Available online at www.sciencedirect.com

ScienceDirect

Journal of the Franklin Institute 360 (2023) 12150–12169

www.elsevier.com/locate/jfranklin

Lagrange programming neural network for robust passive elliptic positioning

Keyuan Hu[a], Wenxin Xiong[b,*], Yuwei Wang[c], Zhang-Lei Shi[d], Ge Cheng[e], Hing Cheung So[a], Zhi Wang[c]

[a] *Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China*
[b] *Department of Computer Science, University of Freiburg, Freiburg 79110, Germany*
[c] *State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 311121, China*
[d] *College of Science, China University of Petroleum (East China), Qingdao 266580, China*
[e] *Shenzhen Water Planning & Design Institute Co., Ltd., Shenzhen, China*

Received 10 June 2023; received in revised form 20 July 2023; accepted 17 September 2023
Available online 22 September 2023

Abstract

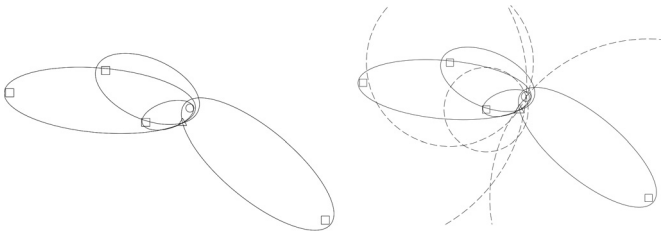
This contribution studies passive elliptic positioning (PEP) with unknown transmitter locations, a localization technique having great potential applicability ranging from underwater wireless sensor networks to intelligent transportation systems. Specifically, we aim to address the challenge of employing PEP in complex real-world environments where outliers may exist, by using the concept of robust statistics. To achieve such a goal, we replace the ℓ_2 loss in the traditional nonlinear least squares formulation by a differentiable cost function that possesses outlier-resistance. The neurodynamic approach of Lagrange programming neural network is then adopted to solve the resultant nonconvex statistically robustified PEP problem in a computationally efficient manner. Simulations and acoustic positioning experiments demonstrate the performance superiority of our proposal over its competitors.

1. Introduction

Target localization has been a critical research area for decades, especially in telecommunications [1], mobile communications [2], sonar [3], and wireless sensor networks (WSNs)

* Corresponding author.
E-mail address: xiongw@informatik.uni-freiburg.de (W. Xiong).

https://doi.org/10.1016/j.jfranklin.2023.09.038



(a) (b)

Fig. 1. Illustration of (a) EP and (b) PEP with $M = 1$, $H = 2$, and $L = 4$.

[4]. There are two fundamental components, namely, signals bearing the target information and receivers with known coordinates to pick up the signals. Based on them, localization algorithms are developed. The multistatic setup, in particular, is rapidly advancing due to its ability to use multiple separated antennas for signal transmission and reception. This makes it a promising solution for conducting localization, as signals from a number of spatially diverse antennas can provide more information about the target's position. One notable application in this context is the distributed multiple-input multiple-output (MIMO) radar [5–8], and the corresponding location estimation problem is more widely known as elliptic positioning (EP) or elliptic localization [9].

Mathematically, EP refers to finding the location $\boldsymbol{x} \in \mathbb{R}^H$ of a signal-reflecting target in H-dimensions ($H = 2$ or 3), based on the indirect bistatic range (BR) measurements [10]:

$$\hat{r}_{m,l} = r_{m,l} + \delta_{m,l} = \|\boldsymbol{x} - \boldsymbol{t}_m\|_2 + \|\boldsymbol{x} - \boldsymbol{s}_l\|_2 + \delta_{m,l}, \;\; m = 1, \ldots, M, \;\; l = 1, \ldots, L \tag{1}$$

