

# A Simple Closed-Form Linear Source Localization Algorithm

Zhengyuan Xu, Ning Liu  
Department of Electrical Engineering  
University of California  
Riverside, CA 92521 USA  
*dxu@ee.ucr.edu*

Brian M. Sadler  
Army Research Laboratory  
AMSRD-ARL-CI-CN  
Adelphi, MD 20783 USA  
*bsadler@arl.army.mil*

**Abstract**—A closed-form linear solver is proposed to estimate a source location based on time difference of arrivals from the source measured by two neighboring sensors in a linear array. Those sensors are shown to be located at different foci of paired hyperbolas. Using a geometric representation of a hyperbola that is defined as the locus of points whose distance from the focus is proportional to the distance from a directrix, a set of linear equations for the x-coordinate of the source are derived, and subsequently the y-coordinate is solved from a set of equations describing Cartesian coordinates of source and sensors. The algorithm is applicable to both near-field and far-field scenarios without any approximation. It also enjoys simplicity and computational efficiency without the nonlinearity inherited from hyperbolic equations. The estimation error is characterized by the estimation performance of the time differences. Simulation results show that the proposed algorithm has a similar performance with the maximum-likelihood estimator at high signal to noise ratio.

## I. INTRODUCTION

Source localization has been one of the fundamental problems in radar, sonar, navigation and acoustic tracking [1], [2]. Recent work in this area includes direction of arrival (DOA) estimation in far-field cases and localization in near-field cases. For far-field cases, various narrowband techniques have been proposed [3], [4]. There has been renewed interest in locating wideband sources in the near field, including both the range and the DOA. Many localization methods for near-field cases have been proposed [5], such as an maximum likelihood (ML) method [6], covariance approximation method [7] and MUSIC method [8], [9].

For a sensor array, localization based on the time differences of arrival (TDOAs) was of much interest in the past [10]. A traditional solver using TDOA involves iterative processing [11]. An iterative method typically incurs high computational complexity and its convergence requires initial estimate of the position at a tolerable accuracy. Some optimum processing techniques have been proposed [12], [13], [14]. In [15], a closed-form solution is given, which is an approximated ML estimator when the TDOA measurement errors are small.

In many practical applications, low complexity is a requirement of high priority since sensors may be imposed significant

power constraints. A linear location solver with a linear array of sensors needs to be designed. Meanwhile, the absolute clock information from the source is mostly unavailable. Thus TDOA instead of TOA approaches must be employed. To the best of our knowledge, only linear spherical positioning solvers based on TOAs and nonlinear hyperbolic positioning solvers based on TDOAs have been proposed in the literature [16] although for the latter case it can be linearized by Taylor series expansion [15]. Hence in this paper, a new and close-form linear solver is proposed. Directrices of hyperbolas determined by TDOA measurements are introduced to obtain linear relations between the x-coordinate of the source and TDOAs of each sensor pair in a linear array on the x-axis. Only moderate number of scalar multiplications and additions are involved, and no matrix inversion is required. Meanwhile, it is applicable to both far-field and near-field setups. Localization performance is studied. Applying a perturbation technique, localization error and bias is derived analytically to be dependent on TDOA measurement errors.

The paper is organized as follows: In Section II, the geometry features of hyperbola are reviewed that are used to develop a linear solver of the x-coordinate of the source in Section III. Section IV provides error performance study when TDOAs have errors. Some simulation results are given in Section VI to evaluate the performance of the algorithm. In Section VII, conclusions and future research topics are provided.

## II. MATHEMATICAL MODEL

Consider a case of two sensors observing the signal from the source. The difference of the arrival time denoted by  $\tau$ , depends on the path difference, denoted by  $d$ . As the signal propagates at speed  $v_c$ , we have  $d = v_c\tau$ . So what we know from the two sensors is the path difference from the source point to the sensors. Since hyperbola is the set of points that have the same difference in distance to two fix points, the source is on a hyperbola whose parameters are determined by the difference of arrival time and the distance of the two sensors.

We start from the geometry definition of hyperbola to derive the geometry features of the directrix. Consider a hyperbola shown in Fig. 1, where  $F_1$  and  $F_2$  are two foci, whose distance is  $2c$ . The difference of the distance from any point on the

This research was supported in part by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011.

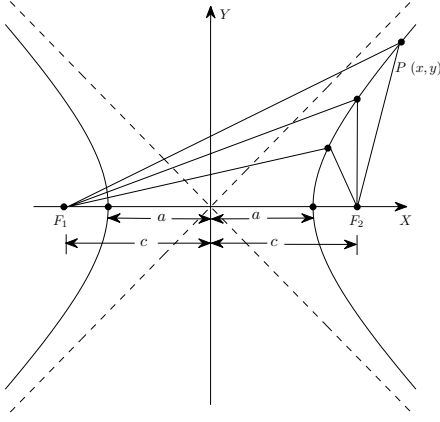


Fig. 1. Definition of hyperbola.

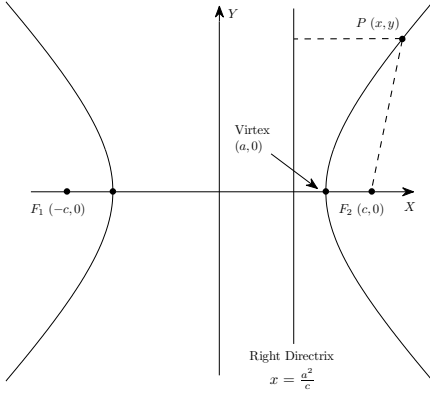


Fig. 2. Directrix of a hyperbola.

hyperbola to the two foci is  $2a$ . If we adopt a Cartesian coordinate system as Fig. 1, from the definition of hyperbola [17], we have

$$\sqrt{(x+c)^2 + y^2} - \sqrt{(x-c)^2 + y^2} = 2a. \quad (1)$$

Moving the second term to the right, squaring both sides and after some manipulations, it becomes

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1,$$

where  $b = \sqrt{c^2 - a^2}$ . It is the standard equation of the hyperbola. For the general case that the center of hyperbola is not at the origin, the standard equation is

$$\frac{(x-x_c)^2}{a^2} - \frac{(y-y_c)^2}{b^2} = 1, \quad (2)$$

where  $(x_c, y_c)$  is the location of the center.

