*Article*

# Fixed Point Iteration Based Algorithm for Asynchronous TOA-Based Source Localization

**Yanbin Zou** [1,*] **, Jingna Fan** [1] **, Liehu Wu** [1] **and Huaping Liu** [2]

1  Department of Electronic and Information Engineering, Shantou University, Shantou 515063, China
2  School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331, USA
*  Correspondence: ybzou@stu.edu.cn

**Abstract:** This paper investigates the problem of source localization using signal time-of-arrival (TOA) measurements in the presence of unknown start transmission time. Most state-of-art methods are based on convex relaxation technologies, which possess global solution for the relaxed optimization problem. However, computational complexity of the convex optimization–based algorithm is usually large, and need CVX toolbox to solve it. Although the two stage weighted least squares (2SWLS) algorithm has very low computational complexity, its estimate performance is susceptible to sensor geometry and threshold phenomenon. A new algorithm that is directly derived from maximum likelihood estimator (MLE) is developed. The newly proposed algorithm is named as fixed point iteration (FPI); it only involves simple calculations, such as addition, multiplication, division, and square-root. Unlike state-of-the-art methods, there is no matrix inversion operation and can avoid the unstable performance incurred by singular matrix. The FPI algorithm can be easily extended to the scenario with sensor position errors. Finally, simulation results demonstrate that the proposed algorithm reaches a good balance between computational complexity and localization accuracy.

**Keywords:** fixed point iteration (FPI); sensor position errors; source localization; time-of-arrival (TOA)

## 1. Introduction

Source localization has a wide range of applications, such as navigation, monitoring, tracking, rescue and so on. There are lots of methods can be used to determine the source position, where time-of-arrival (TOA), time-difference-of-arrival (TDOA), time-delay (TD), received signal strength (RSS), angle-of-arrival (AOA) and their combinations are commonly positioning schemes [1–8].

A TOA-based localization system requires that the source and sensors be accurately time synchronized. However, it is hard to maintain the accurate synchronization when the source is non-cooperative [9]. The pseudo ranges are corrupted with an unknown distance when the start transmission time is unknown in the asynchronous networks. As a result, study on the asynchronous TOA (ATOA)-based localization problem is very important in the practice.

In the work reported in [10–18], the ATOA-based localization problem can be addressed by two ways: (a) convex optimization methods, including semidefinite programming (SDP) and second-order-cone programming (SOCP) [10–17]; (b) two stage weighted least squares (2SWLS) method [18].

Convex optimization methods have the merit of global solution, but their computational complexity is usually huge and need a CVX toolbox to solve the optimization problem [19]. For example, Xu et al. [10] developed two SDP-based algorithms corresponding to two step least squares (2LS) and min–max (MMA) criterion, respectively. Then in [14], Zou et al. showed that the original 2LS-based SDP algorithm cannot provide a good solution because the weighted matrix **G** is singular. In order to improve the tightness of

the original SDP algorithm, second-order-cone constraints and penalty term are jointly added to the original SDP algorithm. However, the algorithm proposed in [14] needs to calculate SDP problem multiple times and results in great computational burden. Recently, in [17], Ma et al. formulated an SDP algorithm to address the constraint existing in the WLS problem, which was formulated by Huang et al. in [18].

In [18], Huang et al. proposed a 2SWLS method to jointly estimate the start transmission time and the location of source. The 2SWLS method has very low computational complexity, but its localization accuracy is highly susceptible to sensor geometry when it is uniform circular array and the source is close to the array center. Besides this, the 2SWLS method presents a threshold phenomenon.

Another challenging issue for ATOA localization is sensor position errors, which could drastically degrade the accuracy [20]. Several works have addressed this problem [10,13–16], which have shown that using convex optimization could effectively address the sensor position errors. However, their computational complexity is also large.

In this paper, we develop a fixed point iteration (FPI) algorithm for the ATOA-based localization problem. The FPI algorithm is entirely and directly developed from maximum likelihood estimator (MLE). The FPI solution satisfies the equation that gradient is equal to zero. It is different from state-of-the-art methods. First, unlike weighted least squares (WLS) based algorithms, there is no approximation in its derivation. Second, unlike semidefinite programming (SDP) based algorithms, there is also no convex relaxation in its derivation. Third, unlike Gauss–Newton or Quasi-Newton based algorithms [21,22], there is no first-order Taylor series expansion in its derivation. Besides this, the proposed FPI algorithm can be easily tailored to the scenario with sensor position errors.

The rest of this paper is organized as follows. Section 2 develops an iterative algorithm when sensors positions are accurate. The algorithm is then extended to the scenario with non-accurate sensors positions in Section 3. Simulation results are presented in Section 4 to compare the performances of the proposed algorithm, state-of-art algorithms and Cramér-Rao lower bound (CRLB).

The following notations are used throughout the paper. Bold lowercase and uppercase letters denote vectors and matrices, respectively; $\mathbf{A}(:,i)$ denotes the $i$th column of matrix $\mathbf{A}$, and $\mathbf{A}(i,:)$ denotes the $i$th row of matrix $\mathbf{A}$; $\mathrm{tr}(\mathbf{A})$ is the trace of $\mathbf{A}$; $\mathbf{I}_M$ is the $M \times M$ identity matrix, $\mathbf{1}_M$ is the column vector of $M$ ones, and $\mathbf{0}_M$ is the column vector of $M$ zeros; $\|\cdot\|$ is the $l_2$ norm; $\otimes$ is the Kroneker product of two matrices; and $\mathbf{E}[\cdot]$ is the expectation.

## 2. Localization with Accurate Sensor Positions

Consider a network with $M$ sensors and one source whose location $\mathbf{u} \in \mathbb{R}^{m \times 1}$ is unknown and to be estimated ($m = 2$ or $3$). The $i$th sensor position $\mathbf{s}_i$ is accurate known.