collected using an array of spatially separated transmitters and receivers, whose positions are known *a priori* and denoted by $\{\boldsymbol{t}_m \in \mathbb{R}^H\}$ and $\{\boldsymbol{s}_l \in \mathbb{R}^H\}$, respectively. In Eq. (1), $\{\delta_{m,l}\}$ are the measurement errors, and $\{r_{m,l}\}$ and $\{\hat{r}_{m,l}\}$ are the error-free BRs and observed BRs, respectively. EP got its name from the geometric interpretation of such a localization problem: every $r_{m,l}$ with $H = 2$ induces a certain ideal-BR-representing ellipse on which the target lies, with the associated transmitter and receiver being its foci [10]. A two-dimensional (2-D) illustration is given in Fig. 1(a), where the transmitter, target, and receivers are denoted by a circle, a triangle, and squares, respectively. Geometrically speaking, $\boldsymbol{x}$ is located at the intersection of the four ellipses. By comparison, in 2-D time-of-arrival (TOA) [resp. time-difference-of-arrival (TDOA)] based localization, the target is located by intersecting multiple circles [resp. hyperbolic curves]. Since the last decade, EP has emerged as a popularly used localization technique in various applications, such as distributed MIMO radar [11], sonar [12], WSNs [13], and radio frequency identification (RFID) systems [14].

1.1. Related works

Over the years, many least squares (LS) formulations and algorithms have been devised to make statistical sense out of BR measurement data $\{\hat{r}_{m,l}\}$ immersed in the zero-mean Gaussian disturbances $\{\delta_{m,l}\}$. Generally speaking, there are direct and indirect approaches. The direct methodology processes the received signals to directly calculate the target location by H-dimensional search or other similar methods, at the cost of huge computational complexity. The indirect approach, on the contrary, determines $\boldsymbol{x}$ by solving estimation problems established from Eq. (1). Depending on the optimization criterion relied on, they can be

roughly categorized into three types: closed-form or exact solutions [11,15–18], semidefinite programming (SDP) [19,20], and iterative methods that are employed to directly realize the nonlinear LS (NLS) estimator [21,22]. While explicit algebraic solutions are famous for their simplicity, convex relaxation approaches usually have a higher level of noise-tolerance (i.e., less governed by the threshold effect). Their direct NLS counterparts, in comparison, will be capable of striking a nicer balance between computational efficiency and estimation accuracy if properly initialized.

For multistatic systems, there are application scenarios of EP where it would be more appropriate to assume that the prior knowledge of the transmitter positions is unavailable, namely, in a passive EP (PEP) configuration. A typical example is underwater WSNs with a simplified hardware structure, where the moored buoys may be exempted from determining and broadcasting their own positions [23,24]. The concern of being unable to acquire the position information of transmitters in time exists in location-based intelligent transportation systems as well [25], given that the neighboring map of a certain on-road vehicle is always changing. In addition to the BR measurements Eq. (1) which are associated merely with the indirect signal transmission paths, PEP with unknown $\{\boldsymbol{t}_m\}$ also utilizes the direct path components [24]:

$$\hat{d}_{m,l} = d_{m,l} + \epsilon_{m,l} = \|\boldsymbol{t}_m - \boldsymbol{s}_l\|_2 + \epsilon_{m,l}, \;\; m = 1, \ldots, M, \;\; l = 1, \ldots, L, \tag{2}$$

where $\{\epsilon_{m,l}\}$, $\{d_{m,l}\}$, and $\{\hat{d}_{m,l}\}$ are the observation errors, the error-free direct ranges, and the observed direct ranges, respectively. An illustration of 2-D PEP based on Fig. 1(a) is offered in Fig. 1(b). It is plainly seen that for $H = 2$ every $d_{m,l}$ defines a receiver-centered circle on which the transmitter must lie. Analogous to the situation of EP, the LS estimation of $\boldsymbol{x}$ and $\{\boldsymbol{t}_m\}$ from $\{\hat{r}_{m,l}, \hat{d}_{m,l}\}$ with zero-mean Gaussian distributed $\{\delta_{m,l}, \epsilon_{m,l}\}$ in PEP can be performed either linearly in closed-form [24] or nonlinearly by means of SDP [26]. Later on, there followed extensions of PEP into the more flexible cases of imperfect transmitter clock synchronization [27] and unknown signal propagation speed [28], respectively. The two corresponding estimation problems have been addressed similarly by linear LS [27] and/or SDP techniques [27,28].