A hyperbola can also be defined as the locus of points whose distance from the focus is proportional to the horizontal distance from a vertical line known as the conic section directrix, where the ratio is the eccentricity  $e = \frac{c}{a}$ . As shown in Fig. 2, the equation of the right directrix is  $x = x_R = \frac{a^2}{c}$ . So for any point  $P(x_P, y_P)$  on the hyperbola, the distance

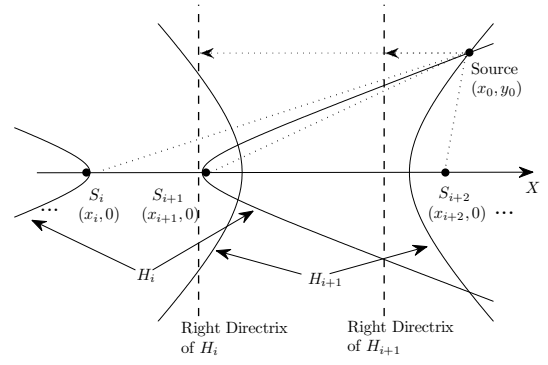


Fig. 3. A linear sensor array with  $M$  sensors on X-axis. The signal source is on the right side of the array.

$D(P, F_2)$  to the right focus  $F_2(c, 0)$  can be expressed simply according to the x-coordinate as

$$D(P, F_2) = e|x_P - x_R|. \quad (3)$$

Following the same line and using the left directrix  $x = x_L = -\frac{a^2}{c}$ , the distance  $D(P, F_1)$  of any point  $P(x_P, y_P)$  on the hyperbola to the left focus  $F_1(-c, 0)$  is

$$D(P, F_1) = e|x_P - x_L|. \quad (4)$$

These two equations are fundamental to our development of a simple closed-form linear solver next.

### III. LINEAR LOCALIZATION ALGORITHM

Assume a linear sensor array with  $M$  sensors, from  $S_1$  to  $S_M$ , is placed on the x-axis of a 2-D plane as shown in Fig. 3. The x-coordinate of  $S_i$  is  $x_i$  for  $i = 1, 2, \dots, M$ . The signal source  $S_0$  is at  $(x_0, y_0)$ . The hyperbola  $H_i$  centered at the midpoint  $((x_i + x_{i+1})/2, 0)$  of two sensors with foci  $S_i$  and  $S_{i+1}$  is determined by the TDOA  $\tau_i$  between these two neighbored sensors. Thus the distance difference of arrival can be written in terms of TDOA as

$$d_i = v_c \tau_i = r_i - r_{i+1}, \quad i = 1, 2, \dots, M-1, \quad (5)$$

where  $r_i$  is the distance from the signal source to sensor  $S_i$ .

The parameters of hyperbola  $H_i$  can be obtained as

$$a_i = \frac{|d_i|}{2}, \quad c_i = \frac{x_{i+1} - x_i}{2},$$

$$e_i = \frac{c_i}{a_i} = \frac{x_{i+1} - x_i}{|d_i|}, \quad (6)$$

$$x_L^i = \frac{x_i + x_{i+1}}{2} - \frac{a_i^2}{c_i} = \frac{x_i + x_{i+1}}{2} - \frac{d_i^2}{2(x_{i+1} - x_i)}, \quad (7)$$

$$x_R^i = \frac{x_i + x_{i+1}}{2} + \frac{a_i^2}{c_i} = \frac{x_i + x_{i+1}}{2} + \frac{d_i^2}{2(x_{i+1} - x_i)}, \quad (8)$$

where  $e_i$  is the eccentricity of  $H_i$ ,  $x_L^i$  and  $x_R^i$  are the x-coordinates of intersections of the left and right directrices of  $H_i$  with the horizontal axis respectively.

Since  $S_{i+1}$  ( $i = 1, 2, \dots, M-2$ ) is both the right focus of hyperbola  $H_i$  and the left focus of hyperbola  $H_{i+1}$ , the

distance from the source  $S_0$  to the sensor  $S_{i+1}$  can be equally expressed in terms of the directrices and eccentricities of both  $H_i$  and  $H_{i+1}$  as

$$e_i|x_0 - x_R^i| = e_{i+1}|x_0 - x_L^{i+1}|. \quad (9)$$

As we know  $x_R^i < x_L^{i+1}$  always holds. If we introduce an indicator of relative location of the source to sensors by  $\nu_i$  that takes -1 when  $x_R^i < x_0 < x_L^{i+1}$  and takes 1 otherwise, the above equation becomes

$$e_i(x_0 - x_R^i) = \nu_i(e_{i+1}x_0 - e_{i+1}x_L^{i+1}). \quad (10)$$

Substituting (6), (7) and (8), (10) becomes

$$\begin{aligned} & \left[ \frac{x_{i+1} - x_i}{|d_i|} - \frac{\nu_i(x_{i+2} - x_{i+1})}{|d_{i+1}|} \right] x_0 \\ &= \frac{x_{i+1}^2 - x_i^2}{2|d_i|} - \frac{\nu_i(x_{i+2}^2 - x_{i+1}^2)}{2|d_{i+1}|} + \frac{\nu_i|d_{i+1}| + |d_i|}{2}. \end{aligned} \quad (11)$$