Under line-of-sight condition, the range-of-arrival (ROA) measurements between sensor $i$ and the source are expressed as

$$r_i = t_0 c + \|\mathbf{u} - \mathbf{s}_i\| + n_i, \ i = 1, \cdots, M \tag{1}$$

where $c$ is the known signal propagation speed, and $t_0$ is unknown start transmission time. In the next, $\mathbf{t}_0 c$ will be replaced by $d_0$, and Equation (1) can be written as

$$r_i = d_0 + \|\mathbf{u} - \mathbf{s}_i\| + n_i \tag{2}$$

where $n_i, i = 1, \cdots, M$, are the ROA measurement noise, which are modeled as zero-mean Gaussian random variables with covariance matrix $\mathbf{Q} = \mathrm{diag}([\sigma_1^2, \ldots, \sigma_M^2])$. The maximum likelihood estimator (MLE) from Equation (2) is formulated as

$$\min_{\mathbf{u}, d_0} \sum_{i=1}^{M} \frac{\left(r_i - d_0 - \|\mathbf{u} - \mathbf{s}_i\|\right)^2}{\sigma_i^2}. \tag{3}$$

The above formulation can also be written as

$$\min_{\mathbf{u},d_0,\mathbf{d}}(\mathbf{r} - d_0\mathbf{1}_M - \mathbf{d})^T\mathbf{Q}^{-1}(\mathbf{r} - d_0\mathbf{1}_M - \mathbf{d}) \tag{4a}$$

$$\text{s.t. } d_i = \|\mathbf{u} - \mathbf{s}_i\|. \tag{4b}$$

where $\mathbf{d} = [d_1,\ldots,d_M]^T$. Let gradient of objective function in (4a) with respect to $d_0$ to zero

$$-2\mathbf{1}_M^T\mathbf{Q}^{-1}(\mathbf{r} - d_0\mathbf{1}_M - \mathbf{d}) = \mathbf{0} \tag{5}$$

and we can obtain

$$d_0 = \frac{\mathbf{1}_M^T\mathbf{Q}^{-1}(\mathbf{r} - \mathbf{d})}{\mathbf{1}_M^T\mathbf{Q}^{-1}\mathbf{1}_M}. \tag{6}$$

Putting above $d_0$ back to Equation (4), we obtain

$$\min_{\mathbf{u},\mathbf{d}}(\mathbf{r} - \mathbf{d})^T\mathbf{G}(\mathbf{r} - \mathbf{d}) \tag{7a}$$

$$\text{s.t. } d_i = \|\mathbf{u} - \mathbf{s}_i\| \tag{7b}$$

where

$$\mathbf{G} = \mathbf{I}_M - \frac{\mathbf{1}_M\mathbf{1}_M^T\mathbf{Q}^{-1}}{\mathbf{1}_M^T\mathbf{Q}^{-1}\mathbf{1}_M}. \tag{8}$$

Equation (7) is a non-convex optimization problem, and it can be resorted to convex relaxation method [14]. However, the convex optimization–based algorithm usually has very high computational complexity, and need CVX toolkit to solve it. Next, we develop an alternative solution for the above problem. First, Equation (7) can be expressed as the form of unconstrained optimization problem

$$\min_{\mathbf{u}}\sum_{i=1}^{M}\sum_{j=1}^{M}G(i,j)(r_i - \|\mathbf{u} - \mathbf{s}_i\|)(r_j - \|\mathbf{u} - \mathbf{s}_j\|). \tag{9}$$

The gradient of objective function in (9) with respect to $\mathbf{u}$ is

$$\mathbf{g} = -2\sum_{i=1}^{M}\sum_{j=1}^{M}G(i,j)(r_i - \|\mathbf{u} - \mathbf{s}_i\|)\frac{\mathbf{u} - \mathbf{s}_j}{\|\mathbf{u} - \mathbf{s}_j\|}. \tag{10}$$

The necessary condition for the optimization problem in (9) is

$$\mathbf{g} = \mathbf{0} \tag{11}$$

i.e.,

$$\sum_{i=1}^{M}G(i,i)\left(r_i\frac{\mathbf{u} - \mathbf{s}_i}{\|\mathbf{u} - \mathbf{s}_i\|} - (\mathbf{u} - \mathbf{s}_i)\right) + \sum_{i=1}^{M}\sum_{j\neq i}^{M}G(i,j)(r_i - \|\mathbf{u} - \mathbf{s}_i\|)\frac{\mathbf{u} - \mathbf{s}_j}{\|\mathbf{u} - \mathbf{s}_j\|} = \mathbf{0}. \tag{12}$$

Next, move the linear function of $\mathbf{u}$ to one side of the equation, and the constant term and nonlinear function of $\mathbf{u}$ to another side of the equation

$$\sum_{i=1}^{M}G(i,i)\mathbf{u} = \sum_{i=1}^{M}G(i,i)\left(r_i\frac{\mathbf{u} - \mathbf{s}_i}{\|\mathbf{u} - \mathbf{s}_i\|} + \mathbf{s}_i\right) + \sum_{i=1}^{M}\sum_{j\neq i}^{M}G(i,j)(r_i - \|\mathbf{u} - \mathbf{s}_i\|)\frac{\mathbf{u} - \mathbf{s}_j}{\|\mathbf{u} - \mathbf{s}_j\|}. \tag{13}$$

Then

$$\mathbf{u} = \frac{1}{\text{tr}(\mathbf{G})} \left( \sum_{i=1}^{M} G(i,i) \left( r_i \frac{\mathbf{u} - \mathbf{s}_i}{\|\mathbf{u} - \mathbf{s}_i\|} + \mathbf{s}_i \right) + \sum_{i=1}^{M} \sum_{j \neq i}^{M} G(i,j)(r_i - \|\mathbf{u} - \mathbf{s}_i\|) \frac{\mathbf{u} - \mathbf{s}_j}{\|\mathbf{u} - \mathbf{s}_j\|} \right). \quad (14)$$