The impact of unreliable sensor observations should further be taken into account in practical localization scenarios. In multistatic systems, outliers might occur because of the mutual interference of multiple transmitted signals under the low signal-to-interference-plus-noise ratio (SINR) regime, or attributed to the non-line-of-sight (NLOS) propagation. Such phenomena are common not only in daily terrestrial environments (e.g., indoor positioning [29] and vehicle self-localization in urban canyons [30]) but also at the oceanic scenes that involve autonomous underwater vehicles [31]. As the occurrence of outliers obviously violates the zero-mean Gaussian noise assumption in the sensor observations, the positioning performance of the traditional LS schemes will be deteriorated greatly if no preventive measures are put in place [32]. A lot of effort has been made to reduce the negative effects of outliers on the BR-only EP estimator, e.g., those that resort to the balancing parameter approximation [33–35] or robust statistics [36–38]. Nonetheless, the extension of the problem to PEP remains open to further investigation because of the model disparity between PEP and EP.

1.2. Motivation and contribution

As outlined in what follows, the motivation and contribution of our study are basically threefold:

(a) This work addresses the outlier-sensitivity from which the off-the-shelf SDP-based PEP approach [26] may suffer, by developing a computationally efficient neurodynamic optimization solution based on the Lagrange programming neural network (LPNN) [39]. In this paper, we specifically focus on the case of PEP, which differs from the existing LPNN solution that is designed for non-passive EP. By considering PEP, our research extends the application of LPNN to a novel and challenging context. We believe that this distinction adds value to the existing literature by addressing a specific problem that has not been previously explored using the LPNN framework.

(b) Additionally, our approach incorporates robustness considerations, which are lacking in conventional PEP schemes that were derived by following the non-outlier-resistant LS criterion [24,26]. our LPNN scheme minimizes a certain differentiable, statistically robustified objective function for PEP and, thus, achieving robustness to the presence of unreliable data. We also discuss in detail how to choose the appropriate objective function in an operational setting. By addressing the presence of outliers, our method enhances the reliability and accuracy of the location estimation results in adverse localization environments.

(c) Furthermore, while the majority of techniques in the literature on EP and PEP were validated just by random synthetic data, this contribution includes results of not only computer simulations but also acoustic positioning experiments. Here, real experimental data that can generally be modeled as zero-mean Gaussian distributions (resp. contain outliers) were collected during sound-based ranging without (resp. with) human body obstructions. Through these extensive studies, the superiority of our proposal over [26] in terms of outlier-resistance and computational efficiency is demonstrated.

1.3. Paper organization and notations

The remainder of this paper is structured as follows. Section 2 briefly reviews the maximum likelihood estimation (MLE) formulation for PEP in zero-mean Gaussian noise and the representative existing work [26], and provides an introduction to the background of LPNN. Section 3 develops our LPNN-based algorithm, whose important properties including the stability and computational complexity are discussed in Section 4. Performance evaluations are conducted in Section 5. Finally, Section 6 concludes the article.

Notations: Boldface lower-case and boldface capital letters are used to denote vectors and matrices, respectively. $\|\cdot\|_2$ stands for the ℓ_2-norm. $|\cdot|$ represents the absolute deviation function. $(\cdot)^T$, $(\cdot)^{-1}$, and $\nabla(\cdot)$ are the transpose, inverse, and gradient operators, respectively. $\mathbf{0}_{a\times b}\in\mathbb{R}^{a\times b}$ represents the $a\times b$ zero matrix. $\boldsymbol{A}\succeq\mathbf{0}$ (resp. $\boldsymbol{A}\succ\mathbf{0}$) means that $\boldsymbol{A}$ is a positive semidefinite (resp. definite) matrix. diag$(\cdot)$ (resp. blkdiag$(\cdot)$) denotes a square (resp. block) diagonal matrix with the given inputs on its main diagonal.