Notice that all quantities are known except  $x_0$ . Therefore, it can be easily solved. In order to improve estimation accuracy, we utilize all  $M-2$  consecutive hyperbola pairs. Considering  $i = 1, \dots, M-2$ , the location model can be written in a vector form

$$\mathbf{e}x_0 = \mathbf{g}, \quad (12)$$

where

$$\mathbf{e} = \begin{bmatrix} \frac{x_2 - x_1}{|d_1|} - \frac{\nu_1(x_3 - x_2)}{|d_2|} \\ \vdots \\ \frac{x_{M-1} - x_{M-2}}{|d_{M-2}|} - \frac{\nu_{M-2}(x_M - x_{M-1})}{|d_{M-1}|} \end{bmatrix},$$

$$\mathbf{g} = \begin{bmatrix} \frac{x_2^2 - x_1^2}{2|d_1|} - \frac{\nu_1(x_3^2 - x_2^2)}{2|d_2|} + \frac{\nu_1|d_2| + |d_1|}{2} \\ \vdots \\ \frac{x_{M-1}^2 - x_{M-2}^2}{2|d_{M-2}|} - \frac{\nu_{M-2}(x_M^2 - x_{M-1}^2)}{2|d_{M-1}|} + \frac{\nu_{M-2}|d_{M-1}| + |d_{M-2}|}{2} \end{bmatrix}.$$

Eq. (12) is over-determined. By minimizing the following error

$$\hat{x}_0 = \arg \min_{x_0} \|\mathbf{g} - x_0 \mathbf{e}\|^2, \quad (13)$$

the least square solution is

$$\hat{x}_0 = (\mathbf{e}^T \mathbf{e})^{-1} \mathbf{e}^T \mathbf{g}. \quad (14)$$

Notice that  $\mathbf{e}^T \mathbf{e}$  is a scalar. No matrix inversion is required, but vector multiplications. The length of each vector is  $M-2$ .

From  $x_0$ , the location model for  $y_0$  can be obtained from  $M-1$  hyperbola equations following a form of (2) as

$$\mathbf{f}y_0^2 = \mathbf{h}, \quad (15)$$

in which

$$\mathbf{f} = \begin{bmatrix} \frac{4}{(x_2 - x_1)^2 - d_1^2} \\ \vdots \\ \frac{4}{(x_{M-1} - x_{M-2})^2 - d_{M-1}^2} \end{bmatrix}, \quad (16)$$

$$\mathbf{h} = \begin{bmatrix} \left(\frac{2x_0 - x_1 - x_2}{d_1}\right)^2 \\ \vdots \\ \left(\frac{2x_0 - x_{M-1} - x_M}{d_{M-1}}\right)^2 \end{bmatrix} - \mathbf{1}, \quad (17)$$

that suggests

$$y_0 = \pm \sqrt{(\mathbf{f}^T \mathbf{f})^{-1} \mathbf{f}^T \mathbf{h}}. \quad (18)$$

By replacing  $x_0$  by its estimate  $\hat{x}_0$  in  $\mathbf{h}$ , an estimate of  $y_0$  is found from (18).

The polarity of  $y_0$  and the value of  $\nu_i$  depend on the source location relative to sensors. In a practical system, some prior information can be acquired through network or other measurements. For example, to determine  $\nu_i$ , compare  $r_i, r_{i+1}$  and  $r_{i+2}$ . If  $r_{i+1} < r_i$  and  $r_{i+1} < r_{i+2}$ , equivalently  $d_i > 0$  and  $d_{i+1} < 0$ , then  $\nu_i = -1$ . Otherwise  $\nu_i = 1$ . In the following development, we restrict our attention to the case where the source sensor is located to the right of the sensor array, leading to  $\nu_i = 1$  for all  $i = 1, \dots, M-2$  and  $d_i > 0$  for all  $i = 1, \dots, M-1$ . However, subsequent discussions are directly applicable to other scenarios. We also assume  $y_0 > 0$  to further resolve the ambiguity in estimating the source  $y$ -coordinate later on.

To summarize, (14) and (18) are the proposed linear solver for the source location.

For comparison purposes, the ML estimator (MLE) can be derived assuming all noise samples from measuring  $d_i$  are jointly Gaussian. Denote noise covariance matrix by  $\mathbf{C}_d$ . Then the MLE can be described by

$$(x_0^{ML}, y_0^{ML}) = \arg \min_{(x_0, y_0)} \mathbf{m}^T \mathbf{C}_d^{-1} \mathbf{m}, \quad (19)$$

where the  $i$ -th ( $i = 1, \dots, M-1$ ) entry of  $\mathbf{m}$  is  $\sqrt{(x_0 - x_i)^2 + y_0^2} - \sqrt{(x_0 - x_{i+1})^2 + y_0^2} - d_i$ .