Moreover,

$$\text{tr}(\mathbf{G}) = \text{tr}\left(\mathbf{I}_M - \frac{\mathbf{1}_M \mathbf{1}_M^T \mathbf{Q}^{-1}}{\mathbf{1}_M^T \mathbf{Q}^{-1} \mathbf{1}_M}\right) = M - 1. \quad (15)$$

As a result, the FPI computation of $\mathbf{u}$ is

$$\mathbf{u}_k = f(\mathbf{u}_{k-1}) = \frac{1}{M-1} \left( \sum_{i=1}^{M} G(i,i) \left( r_i \frac{\mathbf{u}_{k-1} - \mathbf{s}_i}{\|\mathbf{u}_{k-1} - \mathbf{s}_i\|} + \mathbf{s}_i \right) + \sum_{i=1}^{M} \sum_{j \neq i}^{M} G(i,j)(r_i - \|\mathbf{u}_{k-1} - \mathbf{s}_i\|) \frac{\mathbf{u}_{k-1} - \mathbf{s}_j}{\|\mathbf{u}_{k-1} - \mathbf{s}_j\|} \right). \quad (16)$$

It should be noted that the above derivations are based on the necessary condition. In other words, the result from above FPI may be a stationary point, i.e., a local maximum, a local minimum point or a saddle point. In order to obtain the global minimum point, we adopt the following steps. First is to use multiple initial values to calculate FPI in Equation (16). Without loss of generality, the set of initial values can be set as follows (from (16), it can be seen that the initial values should be not equal to the location of sensors. The main principle for the setting of initial values is that it includes points located inside and outside the convex hull of sensors).

$$\mathbf{U}_0 = [\frac{1}{M} \sum_{i=1}^{M} \mathbf{s}_i, 2\mathbf{s}_1 - \frac{1}{M} \sum_{i=1}^{M} \mathbf{s}_i, \dots, 2\mathbf{s}_M - \frac{1}{M} \sum_{i=1}^{M} \mathbf{s}_i]. \quad (17)$$

Then, we calculate the corresponding cost function for each initial value

$$c_f(n) = (\mathbf{r} - \mathbf{d}_n^*)^T \mathbf{G} (\mathbf{r} - \mathbf{d}_n^*), n = 1, \dots, M+1 \quad (18)$$

where $\mathbf{d}_n^* = [\|\mathbf{u}_n^* - \mathbf{s}_1\|, \dots, \|\mathbf{u}_n^* - \mathbf{s}_M\|]^T$. Finally, we choose the result that minimizes the cost function.

Besides this, in order to reduce the calculation amount, we take the following measure. If $c_f(n) < \mu$ for a certain $n$, then we stop calculating the iterations for the rest initial values. Next, we show how to determine the threshold $\mu$. Reviewing (3), we can obtain

$$\mathbf{E}[\sum_{i=1}^{M} \frac{\left( r_i - d_0 - \|\mathbf{u} - \mathbf{s}_i\| \right)^2}{\sigma_i^2}] = \mathbf{E}[\sum_{i=1}^{M} \frac{n_i^2}{\sigma_i^2}] = M \quad (19)$$

and

$$\mathbf{E}[(\mathbf{r} - \mathbf{d})^T \mathbf{G} (\mathbf{r} - \mathbf{d})] = M \quad (20)$$

for true source location $\mathbf{u}$. As a result, the threshold is set to $\mu = M$.

The above proposed algorithm procedures are summarized in Algorithm 1.

---

**Algorithm 1:** Localization with scenario 1: accurate sensor position.

**Data:** $r_i$, $\mathbf{s}_i$, $\mathbf{G}$, $\mathbf{U}_0$, $K_{max}$, $\epsilon$;
**Result:** $\mathbf{u}^*$

```
 1  for n = 1 : M + 1 do
 2  │   u₀ ← U₀(:, n);
 3  │   u* ← u₀;
 4  │   k ← 1;
 5  │   while k ≤ K_max do
 6  │   │   u_k ← compute (16);
 7  │   │   if ‖u* − u_k‖ ≥ ε then
 8  │   │   │   u* ← u_k;
 9  │   │   else
10  │   │   │   break;
11  │   │   end
12  │   │   k ← k + 1;
13  │   end
14  │   u*_n ← u*;
15  │   c_f(n) ← compute (18);
16  │   if c_f(n) < M then
17  │   │   u* ← u*_n ;
18  │   │   break;
19  │   end
20  end
21  u* ← u*_n that minimizes c_f(n);
```

## 3. Localization with Non-Accurate Sensor Positions

In this section, we will extend the FPI localization algorithm to the scenario with non-accurate sensor positions. First, the $i$th true but unknown sensor location is $\mathbf{s}_i^0$. The available sensor location can be denoted as

$$\mathbf{s}_i = \mathbf{s}_i^0 + \boldsymbol{\beta}_i, \ i = 1, \cdots, M \tag{21}$$

where $\boldsymbol{\beta}_i$ is a zero-mean white Gaussian vector with covariance matrix $\delta_i^2 \mathbf{I}_m$ [23]. The MLE is formulated as

$$\min_{\mathbf{u}, d_0, \mathbf{s}_i^0} \sum_{i=1}^{M} \frac{\left(r_i - d_0 - \left\|\mathbf{u} - \mathbf{s}_i^0\right\|\right)^2}{\sigma_i^2} + \sum_{i=1}^{M} \frac{\left\|\mathbf{s}_i - \mathbf{s}_i^0\right\|}{\delta_i^2}. \tag{22}$$