2. Problem formulation and preliminaries

2.1. MLE formulation for PEP in zero-mean Gaussian noise

Based on the principle of MLE, PEP under the assumption that $\{\delta_{m,l},\epsilon_{m,l}\}$ are zero-mean uncorrelated Gaussian processes with variances $\{\sigma^2_{\boldsymbol{r},m,l},\sigma^2_{\boldsymbol{d},m,l}\}$ is formulated as [10,40]

$$\min_{\boldsymbol{x},\{\boldsymbol{t}_m\}}\sum_{m=1}^{M}\sum_{l=1}^{L}\left[\left(\hat{r}_{m,l}-\|\boldsymbol{x}-\boldsymbol{t}_m\|_2-\|\boldsymbol{x}-\boldsymbol{s}_l\|_2\right)^2/\sigma^2_{\boldsymbol{r},m,l}+\left(\hat{d}_{m,l}-\|\boldsymbol{t}_m-\boldsymbol{s}_l\|_2\right)^2/\sigma^2_{\boldsymbol{d},m,l}\right]. \tag{3}$$

2.2. Modified NLS estimator

A natural way to handle nonconvex NLS problems like Eq. (3) is to cast them as quadratically constrained quadratic programs (QCQPs) [38,41], which permit a convenient approximate solution with the help of semidefinite relaxation (SDR) [42]. Nevertheless, as there are many auxiliary distance variables involved with this kind of NLS formulation (i.e., associated with every ℓ_2-norm terms), simply applying SDR thereto will often give rise to tightness issues and result in unreliable location estimates [43]. To bypass these shortcomings, the authors of [26] performed a nonlinear model transformation to Eqs. (1) and (2) before establishing a modified weighted NLS problem. They, unlike [38], defined auxiliary variable only for the transmitter-target distances: $d_m^t = \|\boldsymbol{x} - \boldsymbol{t}_m\|_2$ (for $m = 1, \ldots, M$). As such, squaring both sides of $r_{m,l} - d_m^t = \|\boldsymbol{x} - \boldsymbol{s}_l\|_2$, substituting $r_{m,l}$ with $\hat{r}_{m,l} - \delta_{m,l}$, and neglecting $\{\delta_{m,l}^2\}$ brought them to

$$\begin{aligned} &\tfrac{1}{2}\left(\hat{r}_{m,l}^2 - \|\boldsymbol{s}_l\|_2^2\right) + \boldsymbol{s}_l^T\boldsymbol{x} - \tfrac{1}{2}\|\boldsymbol{x}\|_2^2 - d_m^t\hat{r}_{m,l} + \tfrac{1}{2}(d_m^t)^2 \approx \|\boldsymbol{x} - \boldsymbol{s}_l\|_2\delta_{m,l}, \\ &\qquad m = 1, \ldots, M, \;\; l = 1, \ldots, L. \end{aligned} \tag{4}$$

Analogously, for the direct ranges Eq. (2) the following relations hold:

$$\tfrac{1}{2}\left(\hat{d}_{m,l}^2 - \|\boldsymbol{s}_l\|_2^2\right) + \boldsymbol{s}_l^T\boldsymbol{t}_m - \tfrac{1}{2}\|\boldsymbol{t}_m\|_2^2 \approx \|\boldsymbol{t}_m - \boldsymbol{s}_l\|_2\epsilon_{m,l}, \;\; m = 1, \ldots, M, l = 1, \ldots, L. \tag{5}$$

Rewriting Eqs. (4) and (5) in a more compact matrix form $\boldsymbol{b} \approx \boldsymbol{A}\boldsymbol{z} + \boldsymbol{e}$, yields an approximate weighted NLS problem[1]:

$$\min_{\boldsymbol{z}} \; (\boldsymbol{b} - \boldsymbol{A}\boldsymbol{z})^T\boldsymbol{W}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{z}) \tag{6}$$

where

$$\boldsymbol{z} = \left[\boldsymbol{x}^T, \boldsymbol{t}_1^T, \ldots, \boldsymbol{t}_M^T, \|\boldsymbol{x}\|_2^2, \|\boldsymbol{t}_1\|_2^2, \ldots, \|\boldsymbol{t}_M\|_2^2, d_1^t, \ldots, d_M^t, (d_1^t)^2, \ldots, (d_M^t)^2\right]^T \in \mathbb{R}^{H+HM+1+3M} \tag{7}$$