#### IV. ERROR ANALYSIS

The estimation error of the source location may be from the errors of distance difference (or equivalently TDOA) measurements or inaccurate coordinates of sensors. Here we only consider the distance difference measurement errors denoted by  $\delta \mathbf{d} = [\delta d_1, \dots, \delta d_{M-1}]^T$ . For simplicity, assume these errors have zero means. However, the following analyses can easily incorporate non-zero means. If they are independent, then  $\mathbf{C}_d$  becomes diagonal

$$\mathbf{C}_d = E\{(\delta \mathbf{d})(\delta \mathbf{d})^T\} = \text{diag}\{\sigma_1^2, \dots, \sigma_{M-1}^2\}. \quad (20)$$

These errors will introduce errors to  $\mathbf{e}$  and  $\mathbf{g}$  in (14). Denoting the square error vector as  $\boldsymbol{\epsilon} = [(\delta d_1)^2, \dots, (\delta d_{M-1})^2]^T$  and using second order perturbation analysis, their errors can be directly found to be

$$\delta \mathbf{e} \approx \mathbf{P} \delta \mathbf{d} + \mathbf{R} \boldsymbol{\epsilon}, \quad (21)$$

$$\delta \mathbf{g} \approx \mathbf{Q} \delta \mathbf{d} + \mathbf{S} \boldsymbol{\epsilon}, \quad (22)$$

where

$$\mathbf{P} = \begin{bmatrix} p_1 & -p_2 & 0 & \dots & 0 \\ 0 & p_2 & -p_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & p_{M-2} & -p_{M-1} \end{bmatrix},$$

$$\begin{aligned}
\mathbf{Q} &= \begin{bmatrix} \frac{q_1+1}{2} & \frac{-q_2+1}{2} & 0 & \dots & 0 \\ 0 & \frac{q_2+1}{2} & \frac{-q_3+1}{2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{q_{M-2}+1}{2} & \frac{-q_{M-1}+1}{2} \end{bmatrix}, \\
\mathbf{R} &= \begin{bmatrix} r_1 & -r_2 & 0 & \dots & 0 \\ 0 & r_2 & -r_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & r_{M-2} & -r_{M-1} \end{bmatrix}, \\
\mathbf{S} &= \begin{bmatrix} s_1 & -s_2 & 0 & \dots & 0 \\ 0 & s_2 & -s_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & s_{M-2} & -s_{M-1} \end{bmatrix}, \quad (23)
\end{aligned}$$

whose entries are related to  $p_i$ ,  $q_i$ ,  $r_i$  and  $s_i$  defined as

$$\begin{aligned}
p_i &= \frac{x_i - x_{i+1}}{d_i^2}, \\
q_i &= \frac{x_i^2 - x_{i+1}^2}{d_i^2}, \\
r_i &= \frac{x_{i+1} - x_i}{d_i^3}, \\
s_i &= \frac{x_{i+1}^2 - x_i^2}{2d_i^3}, \quad (24)
\end{aligned}$$

for  $i = 1, \dots, M-1$ .

These errors lead to an error  $\delta x_0$  in estimating  $x_0$ . It can be derived from (14). Replace  $\mathbf{e}$  by  $\mathbf{e} + \delta \mathbf{e}$  and  $\mathbf{g}$  by  $\mathbf{g} + \delta \mathbf{g}$  respectively, and substitute (21) and (22). The estimation error up to the second order of  $\delta \mathbf{d}$  can be easily found to be

$$\delta x_0 \approx \mathbf{u}^T \delta \mathbf{d} + \delta \mathbf{d}^T \mathbf{V} \delta \mathbf{d} + \mathbf{w}^T \boldsymbol{\epsilon}, \quad (25)$$

where

$$\begin{aligned}
\mathbf{u} &= \frac{(\mathbf{e}^T \mathbf{e}) \mathbf{P}^T \mathbf{g} - 2(\mathbf{e}^T \mathbf{g}) \mathbf{P}^T \mathbf{e} + (\mathbf{e}^T \mathbf{e}) \mathbf{Q}^T \mathbf{e}}{(\mathbf{e}^T \mathbf{e})^2}, \\
\mathbf{V} &= \frac{\mathbf{P}^T \mathbf{Q}}{\mathbf{e}^T \mathbf{e}} + \frac{4(\mathbf{e}^T \mathbf{g}) \mathbf{P}^T \mathbf{e} \mathbf{e}^T \mathbf{P}}{(\mathbf{e}^T \mathbf{e})^3} \\
&\quad - \frac{2\mathbf{P}^T \mathbf{e} \mathbf{g}^T \mathbf{P} + 2\mathbf{P}^T \mathbf{e} \mathbf{e}^T \mathbf{Q} + (\mathbf{e}^T \mathbf{g}) \mathbf{P}^T \mathbf{P}}{(\mathbf{e}^T \mathbf{e})^2}, \\
\mathbf{w} &= \frac{(\mathbf{e}^T \mathbf{e}) \mathbf{R}^T \mathbf{g} - 2(\mathbf{e}^T \mathbf{g}) \mathbf{R}^T \mathbf{e} + (\mathbf{e}^T \mathbf{e}) \mathbf{S}^T \mathbf{e}}{(\mathbf{e}^T \mathbf{e})^2}. \quad (26)
\end{aligned}$$

The second order terms in (25) helps to obtain the mean error  $E\{\delta x_0\}$  of the estimate. Expressing  $\boldsymbol{\epsilon}$  as  $\text{diag}\{(\delta d_1)^2, \dots, (\delta d_{M-1})^2\} \mathbf{1}$  where  $\mathbf{1} = [1, \dots, 1]^T$ , the mean error is found to be

$$E\{\delta x_0\} = \text{tr}\{\mathbf{V} \mathbf{C}_d\} + \mathbf{w}^T (\mathbf{C}_d \odot \mathbf{I}) \mathbf{1}, \quad (27)$$

where  $\odot$  represents Hadamard product,  $\mathbf{I}$  is an identity matrix of dimension  $M-1$ .

Keeping the first order term in (25), the mean-square error (MSE) in estimating  $x_0$ , defined as  $E\{[\hat{x}_0 - x_0]^2\}$ , is obtained as

$$E\{(\delta x_0)^2\} \approx \mathbf{u}^T \mathbf{C}_d \mathbf{u}. \quad (28)$$

Eq. (28) shows that the MSE is approximately proportional to the variance of  $d_i$ . This trend will also be observed from the simulation results in the next section.