Putting (6) into (22), this results in

$$\min_{\mathbf{u}, \mathbf{d}, \mathbf{s}_i^0} (\mathbf{r} - \mathbf{d})^T \mathbf{G} (\mathbf{r} - \mathbf{d}) + \sum_{i=1}^{M} \frac{\left\|\mathbf{s}_i - \mathbf{s}_i^0\right\|^2}{\delta_i^2} \tag{23a}$$

$$\text{s.t. } d_i = \left\|\mathbf{u} - \mathbf{s}_i^0\right\|. \tag{23b}$$

Let $\mathbf{x} = [\mathbf{u}; \mathbf{s}_1^0; \ldots; \mathbf{s}_M^0]$, $\mathbf{s} = [\mathbf{s}_1; \ldots; \mathbf{s}_M]$, $\mathbf{W}_s = \mathbf{Q}_s \otimes \mathbf{I}_m$, $\mathbf{Q}_s = \text{diag}([\frac{1}{\delta_1^2}, \ldots, \frac{1}{\delta_M^2}])$, $\mathbf{P}_i = [\mathbf{I}_m, \mathbf{0}_{m,m(i-1)}, -\mathbf{I}_m, \mathbf{0}_{m,m(M-i)}]$, $\mathbf{B} = [\mathbf{0}_{mM,m}, \mathbf{I}_{mM}]$. Equation (23) can be written as

$$\min_{\mathbf{x}} \sum_{i=1}^{M} \sum_{j=1}^{M} G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}\|)(r_j - \|\mathbf{P}_j\mathbf{x}\|) + \sum_{i=1}^{mM} W_s(i,i)(s(i) - \mathbf{B}(i,:)\mathbf{x})^2. \tag{24}$$

The gradient of objective function in Equation (24) with respect to **x** is

$$\mathbf{g} = -2\sum_{i=1}^{M}\sum_{j=1}^{M}G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}\|)\frac{\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}}{\|\mathbf{P}_j\mathbf{x}\|} - 2\sum_{i=1}^{mM}W_s(i,i)\big(s(i) - \mathbf{B}(i,:)\mathbf{x}\big)\mathbf{B}(i,:)^T. \quad (25)$$

The necessary condition for the optimization problem in (24) is

$$\mathbf{g} = \mathbf{0}. \quad (26)$$

Equivalent to

$$\sum_{i=1}^{M}G(i,i)r_i\frac{\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}}{\|\mathbf{P}_i\mathbf{x}\|} - \sum_{i=1}^{M}G(i,i)\mathbf{P}_i^T\mathbf{P}_i\mathbf{x} + \sum_{i=1}^{M}\sum_{j\neq i}^{M}G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}\|)\frac{\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}}{\|\mathbf{P}_j\mathbf{x}\|} +$$

$$\sum_{i=1}^{mM}W_s(i,i)s(i)\mathbf{B}(i,:)^T - \sum_{i=1}^{mM}W_s(i,i)\mathbf{B}(i,:)^T\mathbf{B}(i,:)\mathbf{x} = \mathbf{0} \quad (27)$$

i.e.,

$$\Big(\sum_{i=1}^{M}G(i,i)\mathbf{P}_i^T\mathbf{P}_i + \sum_{i=1}^{mM}W_s(i,i)\mathbf{B}(i,:)^T\mathbf{B}(i,:)\Big)\mathbf{x} = \sum_{i=1}^{M}G(i,i)r_i\frac{\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}}{\|\mathbf{P}_i\mathbf{x}\|} + \sum_{i=1}^{M}\sum_{j\neq i}^{M}G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}\|)\frac{\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}}{\|\mathbf{P}_j\mathbf{x}\|}$$

$$+ \sum_{i=1}^{mM}W_s(i,i)s(i)\mathbf{B}(i,:)^T. \quad (28)$$

Finally,

$$\mathbf{x} = \Big(\sum_{i=1}^{M}G(i,i)\mathbf{P}_i^T\mathbf{P}_i + \sum_{i=1}^{mM}W_s(i,i)\mathbf{B}(i,:)^T\mathbf{B}(i,:)\Big)^{-1}.$$

$$[\sum_{i=1}^{M}G(i,i)r_i\frac{\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}}{\|\mathbf{P}_i\mathbf{x}\|} + \sum_{i=1}^{M}\sum_{j\neq i}^{M}G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}\|)\frac{\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}}{\|\mathbf{P}_j\mathbf{x}\|} + \sum_{i=1}^{mM}W_s(i,i)s(i)\mathbf{B}(i,:)^T]. \quad (29)$$

The FPI computation of **x** is

$$\mathbf{x}_k = \Big(\sum_{i=1}^{M}G(i,i)\mathbf{P}_i^T\mathbf{P}_i + \sum_{i=1}^{mM}W_s(i,i)\mathbf{B}(i,:)^T\mathbf{B}(i,:)\Big)^{-1}.$$

$$[\sum_{i=1}^{M}G(i,i)r_i\frac{\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}_{k-1}}{\|\mathbf{P}_i\mathbf{x}_{k-1}\|} + \sum_{i=1}^{mM}W_s(i,i)s(i)\mathbf{B}(i,:)^T + \sum_{i=1}^{M}\sum_{j\neq i}^{M}G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}_{k-1}\|)\frac{\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}_{k-1}}{\|\mathbf{P}_j\mathbf{x}_{k-1}\|}]. \quad (30)$$

The proposed algorithm for scenario 2 is shown in Algorithm 2. It includes two stages. The first stage is to carry out Algorithm 1 to obtain an initial guess of source position by using the non-accurate sensor positions. The second stage is to update **x** using Equation (30) and taking the result from stage one as initial value.

