T. Wu, L. Sun, Q. Cheng, and P. K. Varshney, “Fusion of multiple microphone arrays for blind source separation and localization,” in *IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM’12)*, 2012, pp. 173–176.

[15] A. Canclini, E. Antonacci, A. Sarti, and S. Tubaro, “Acoustic source localization with distributed asynchronous microphone networks,” *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)*, vol. 21, no. 2, pp. 439–443, 2013.

[16] A. Canclini, P. Bestagini, F. Antonacci, M. Compagnoni, A. Sarti, and S. Tubaro, “A robust and low-complexity source localization algorithm for asynchronous distributed microphone networks,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, vol. 23, no. 10, pp. 1563–1575, 2015.

[17] K. Yedavalli, B. Krishnamachari, S. Ravula, and B. Srinivasan, “Ecolocation: a sequence based technique for rf localization in wireless sensor networks,” in *IEEE/ACM IPSN*, 2005, p. 38.

[18] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, “Grid coverage for surveillance and target location in distributed sensor networks,” *IEEE Transactions on Computers (TOC)*, vol. 51, no. 12, pp. 1448–1453, 2002.

[19] S. Ray, D. Starobinski, A. Trachtenberg, and R. Ungrangsi, “Robust location detection with sensor networks,” *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 22, no. 6, pp. 1016–1025, 2004.

[20] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, “Range-free localization schemes for large scale sensor networks,” in *ACM MOBICOM*, 2003, pp. 81–95.

[21] Z. Zhong, T. Zhu, D. Wang, and T. He, “Tracking with unreliable node sequences,” in *IEEE INFOCOM*, 2009, pp. 1215–1223.

[22] Z. Zhong and T. He, “Rsd: a metric for achieving range-free localization beyond connectivity,” *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, vol. 22, no. 11, pp. 1943–1951, 2011.

[23] S. Guo, H. Zhang, Z. Zhong, J. Chen, Q. Cao, and T. He, “Detecting faulty nodes with data errors for wireless sensor networks,” *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 3, p. 40, 2014.

[24] Y. Shu, P. Cheng, Y. Gu, J. Chen, and T. He, “Toc: Localizing wireless rechargeable sensors with time of charge,” *ACM Transactions on Sensor Networks (TOSN)*, vol. 11, no. 3, p. 44, 2015.

[25] Z. Zhong and T. He, “Wireless sensor node localization by multisequence processing,” *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 11, no. 1, p. 3, 2012.

[26] S. Yang, P. Dessai, M. Verma, and M. Gerla, “Freeloc: Calibration-free crowdsourced indoor localization,” in *IEEE INFOCOM*, 2013, pp. 2481–2489.