To obtain the estimation error for  $y_0$ , we consider  $y_0^2$  first for convenience. Notice that the square of (18) has a similar form as (14). The errors in measuring all  $d_i$  bring errors to  $\mathbf{f}$  and  $\mathbf{h}$  up to the second order of  $\delta \mathbf{d}$  as

$$\delta \mathbf{f} \approx \mathbf{A} \delta \mathbf{d} + \mathbf{J} \boldsymbol{\epsilon}, \quad (29)$$

$$\delta \mathbf{h} \approx \mathbf{B} \delta \mathbf{d} + \mathbf{l} \delta \mathbf{d}^T \mathbf{V} \delta \mathbf{d} + \mathbf{K} \delta \mathbf{d} \delta \mathbf{d}^T \mathbf{u} + \mathbf{l} \mathbf{w}^T \boldsymbol{\epsilon}, \quad (30)$$

where

$$\mathbf{A} = \text{diag}\{a_1, \dots, a_{M-1}\},$$

$$\mathbf{J} = \text{diag}\{j_1, \dots, j_{M-1}\},$$

$$\mathbf{l} = [l_1, \dots, l_{M-1}]^T,$$

$$\mathbf{B} = \text{diag}\{b_1, \dots, b_{M-1}\} + \mathbf{l} \mathbf{u}^T,$$

$$\mathbf{K} = \text{diag}\{k_1, \dots, k_{M-1}\} + \left[ \frac{4}{d_1^2}, \dots, \frac{4}{d_{M-1}^2} \right]^T \mathbf{u}^T, \quad (31)$$

whose entries are related to  $a_i$ ,  $b_i$ ,  $j_i$ ,  $k_i$  and  $l_i$  defined as

$$a_i = \frac{8d_i}{[(x_{i+1} - x_i)^2 - d_i^2]^2},$$

$$b_i = \frac{-2(2x_0 - x_{i+1} - x_i)^2}{d_i^3},$$

$$j_i = \frac{4(x_{i+1} - x_i)^2 + 12d_i^2}{[(x_{i+1} - x_i)^2 - d_i^2]^3},$$

$$k_i = \frac{8(x_{i+1} - x_i - 2x_0)}{d_i^3},$$

$$l_i = \frac{4(2x_0 - x_{i+1} - x_i)}{d_i^2},$$

for  $i = 1, \dots, M-1$ .

Applying the similar perturbation analysis technique as before, the error in estimating  $y_0^2$  up to the second order of  $\delta \mathbf{d}$  can be derived from as

$$\delta(y_0^2) \approx \boldsymbol{\eta}^T \delta \mathbf{d} + \delta \mathbf{d}^T \mathbf{T} \delta \mathbf{d} + \boldsymbol{\lambda}^T \boldsymbol{\epsilon}. \quad (32)$$

where

$$\boldsymbol{\eta} = \frac{(\mathbf{f}^T \mathbf{f}) \mathbf{A}^T \mathbf{h} - 2(\mathbf{f}^T \mathbf{h}) \mathbf{A}^T \mathbf{f} + (\mathbf{f}^T \mathbf{f}) \mathbf{B}^T \mathbf{f}}{(\mathbf{f}^T \mathbf{f})^2},$$

$$\begin{aligned}
\mathbf{T} &= \frac{\mathbf{A}^T \mathbf{B} + \mathbf{f}^T \mathbf{l} \mathbf{v} + \mathbf{u} \mathbf{f}^T \mathbf{K}}{\mathbf{f}^T \mathbf{f}} + \frac{4(\mathbf{f}^T \mathbf{h}) \mathbf{A}^T \mathbf{f} \mathbf{f}^T \mathbf{A}}{(\mathbf{f}^T \mathbf{f})^3} \\
&\quad - \frac{2\mathbf{A}^T \mathbf{f} \mathbf{h}^T \mathbf{A} + 2\mathbf{A}^T \mathbf{f} \mathbf{f}^T \mathbf{B} + (\mathbf{f}^T \mathbf{h}) \mathbf{A}^T \mathbf{A}}{(\mathbf{f}^T \mathbf{f})^2},
\end{aligned}$$

$$\lambda = \frac{(\mathbf{f}^T \mathbf{f}) \mathbf{J}^T \mathbf{h} - 2(\mathbf{f}^T \mathbf{h}) \mathbf{J}^T \mathbf{f} + (\mathbf{f}^T \mathbf{f}) \mathbf{w} \mathbf{l}^T \mathbf{f}}{(\mathbf{f}^T \mathbf{f})^2}. \quad (33)$$

Since  $\delta(y_0^2) = 2y_0\delta y_0$ , using (32), the mean error and MSE in estimating  $y_0$  can be found in terms of  $\delta(y_0^2)$  as follows

$$E\{\delta y_0\} \approx \frac{E\{\delta(y_0^2)\}}{2y_0} = \frac{\text{tr}\{\mathbf{T}\mathbf{C}_d\} + \boldsymbol{\lambda}^T (\mathbf{C}_d \odot \mathbf{I}) \mathbf{1}}{2y_0}, \quad (34)$$

$$E\{(\delta y_0)^2\} \approx \frac{E\{[\delta(y_0^2)]^2\}}{4y_0^2} = \frac{\boldsymbol{\eta}^T \mathbf{C}_d \boldsymbol{\eta}}{4y_0^2}. \quad (35)$$

These analytical results will be verified by simulations.

The advantage of the proposed estimator is that it is a linear solver in a closed-form and thus has a much lower computation complexity compared to the MLE or other iterative algorithms. If the MLE is implemented using a gradient method, the number of multiplications is at the order of  $6(M-1)N$ , where  $N$  is the number of iterations. The proposed estimator is at the order of  $6M$ . When the required precision is high,  $N$  for MLE could be a very large number. Moreover the proposed linear estimator does not have a slow convergence problem since it does not need an iterating process.