---

**Algorithm 2:** Localization with scenario 2: non-accurate sensor positions.

**Data:** $r_i$, $\mathbf{s}_i$, $\mathbf{s}$, $\mathbf{W}_s$, $\mathbf{G}$, $\mathbf{P}_i$, $\mathbf{B}$, $\mathbf{U}_0$, $K_{max}$, $\epsilon$;

**Result:** $\mathbf{u}^*$

1 **first stage**: deeming $\mathbf{s}_i$ are accurate to obtain an initial value of $\mathbf{u}$;

2 **for** $n = 1 : M + 1$ **do**

3    $\mathbf{u}_0 \leftarrow \mathbf{U}_0(:, n)$;

4    $\mathbf{u}^* \leftarrow \mathbf{u}_0$;

5    $k \leftarrow 1$;

6    **while** $k \leq K_{max}$ **do**

7       $\mathbf{u}_k \leftarrow$ compute (16);

8       **if** $\|\mathbf{u}^* - \mathbf{u}_k\| \geq \epsilon$ **then**

9          $\mathbf{u}^* \leftarrow \mathbf{u}_k$;

10       **else**

11          break;

12       **end**

13       $k \leftarrow k + 1$;

14    **end**

15    $c_f(n) \leftarrow$ compute (18);

16    $\mathbf{u}_m^* \leftarrow \mathbf{u}^*$;

17    **if** $c_f(n) < M$ **then**

18       break;

19    **end**

20 **end**

21 $\mathbf{u}^* \leftarrow \mathbf{u}_n^*$ that minimizes $c_f(n)$;

22 **second stage:** update $\mathbf{x}$;

23 $\mathbf{x}_0 \leftarrow [\mathbf{u}^*; \mathbf{s}]$;

24 $\mathbf{x}^* \leftarrow \mathbf{x}_0$;

25 **while** $k \leq K_{max}$ **do**

26    $\mathbf{x}_k \leftarrow$ compute (30);

27    **if** $\|\mathbf{x}^* - \mathbf{x}_k\| \geq \epsilon$ **then**

28       $\mathbf{x}^* \leftarrow \mathbf{x}_k$;

29    **else**

30       break;

31    **end**

32    $k \leftarrow k + 1$;

33 **end**

---

## 4. Simulation Results

In this section, two scenarios with accurate and non-accurate sensor positions are respectively considered to demonstrate the performance of two proposed algorithms.

In the first scenario, we compare several algorithms: the proposed Algorithm 1 (with one middle point as initial value labeled as 'Proposed-Mid', with multiple initial values labeled as 'Proposed-Multi'), the Gauss–Newton method (in the simulation, the result of Gauss–Newton method may be NaN due to the matrix is singular in the calculation of matrix inversion. In order to avoid this case, it is set that, if the distance between the updated estimate and the initial value is large than 1000, then the iteration will be stopped and output the initial value as its result) with one middle point as initial value labeled as 'Gauss–Newton-Mid', with multiple initial values labeled as 'Gauss–Newton-Multi'; the derivation for the Gauss–Newton method is shown in Appendix A), 2SWLS [18], SDP-Zou [14], SDP–Ma [17] and CRLB.

In the second scenario, we compare the proposed Algorithm 2 (labeled as 'Proposed') with the Gauss–Newton method (labeled as 'Gauss–Newton'), SDP–Zou [14] and CRLB when sensor positions are non-accurate. The derivation of the Gauss–Newton method with position errors for the problem of (24) is given in Appendix B.

The setting of parameters are the same as in [14]: a network with four sensors, and one source is simulated. The positions of the sensors are $[-10, -10]^T m$, $[-10, 10]^T m$, $[10, -10]^T m$, $[10, 10]^T m$, and the location of source is randomly chosen from a square $[-15, 15]m \times [-15, 15]m$. The start transmission time $t_0$ is drawn from uniform distribution $U[10, 40]ns$. Sensors' position covariance matrix is $\delta^2 \mathbf{I}_2$. The range measurement covariance matrix is $\mathbf{Q} = \sigma^2 \mathbf{I}_M$. The initial set for Algorithm 1 is $\mathbf{U}_0 = [0, -20, -20, 20, 20; 0, -20, 20, -20, 20]$. Five penalty factors $\eta_1, \eta_2, \eta_3, \eta_4, \eta_5$ equal to $10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0$ are used for the SDP–Zou algorithm. The SDP–Zou and SDP–Ma algorithms are implemented by CVX toolbox using SeDuMi as a solver and with best precision [24]. The Gauss–Newton methods and the proposed FPI algorithm have same initials. Root mean square errors (RMSEs) are drawn from 4000 Monte Carlo realizations in the following simulations.

From Figure 1, we have following observations: (1) only the proposed algorithm with multiple initial values and the SDP–Zou algorithm can reach CRLB; (2) the 'Proposed-Multi' algorithm is superior to the 'Proposed-Mid' algorithm, which validates the existence of other stationary point except for the global minimum point in the problem of MLE; (3) the Gauss–Newton methods have inferior performance due to its first-order approximation in the derivations; (4) the 2SWLS algorithm has the worst performance even with small noise condition, which is due to the sensor geometry is uniform circular array and the source is close to the array center occasionally [25]; (5) the SDP–Ma algorithm has poor performance, which is due to the number of sensors, namely, 4, which results in rank-1 solution being hard to obtain [17].
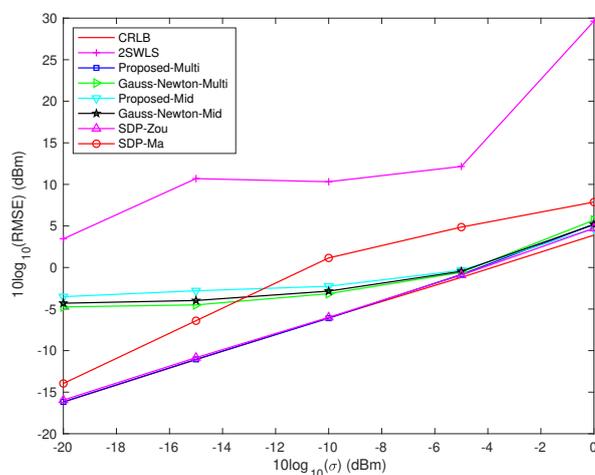


**Figure 1.** RMSE vs. $\sigma$ in scenario 1.

Besides this, the average running times of different algorithms are given in Table 1. It can be seen that: (1) the 2SWLS algorithm has the lowest computation time, because it only involves three times computation of WLS solution; (2) both SDP–Zou and SDP–Ma algorithms are time-consuming to obtain the solutions by CVX toolbox; moreover, the SDP–Zou algorithm needs to solve the SDP problem five times; (3) the average running time of 'Proposed-Multi' is large than but less than twice of the 'Proposed-Mid', which validates that the stop criterion in (20) can help to reduce the calculation amount. In the simulation, we found that if $\mu = 2M$ the calculation amount can be further reduced; (4) the Gauss–Newton algorithms are faster than the proposed FPI algorithms.