denoting the parameter vector formulated as a replacement for the intractable ML estimation problem Eq. (3). Subsequently, [26] followed a procedure that had perhaps become standardized in the literature on SDP-based localization [41,44,45] to construct a matrix $\boldsymbol{Z}$ as $\boldsymbol{Z} = \boldsymbol{z}\boldsymbol{z}^T$ and utilized it to re-express the objective function of Eq. (6) and the constraints specifying relations between the elements of $\boldsymbol{z}$ all in a disciplined convex programming form Grant and Boyd [46]. Relaxing $\boldsymbol{Z} = \boldsymbol{z}\boldsymbol{z}^T$ to $\left[\boldsymbol{Z}, \boldsymbol{z}; \boldsymbol{z}^T, 1\right] \succeq \boldsymbol{0}_{(H+HM+2+3M)\times(H+HM+2+3M)}$ ultimately yields a disciplined SDP problem, which can be easily solved using the MATLAB CVX package [47].

2.3. PEP in outlier environment

To remedy the vulnerability of Eqs. (3) and (6) in the practical situations where $\{\hat{r}_{m,l}, \hat{d}_{m,l}\}$ may contain outliers, here we formulate the following statistically robustified variant of Eq. (3):

$$\min_{\boldsymbol{x}, \{\boldsymbol{t}_m\}} \; \sum_{m=1}^{M}\sum_{l=1}^{L}\left[\psi\left(\hat{r}_{m,l} - r_{m,l}\right) + \psi\left(\hat{d}_{m,l} - d_{m,l}\right)\right], \tag{8}$$

[1] Please see Appendix B in [26] for the definitions of the regressand vector $\boldsymbol{b} \in \mathbb{R}^{2ML}$, regressor matrix $\boldsymbol{A} \in \mathbb{R}^{(2ML)\times(H+HM+1+3M)}$, error vector $\boldsymbol{e} \in \mathbb{R}^{2ML}$, and the weighting matrix $\boldsymbol{W} \in \mathbb{R}^{(2ML)\times(2ML)}$.

where $\psi(\cdot)$ is a certain differentiable cost function that is less sensitive to outliers than its ℓ_2 competitor.

The primary goal of this work is to develop algorithms for effectively solving Eq. (8)[2] and, even more crucially, doing so in a computationally feasible way. Our solution is built upon the LPNN [39], whose fundamental operating mechanism is explained in the following subsection.

2.4. LPNN

LPNN is a commonly used neurodynamic approach for solving differentiable nonlinear optimization problems with equality constraints [39]. This methodology is based on the augmented Lagrange multiplier theory, using the neural dynamics to drive neuron state transformation. When the network reaches its equilibrium, an optimal solution that satisfies the constraints can be obtained by measuring the neuron outputs.

Consider a general nonlinear constrained optimization problem:

$$\min_{\boldsymbol{u}} \ f(\boldsymbol{u}), \quad \text{s.t. } \boldsymbol{g}(\boldsymbol{u}) = \mathbf{0}_{M\times 1}, \tag{9}$$

where $\boldsymbol{u} = [u_1, \ldots, u_N]^T \in \mathbb{R}^N$, $f \in \mathbb{R}^N \to \mathbb{R}$, and $\boldsymbol{g} \in \mathbb{R}^N \to \mathbb{R}^M$. The LPNN tackles Eq. (9) by constructing the augmented Lagrangian:

$$\mathcal{L}(\boldsymbol{u}, \boldsymbol{\lambda}) = f(\boldsymbol{u}) + \boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{u}) + \tfrac{\rho}{2} h(\boldsymbol{u}), \tag{10}$$

where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_M]^T \in \mathbb{R}^M$ is a vector containing the Lagrange multipliers and the last term is an augmented term that is added to improve convexity and stability, with $\rho > 0$ being the augmented Lagrangian parameter [39].