## V. IMPROVED LINEAR LOCALIZATION ALGORITHM

In the above proposed algorithm, linear localization equations (12) and (15) are solved by a standard least squares (LS) technique. To improve the localization performance, it is possible to use instead a weighted LS procedure. Then the covariance matrices of the error vectors on both x- and y-coordinates need to be found firstly. In practice, the covariance matrices can be obtained by using previous estimated location through an iterative process. The following derivation is based on the theoretical error analysis in the previous section. The error vector in estimating X-coordinate is

$$\phi_x = \delta \mathbf{g} - x_0 \delta \mathbf{e} \approx (\mathbf{Q} - x_0 \mathbf{P}) \delta \mathbf{d} + (\mathbf{S} - x_0 \mathbf{R}) \boldsymbol{\epsilon}, \quad (36)$$

and the mean vector of  $\phi_x$  is

$$E\{\phi_x\} = E\{\delta \mathbf{g}\} - x_0 E\{\delta \mathbf{e}\} \approx (\mathbf{S} - x_0 \mathbf{R}) (\mathbf{C}_d \odot \mathbf{I}) \mathbf{1}, \quad (37)$$

because  $E\{\delta \mathbf{d}\} = 0$ . In practice the second-order item is small. When ignoring the second-order item in (36),  $\phi_x$  becomes a Gaussian random vector, and the covariance matrix is

$$\boldsymbol{\Phi}_x = E\{\phi_x \phi_x^T\} \approx (\mathbf{Q} - x_0 \mathbf{P}) \mathbf{C}_d (\mathbf{Q} - x_0 \mathbf{P})^T \quad (38)$$

Similarly, the error vector in estimating y-coordinate is

$$\begin{aligned} \phi_{y^2} &= \delta \mathbf{h} - y_0^2 \delta \mathbf{f} \approx (\mathbf{B} - y_0^2 \mathbf{A}) \delta \mathbf{d} + (\mathbf{l} \mathbf{w}^T - y_0^2 \mathbf{J}) \boldsymbol{\epsilon} \\ &\quad + \mathbf{l} \delta \mathbf{d}^T \mathbf{V} \delta \mathbf{d} + \delta \mathbf{d}^T \mathbf{u} \mathbf{K} \delta \mathbf{d}. \end{aligned} \quad (39)$$

The mean vector of  $\phi_{y^2}$  is

$$\begin{aligned} E\{\phi_{y^2}\} &= E\{\delta \mathbf{h}\} - y_0^2 E\{\delta \mathbf{f}\} \approx (\mathbf{l} \mathbf{w}^T - y_0^2 \mathbf{J}) (\mathbf{C}_d \odot \mathbf{I}) \mathbf{1} \\ &\quad + \mathbf{l} \text{tr}\{\mathbf{V} \mathbf{C}_d\} + \mathbf{K} \mathbf{C}_d \mathbf{u}, \end{aligned} \quad (40)$$

and the covariance matrix when ignoring the second-order items is

$$\boldsymbol{\Phi}_{y^2} = E\{\phi_{y^2} \phi_{y^2}^T\} \approx (\mathbf{B} - y_0^2 \mathbf{A}) \mathbf{C}_d (\mathbf{B} - y_0^2 \mathbf{A})^T \quad (41)$$

With the covariance matrix, the x-coordinate estimation in (13) and (14) is optimized as

$$\begin{aligned} \hat{x}_0 &= \arg \min_{x_0} \{(\mathbf{g} - x_0 \mathbf{e})^T \boldsymbol{\Phi}_x^{-1} (\mathbf{g} - x_0 \mathbf{e})\} \\ &= (\mathbf{e}^T \boldsymbol{\Phi}_x^{-1} \mathbf{e})^{-1} \mathbf{e}^T \boldsymbol{\Phi}_x^{-1} (\mathbf{g} + E\{\phi_x\}), \end{aligned} \quad (42)$$

which can be recognized as the weighted LS solution of (12). Also the improved y-coordinate estimation can be found as

$$\hat{y}_0 = \pm \sqrt{(\mathbf{f}^T \boldsymbol{\Phi}_{y^2}^{-1} \mathbf{f})^{-1} \mathbf{f}^T \boldsymbol{\Phi}_{y^2}^{-1} (\mathbf{h} + E\{\phi_{y^2}\})}. \quad (43)$$

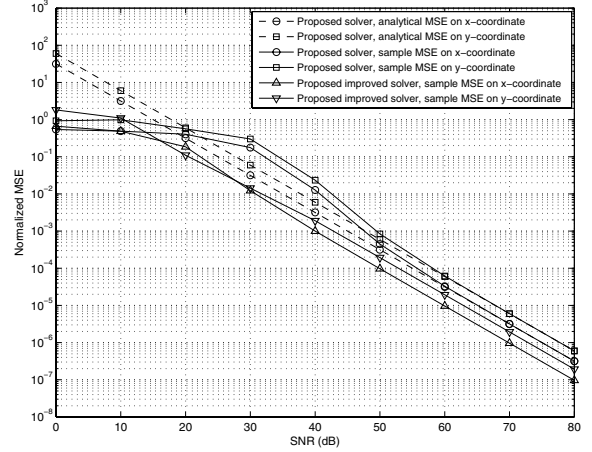


Fig. 4. Comparison of normalized MSE on both x- and y-coordinate between the proposed and the proposed improved algorithms.