**Table 1.** The average running time [ms] of the considered algorithms with the two scenarios, in which a star denotes that the algorithm is not fitting for a certain scenario. CPU: i7-8700 3.2 GHz.

| Scenarios Algorithms | 1 | 2 |
|---|---|---|
| 2SWLS | 0.19 | * |
| Proposed-Multi | 8.6 | * |
| Gauss–Newton-Multi | 0.94 | * |
| Proposed-Mid | 5.3 | * |
| Gauss–Newton-Mid | 0.67 | * |
| SDP–Ma | 412 | * |
| SDP–Zou | 1773 | 1861 |
| Proposed | * | 28.6 |
| Gauss–Newton | * | 2.6 |

Figures 2 and 3, respectively, evaluate the source and sensors estimate performances of different algorithms versus $\delta$ when the locations of sensors are non-accurate. From these two figures, it is observed that the performance of the proposed algorithm is superior to the other two algorithms. Besides this, in Table 1, it can be seen that the average running time of the proposed algorithm is much less than the SDP–Zou algorithm, but greater than the Gauss–Newton algorithm.
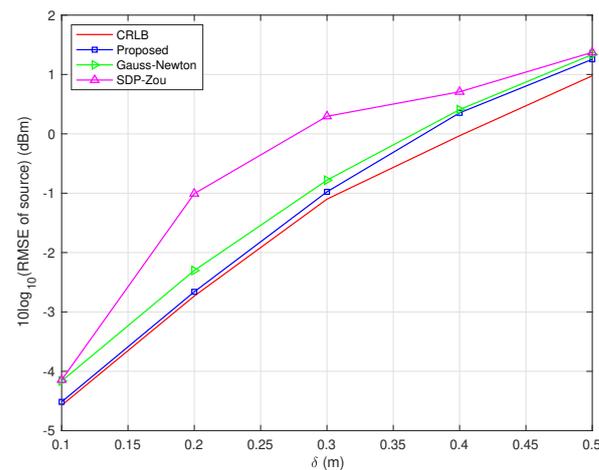


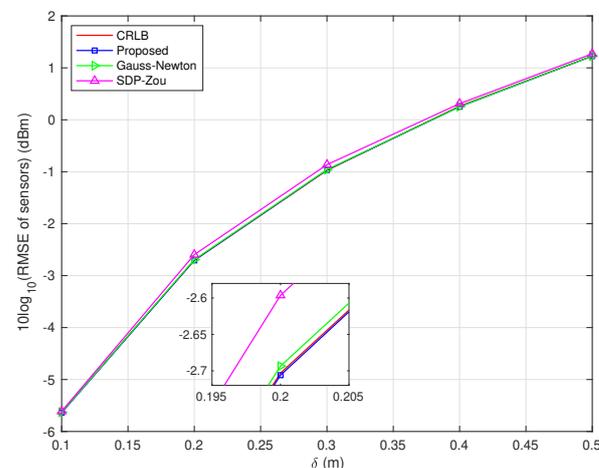**Figure 2.** RMSE of source vs. $\delta$ with $\sigma = 0.1$ m in scenario 2.



**Figure 3.** RMSE of sensors vs. $\delta$ with $\sigma = 0.1$ m in scenario 2.

Finally, from the above results and analyses, we can obtain a conclusion: The proposed FPI algorithms reach a good balance between estimation accuracy and calculation amount.

## 5. Conclusions

Two FPI algorithms for ATOA-based localization problem are respectively developed corresponding to the scenarios with accurate and non-accurate sensor positions. First, the problem of MLE for the unknown start transmission time and source position is formulated. Next, the gradient of objective function in MLE is derived, then we simplify the equation with gradient equal to zero. Finally, we extract the linear variable to one side of the equation and obtain the FPI algorithm. The simulation results validate the performances of the two proposed algorithms in terms of accuracy and computation.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

**Sample Availability:** The codes of simulations are available from the authors via emails ybzou@stu.edu.cn and huaping.liu@oregonstate.edu.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| TOA | Time-of-arrival |
| 2SWLS | Two-step weighted least squares |
| MLE | Maximum likelihood |
| PFI | Fixed point iteration |
| TDOA | Time-difference-of-arrival |
| TD | Time delay |
| RSS | Received signal strength |
| AOA | angle-of-arrival |
| ATOA | Asynchronous TOA |
| SDP | Semidefinite programming |
| SOCP | Second-order cone programming |
| WLS | Weighted least squares |
| CRLB | Cramer-Rao lower bound |
| RMSE | Root mean-square error |

## Appendix A. Gauss–Newton Method for Scenario 1

In this appendix, the Gauss–Newton method for scenario 1 with accurate sensor positions is derived. Rewriting the MLE problem in (9)

$$\min_{\mathbf{u}} \sum_{i=1}^{M} \sum_{j=1}^{M} G(i,j)(r_i - \|\mathbf{u} - \mathbf{s}_i\|)(r_j - \|\mathbf{u} - \mathbf{s}_j\|) \tag{A1}$$

The first-order Taylor series expansion of $\|\mathbf{u} - \mathbf{s}_i\|$ is

$$\|\mathbf{u} - \mathbf{s}_i\| \approx \|\mathbf{u}_0 - \mathbf{s}_i\| + \frac{(\mathbf{u}_0 - \mathbf{s}_i)^T (\mathbf{u} - \mathbf{u}_0)}{\|\mathbf{u}_0 - \mathbf{s}_i\|} \tag{A2}$$