As for the use case of Eq. (9), in the LPNN there are N variable neurons and M Lagrange neurons to hold $\boldsymbol{u}$ and $\boldsymbol{\lambda}$, respectively. The associated dynamical equations are

$$\frac{1}{\epsilon}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} = -\frac{\partial \mathcal{L}(\boldsymbol{u}, \boldsymbol{\lambda})}{\partial \boldsymbol{u}}, \tag{11a}$$

$$\frac{1}{\epsilon}\frac{\mathrm{d}\boldsymbol{\lambda}}{\mathrm{d}t} = \frac{\partial \mathcal{L}(\boldsymbol{u}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}}, \tag{11b}$$

where t is the time variable and ϵ a time constant depending on the neural circuit. In this paper, we simply assume that $\epsilon = 1$. While Eq. (11a) aims to seek for a solution of $\boldsymbol{u}$ with the minimum objective value, Eq. (11b) plays the role of guiding the solution trajectory towards the feasible region. Obviously, the objective and constraint functions have to be all differentiable so that the dynamics described by Eq. (11) are enabled. When the LPNN dynamics settle down at an equilibrium point, the output of the neurons will correspond to a Karush-Kuhn-Tucker solution for Eq. (9), i.e., satisfying the first-order necessary conditions of optimality.

The augmented term in Eq. (10) will accelerate the convergence of the neurodynamics, and any neural state that does not satisfy the constraints will be penalized by it. In the initial state, the constraints might be severely violated and hence the augmented term forces the

[2] The general form of Eq. (8) offers us a higher degree of flexibility in different scenarios. While the specific discussions about how we choose the objective function of Eq. (8) according to the PEP environments, will be given in Section 5.

state to approach the feasible region quickly. As the network approaches the feasible region, it will have gradually less impact on the convergence. When the network finally reaches a stable state, the augmented term will be equal to 0, i.e., not affecting the optimal solution.

3. Algorithm development

In this section, the LPNN is applied to cope with Eq. (8).

3.1. Equality constrained reformulation of Eq. (8)

By incorporating additional auxiliary distance variables, the unconstrained optimization problem Eq. (8) can be equivalently written as

$$\min_{\boldsymbol{y}} \sum_{m=1}^{M}\sum_{l=1}^{L}\left[\psi(\hat{r}_{m,l}-d_m^t-d_l^s)+\psi(\hat{d}_{m,l}-d_{m,l})\right],$$

$$\text{s.t. } (d_m^t)^2=\|\boldsymbol{x}-\boldsymbol{t}_m\|_2^2,\ m=1,\ldots,M, \tag{12a}$$

$$(d_l^s)^2=\|\boldsymbol{x}-\boldsymbol{s}_l\|_2^2,\ l=1,\ldots,L, \tag{12b}$$

$$d_{m,l}^2=\|\boldsymbol{t}_m-\boldsymbol{s}_l\|_2^2,\ m=1,\ldots,M,\ l=1,\ldots,L, \tag{12c}$$

$$d_m^t\geq 0, m=1,\ldots,M, \tag{12d}$$

$$d_l^s\geq 0, l=1,\ldots,L, \tag{12e}$$

$$d_{m,l}\geq 0, m=1,\ldots,M, l=1,\ldots,L, \tag{12f}$$

where $\psi(d_m^t,d_l^s)$ and $\psi(d_{m,l})$ are alterable robust loss functions imposed upon the indirect and direct path components, and $\boldsymbol{y}$ is a vector containing $H+HM+M+L+ML$ decision variables:

$$\boldsymbol{y}=\left[\boldsymbol{x}^T,\boldsymbol{t}_1^T,\ldots,\boldsymbol{t}_M^T,d_1^t,\ldots,d_M^t,d_1^s,\ldots,d_L^s,d_{1,1},\ldots,d_{M,1},d_{1,2},\ldots,d_{M,L}\right]^T\in\mathbb{R}^{H+HM+M+L+ML} \tag{13}$$

Clearly, the existence of inequality constraints in Eqs. (12d), (12e) and (12f) impedes the direct application of LPNN. In the Appendix, we show that these inequality constraints can actually be removed under mild assumptions on the range measurements, thereby simplifying the formulation to

$$\min_{\boldsymbol{y}} \sum_{m=1}^{M}\sum_{l=1}^{L}\left[\psi(\hat{r}_{m,l}-d_m^t-d_l^s)+\psi(\hat{d}_{m,l}-d_{m,l})\right],$$

$$\text{s.t. } (d_m^t)^2=\|\boldsymbol{x}-\boldsymbol{t}_m\|_2^2,\ m=1,\ldots,M, \tag{14a}$$

$$(d_l^s)^2=\|\boldsymbol{x}-\boldsymbol{s}_l\|_2^2,\ l=1,\ldots,L, \tag{14b}$$

$$d_{m,l}^2=\|\boldsymbol{t}_m-\boldsymbol{s}_l\|_2^2,\ m=1,\ldots,M,\ l=1,\ldots,L, \tag{14c}$$