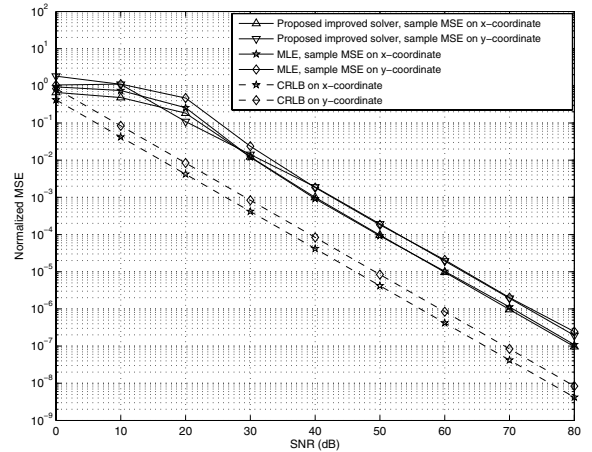


Fig. 5. Comparison of normalized MSE on both x- and y-coordinate between the proposed improved algorithm, MLE and CRLB.

## VI. SIMULATION RESULTS

We verify our theoretical development by simulation and compare the proposed and improved localization algorithms with the MLE and CRLB. For simplicity, the additive noises in distance difference measurements are assumed white Gaussian distributed and the SNR of all sensor pairs are identical as  $1/\sigma^2$ . Those measurements are generated by adding random

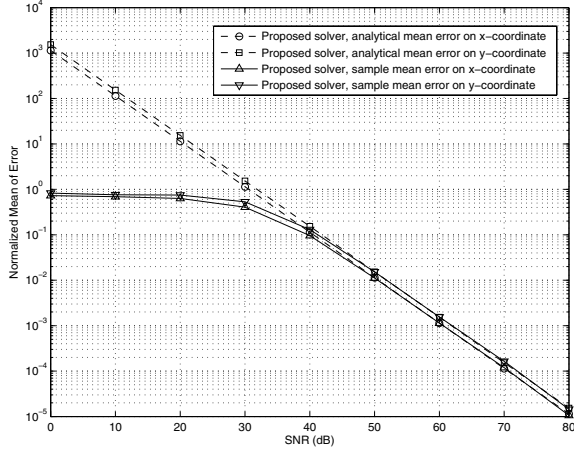


Fig. 6. Normalized mean errors on both x- and y-coordinate.

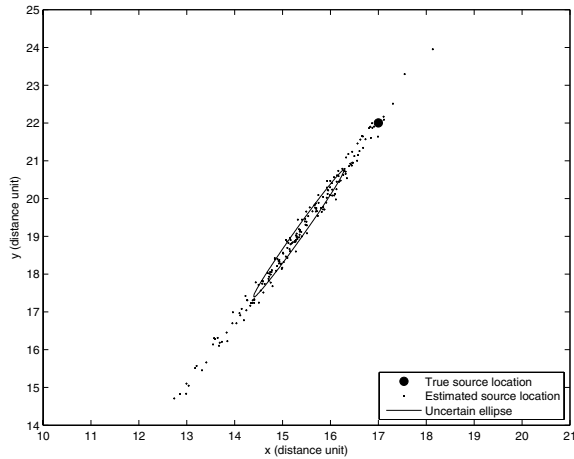


Fig. 7. Estimated positions from 200 realizations under SNR=40dB.

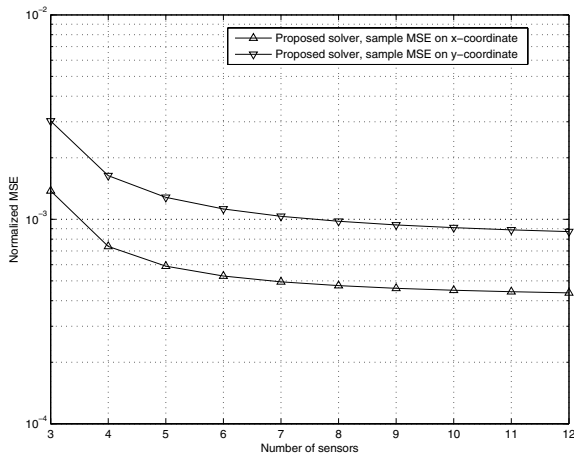


Fig. 8. Effect of the number of sensors.

noises (with zero mean and diagonal covariance matrix  $C_d$ ) to the true distance differences.

Fig. 4 compares the normalized MSE of the original proposed least square algorithm with the improved weighted one for a linear array with  $M = 10$  sensors. The sensor positions are  $(x_i = i - 1, y_i = 0), i = 1, 2, \dots, 10$ . The source is at  $(x_0 = 17, y_0 = 22)$ . The SNR range is  $[0, 80]$ dB. The normalized MSEs of  $E\{(\hat{x}_0 - x_0)^2\}/x_0^2$  and  $E\{(\hat{y}_0 - y_0)^2\}/y_0^2$  are obtained from the average of 50,000 independent realizations. In the high SNR region each MSE of the proposed and improved algorithms decrease linearly with the increase of SNR, and the performance of the original proposed algorithm approaches the improved solution. While in the low SNR region, the MSEs of the original algorithm do not have a linear relation with SNR. But thanks to the weighting covariance matrices, the MSE of improved weighted least-square solver still varies linearly and has a 5dB gain compared to the original algorithm.

Fig. 5 compares the normalized MSE of the improved proposed algorithm with MLE and CRLB. The linear sensor array is configured the same as previous and the simulation condition is also the same. Each MSE of the improved algorithm approaches the MLE at high SNR at much lower complexity. The simulated MSE is observed to converge to the analytical MSE starting from about SNR=30dB. The outlier at low SNR regime appears to be similar to typical time delay estimation performance [18]. Fig. 6 plots the normalized mean errors of the original proposed algorithm for both x-coordinate and y-coordinate of the source and compares them with the analytical results.