Then, taking (A2) into (A1) results

$$\min_{\mathbf{u}} \sum_{i=1}^{M} \sum_{j=1}^{M} G(i,j) \Big( r_i - \|\mathbf{u}_0 - \mathbf{s}_i\| - \frac{(\mathbf{u}_0 - \mathbf{s}_i)^T(\mathbf{u} - \mathbf{u}_0)}{\|\mathbf{u}_0 - \mathbf{s}_i\|} \Big) \Big( r_j - \|\mathbf{u}_0 - \mathbf{s}_j\| - \frac{(\mathbf{u}_0 - \mathbf{s}_j)^T(\mathbf{u} - \mathbf{u}_0)}{\|\mathbf{u}_0 - \mathbf{s}_j\|} \Big) \tag{A3}$$

Let the gradient of objective function in (A3) with $\mathbf{u}$ to zero

$$\sum_{i=1}^{M} \sum_{j=1}^{M} -2G(i,j) \Big( r_i - \|\mathbf{u}_0 - \mathbf{s}_i\| - \frac{(\mathbf{u}_0 - \mathbf{s}_i)^T(\mathbf{u} - \mathbf{u}_0)}{\|\mathbf{u}_0 - \mathbf{s}_i\|} \Big) \frac{(\mathbf{u}_0 - \mathbf{s}_j)}{\|\mathbf{u}_0 - \mathbf{s}_j\|} = \mathbf{0} \tag{A4}$$

Set

$$\mathbf{h}_0 = [r_1 - \|\mathbf{u}_0 - \mathbf{s}_1\|, \dots, r_M - \|\mathbf{u}_0 - \mathbf{s}_M\|]^T \tag{A5a}$$

$$\mathbf{H}_0 = [\frac{(\mathbf{u}_0 - \mathbf{s}_1)}{\|\mathbf{u}_0 - \mathbf{s}_1\|}, \dots, \frac{(\mathbf{u}_0 - \mathbf{s}_M)}{\|\mathbf{u}_0 - \mathbf{s}_M\|}] \tag{A5b}$$

Then (A4) can be recast as

$$\mathbf{H}_0 \mathbf{G} \mathbf{h}_0 = \mathbf{H}_0 \mathbf{G} \mathbf{H}_0^T (\mathbf{u} - \mathbf{u}_0) \tag{A6}$$

Finally, the updating of $\mathbf{u}$ is

$$\mathbf{u} = \mathbf{u}_0 + (\mathbf{H}_0 \mathbf{G} \mathbf{H}_0^T)^{-1} \mathbf{H}_0 \mathbf{G} \mathbf{h}_0 \tag{A7}$$

## Appendix B. Gauss–Newton Method for Scenario 2

In this appendix, the Gauss–Newton method for scenario 2 with non-accurate sensor positions is derived. Rewriting the MLE problem in (24)

$$\min_{\mathbf{x}} \sum_{i=1}^{M} \sum_{j=1}^{M} G(i,j)(r_i - \|\mathbf{P}_i\mathbf{x}\|)(r_j - \|\mathbf{P}_j\mathbf{x}\|) + \sum_{i=1}^{mM} W_s(i,i)(s(i) - \mathbf{B}(i,:)\mathbf{x})^2 \tag{A8}$$

The first-order Taylor series expansion of $\|\mathbf{P}_i\mathbf{x}\|$ is

$$\|\mathbf{P}_i\mathbf{x}\| \approx \|\mathbf{P}_i\mathbf{x}_0\| + \frac{(\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0)}{\|\mathbf{P}_i\mathbf{x}_0\|} \tag{A9}$$

Then, taking (A9) into (A8) results

$$\min_{\mathbf{u}} \sum_{i=1}^{M} \sum_{j=1}^{M} G(i,j) \Big( r_i - \|\mathbf{P}_i\mathbf{x}_0\| - \frac{(\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0)}{\|\mathbf{P}_i\mathbf{x}_0\|} \Big) \Big( r_j - \|\mathbf{P}_j\mathbf{x}_0\| - \frac{(\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0)}{\|\mathbf{P}_j\mathbf{x}_0\|} \Big) +$$
$$\sum_{i=1}^{mM} W_s(i,i)(s(i) - \mathbf{B}(i,:)\mathbf{x})^2 \tag{A10}$$

Let the gradient of objective function in (A10) with $\mathbf{x}$ to zero

$$\sum_{i=1}^{M} \sum_{j=1}^{M} -2G(i,j) \Big( r_i - \|\mathbf{P}_i\mathbf{x}_0\| - \frac{(\mathbf{P}_i^T\mathbf{P}_i\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0)}{\|\mathbf{P}_i\mathbf{x}_0\|} \Big) \frac{\mathbf{P}_j^T\mathbf{P}_j\mathbf{x}_0}{\|\mathbf{P}_j\mathbf{x}_0\|} - \sum_{i=1}^{mM} 2W_s(i,i)(s(i) - \mathbf{B}(i,:)\mathbf{x})\mathbf{B}(i,:)^T = \mathbf{0} \tag{A11}$$

Set

$$\mathbf{h}_0 = [r_1 - \|\mathbf{P}_1\mathbf{x}_0\|, \dots, r_M - \|\mathbf{P}_M\mathbf{x}_0\|]^T \tag{A12a}$$

$$\mathbf{H}_0 = [\frac{\mathbf{P}_1^T\mathbf{P}_1\mathbf{x}_0}{\|\mathbf{P}_1\mathbf{x}_0\|}, \dots, \frac{\mathbf{P}_M^T\mathbf{P}_M\mathbf{x}_0}{\|\mathbf{P}_M\mathbf{x}_0\|}] \tag{A12b}$$

Then (A11) can be recast as

$$\mathbf{H}_0\mathbf{G}\left(\mathbf{h}_0 - \mathbf{H}_0^T(\mathbf{x} - \mathbf{x}_0)\right) + \mathbf{B}^T\mathbf{W}_s(\mathbf{s} - \mathbf{B}\mathbf{x}) = \mathbf{0} \qquad (A13)$$