Fig. 7 shows a scatter plot of the source location estimated for 200 realizations at SNR=40dB. The uncertain ellipse [19],[20] is also plotted on this figure for comparison. The estimated source locations obviously have an average bias on both x and y directions about -1.7 and -2.6 units respectively. That means the proposed location estimator is a biased estimator. This bias can be compensated by a mean error estimator. The bias in estimating x-coordinate is smaller than that in y-coordinate.

The effect of the number of sensors  $M$  is demonstrated by Fig. 8. In this simulation the ratio of  $10 \log_{10}(\frac{\sigma_1^2}{\sigma_2^2})$  of each sensor pair is set to 40dB. The positions of  $S_1$  and  $S_M$  are fixed and the spacing between uniform sensors decreases with the increased  $M$ . It can be seen that the MSE decreases quickly when  $M$  increases from 3 to 6 while slowly when  $M$  is larger than 7.

## VII. CONCLUSIONS AND FUTURE WORK

This paper presented a non-iterative linear algorithm to locate the source on a 2-D plane based on the TDOA measurements of a linear sensor array on the x-axis. The directrices of a set of paired hyperbolas were exploited to obtain linear relations between the x-coordinate of the source and source's distances to sensors. Therefore the problem of localization was solved by a LS estimation technique. The analytical results showed the mean estimation error and MSE

linearly decrease with increased SNR. Compared with the MLE, the algorithm shows some performance loss at low SNR, while approaching the MLE at high SNR at much lower computational complexity. However, the improved LS algorithm shows similar performance as the MLE.

In practical environments, the bias of the estimator can be compensated through mean and SNR estimation. The resulting estimator performance will be compared with currently proposed LS and weighted LS algorithms in the future. The ranging and angle estimation errors will be derived from current coordinate estimation errors. The effects of sensor calibration errors and the mobility of the signal source will be studied.

## REFERENCES

- [1] G. Arslan and F. A. Sakarya, "A unified neural-network-based speaker localization technique," *IEEE Trans. on Neural Networks*, vol. 11, no. 4, pp. 997-1002, Jul. 2000.
- [2] F. Asano, H. Asoh, and T. Matsui, "Sound source localization and signal separation for office robot" *Multisensor Fusion and Integration for Intelligent Systems*, vol. 15-18, pp. 243-248, Aug. 1999.
- [3] F. C. Schwegge, "Sensor array data processing for multiple signal sources," *IEEE Trans. on Information Theory*, vol. 14, no. 2, pp. 294-305, Mar. 1968.
- [4] I. Ziskind and M. Wax, "Maximum likelihood localization of multiple sources by alternating projection," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 36, no. 10, pp. 1553-1560, Oct. 1998.
- [5] E. Grosicki, K. Abed-Meraim, and Y. Hua, "A weighted linear prediction method for near-field source localization," *IEEE trans. on Signal Processing*, vol. 53, no. 10, pp. 3651-3660, 2005.
- [6] J. C. Chen and K. Yao, "Wideband signal detection and maximum-likelihood source localization," *Signals, Systems and Computers*, vol. 2, no. 4-7, pp. 931-935, Nov. 2001.
- [7] J.-H. Lee, Y.-M. Chen, and C.-C. Yeh, "A covariance approximation method for near-field direction finding using a uniform linear array," *IEEE Trans. on Signal Processing*, vol. 43, no. 5, pp. 1293-1298, May 1995.
- [8] Y. D. Huang and M. Barka, "Near-field multiple source localization by passive sensor array," *IEEE Trans. on Antennas and Propagation*, vol. 39, no. 7, pp. 968-975, Jul. 1991.
- [9] H.-S. Hung, S.-H. Chang, and C.-H. Wu, "3-d music with polynomial rooting for near-field source localization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 6, no. 7-10, pp. 3065-3068, May 1996.
- [10] H. Lee, "A Novel procedure for assessing the accuracy of hyperbolic multilateration systems," *IEEE Trans. on Aerospace and Electronic Systems*, vol. AES-11, no. 1, pp. 2-15, Jan. 1975.
- [11] W. H. Foy, "Position-location solutions by Taylor-series estimation," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 12, pp. 187-194, Mar. 1976.
- [12] G. Carter, "Time delay estimation for passive sonar signal processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 463-470, June. 1981.
- [13] W. Hahn, "Optimum signal processing for passive sonar range and bearing estimation," *J. Acoust. Soc. Am.*, vol. 58, pp. 201-207, Jan. 1975.
- [14] J. S. Abel and J. O. Smith, "Source range and depth estimation from multipath range difference measurements," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1157-1165, Aug. 1989.
- [15] Y. T. Chan, "A simple and efficient estimator for hyperbolic location," *IEEE Trans. on Signal Processing*, vol. 42, pp. 1905-1915, Aug. 1994.
- [16] M.-G. Di Benedetto and G. Giancola, *Understanding Ultra Wide Band Radio Fundamentals*, Prentice Hall, Upper Saddle River, NJ, 2004.
- [17] A. Gray, *Modern Differential Geometry of Curves and Surfaces*, Boca Raton, FL: CRC Press, c1993.
- [18] Z. Xu and B. M. Sadler, "Time delay estimation bounds in convolutive random channels," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Performance Limits of Ultra-Wideband Systems*, vol. 1, no. 3, October 2007 (in press).
- [19] P. Kontkanen, P. Myllymaki, T. Roos, H. Tirri, K. Valtonen, and H. Wetzig, "Topics in probabilistic location estimation in wireless networks," *Proc. 15th IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'04)*, Barcelona, Spain, 2004.
- [20] B. M. Sadler, R. J. Kozick, and L. Tong, "Multi-modal sensor localization using a mobile access point," *Proc. ICASSP'05*, pp.753-756, Mar. 2005.