Finally, the updating of $\mathbf{x}$ is

$$\mathbf{x} = \left(\mathbf{H}_0\mathbf{G}\mathbf{H}_0^T + \mathbf{B}^T\mathbf{W}_s\mathbf{B}\right)^{-1}\left(\mathbf{H}_0\mathbf{G}(\mathbf{h}_0 + \mathbf{H}_0^T\mathbf{x}_0) + \mathbf{B}^T\mathbf{W}_s\mathbf{s}\right) \qquad (A14)$$

## References

1. Zafari, F.; Gkelias, A.; Leung, K. A survey of indoor localization systems and technologies. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2568–2599. [CrossRef]
2. Zou, Y.; Liu, H. A simple and efficient iterative method for TOA localization. In Proceedings of the IEEE International Conference of Acoustics, Speech, and Signal Processing, Barcelona, Spain, 4–8 May 2020; pp. 4881–4884.
3. Liang, J.; Chen, Y.; So, H.; Yang, J. Circular/hyperbolic/elliptic localization via euclidean norm elimination. *Signal Process.* **2018**, *148*, 102–113. [CrossRef]
4. Zou, Y.; Liu, H. RSS-based target localization with unknown model parameters and sensor position errors. *IEEE Trans. Veh. Technol.* **2021**, *70*, 6969–6982. [CrossRef]
5. Zhang, Y.; Ho, K. Multistatic localization in the absence of transmitter position. *IEEE Trans. Signal Process.* **2019**, *67*, 4745–4760. [CrossRef]
6. Luo, J.; Shao, X.; Peng, D.; Zhang, X. A novel subspace approach for bearing-only target localization. *IEEE Sens. J.* **2019**, *19*, 8174–8182. [CrossRef]
7. Yin, J.; Wan, Q.; Yang, S.; Ho, K. A simple and accurate TDOA-AOA localization method using two stations. *IEEE Signal Process. Lett.* **2016**, *23*, 144–148. [CrossRef]
8. Amiri, R.; Behnia, F.; Zamani, H. Efficient 3-D positioning using time-delay and aoa measurements in MIMO radar systems. *IEEE Commun. Lett.* **2017**, *21*, 2614–2617. [CrossRef]
9. Zou, Y.; Liu, H.; Wan, Q. Joint synchronization and localization in wireless sensor networks using semidefinite programming. *IEEE Internet Things J.* **2018**, *5*, 199–205. [CrossRef]
10. Xu, E.; Ding, Z.; Dasgupta, S. Source localization in wireless sensor networks from signal time-of-arrival measurements. *IEEE Trans. Signal Process.* **2011**, *59*, 2887–2897. [CrossRef]
11. Vaghefi, R.; Buehrer, R. Asynchronous time-of-arrival-based source localization. In Proceedings of the IEEE International Conference of Acoustics, Speech, and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 4086–4090.
12. Wang, G.; Cai, S.; Li, Y.; Jin, M. Second-order cone relaxation for TOA-based source localization with unknown start transmission time. *IEEE Trans. Veh. Technol.* **2014**, *63*, 2973–2977. [CrossRef]
13. Mekonnen, Z.; Wittneben, A. Robust TOA based localization for wireless sensor networks with anchor position uncertainties. In Proceedings of the IEEE International Symposium on Personal, Indoor, Mobile Radio Communication, Washington, DC, USA, 2–5 September 2014; pp. 2029–2033.
14. Zou, Y.; Wan, Q. Asynchronous time-of-arrival-based source localization with sensor position uncertainties. *IEEE Commun. Lett.* **2016**, *20*, 1860–1863. [CrossRef]
15. Yan, Y.; Yang, G.; Wang, H.; Shen, X. Semidefinite relaxation for source localization with quantized TOA measurements and transmission uncertainty in sensor networks. *IEEE Trans. Commun.* **2021**, *69*, 1201–1213. [CrossRef]
16. Yang, G.; Yan, Y.; Wang, H.; Shen, X. Improved robust TOA-based source localization with individual constraint of sensor location uncertainty. *Signal Process.* **2022**, *196*, 108504. [CrossRef]
17. Ma, X.; Hao, B.; Zhang, H.; Wan, P. Semidefinite relaxation for source localization by TOA in unsynchronized networks. *IEEE Signal Process. Lett.* **2022**, *29*, 622–626. [CrossRef]
18. Huang, J.; Xue, Y.; Yang, L. An efficient closed-form solution for joint synchronization and localization using TOA. *Future Gener. Comput. Syst.* **2013**, *29*, 776–781. [CrossRef]
19. CVX: Matlab Software for Disciplined Convex Programming, Version 1.21. 2010. Available online: http://cvxr.com/cvx (accessed on 31 August 2022).
20. Ho, K.; Lu, X.; Kovavisaruch, L. Source localization using TDOA and FDOA measurements in the presence of receiver location errors: Analysis and solution. *IEEE Trans. Signal Process.* **2007**, *55*, 684–696. [CrossRef]
21. Jia, T.; Ho, K.C.; Wang, H.; Shen, X. Localization of a moving object with sensors in motion by time delays and Doppler shifts. *IEEE Trans. Signal Process.* **2020**, *68*, 5824–5841. [CrossRef]
22. Gao, W.; Goldfarb, D. Quasi-Newton methods: superlinear convergence without line searches for self-concordant functions. *Optim. Methods Softw.* **2018**, *34*, 194–217. [CrossRef]
23. Ho, K.; Yang, L. On the use of a calibration emitter for source localization in the presence of sensor position uncertainty. *IEEE Trans. Signal Process.* **2008**, *56*, 5758–5772. [CrossRef]

24.    Sturm, J. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optim. Methods Softw.* **1999**, *11*, 625–653. [CrossRef]

25.    Lin, L.; So, H.; Chan, F.; Chan, Y.; Ho, K. A new constrained weighted least squares algorithm for TDOA-based localization. *Signal Process.* **2013**, *93*, 2872–2878. [CrossRef]