3.2. Dynamical equation derivations

The augmented Lagrangian of Eq. (14) is

$$\begin{aligned}\mathcal{L}_\rho(\mathbf{y},\boldsymbol{\lambda}) = &\sum_{m=1}^{M}\sum_{l=1}^{L}\Big[\psi(\hat{r}_{m,l}-d_m^t-d_l^s)+\psi(\hat{d}_{m,l}-d_{m,l})\Big]+\sum_{m=1}^{M}\lambda_m\big[(d_m^t)^2-\|\boldsymbol{x}-\boldsymbol{t}_m\|_2^2\big]\\ &+\sum_{l=1}^{L}\lambda_{M+l}\big[(d_l^s)^2-\|\boldsymbol{x}-\boldsymbol{s}_l\|_2^2\big]+\sum_{m=1}^{M}\sum_{l=1}^{L}\lambda_{M+L+M(l-1)+m}\big(d_{m,l}^2-\|\boldsymbol{t}_m-\boldsymbol{s}_l\|_2^2\big)\\ &+\frac{\rho}{2}\sum_{m=1}^{M}\big((d_m^t)^2-\|\boldsymbol{x}-\boldsymbol{t}_m\|_2^2\big)^2+\frac{\rho}{2}\sum_{l=1}^{L}\big((d_l^s)^2-\|\boldsymbol{x}-\boldsymbol{s}_l\|_2^2\big)^2\\ &+\frac{\rho}{2}\sum_{m=1}^{M}\sum_{l=1}^{L}\big(d_{m,l}^2-\|\boldsymbol{t}_m-\boldsymbol{s}_l\|_2^2\big)^2,\end{aligned}\tag{15}$$

where $\boldsymbol{\lambda}\in\mathbb{R}^{M+L+ML}$ is the Lagrange multiplier vector. As described in Eq. (10), the last three terms in Eq. (15) serve as auxiliary terms within the augmented LPNN model that are added to improve convexity and stability, with $\rho>0$ being the augmented Lagrangian parameter [39].

The choice of different values for ρ can have an impact on the effectiveness of the augmented terms in LPNN. Simply put, an appropriate value of ρ can help accelerate convergence [39], while too large/small ρ will lead to instability of the dynamics. In Section 5, we will give an explanation regarding the selection of ρ in our numerical tests.

According to Eq. (11), the dynamical equations for the LPNN handling Eq. (14) will be

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = &-2\sum_{i=1}^{M}(\boldsymbol{t}_i-\boldsymbol{x})\cdot\big(\lambda_i+\rho\big[(d_i^t)^2-\|\boldsymbol{x}-\boldsymbol{t}_i\|_2^2\big]\big)\\ &-2\sum_{j=1}^{L}(\boldsymbol{s}_j-\boldsymbol{x})\cdot\big(\lambda_{M+j}+\rho\big[(d_j^s)^2-\|\boldsymbol{x}-\boldsymbol{s}_j\|_2^2\big]\big),\\ &i=1,\ldots,M,\ \ j=1,\ldots,L,\end{aligned}\tag{16}$$

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{t}_i}{\mathrm{d}t} = &-2(\boldsymbol{x}-\boldsymbol{t}_i)\cdot\big(\lambda_i+\rho\big[(d_i^t)^2-\|\boldsymbol{x}-\boldsymbol{t}_i\|_2^2\big]\big)\\ &-2\sum_{j=1}^{L}(\boldsymbol{s}_j-\boldsymbol{t}_i)\cdot\big[\lambda_{M+L+M(j-1)+i}+\rho\big(d_{i,j}^2-\|\boldsymbol{t}_i-\boldsymbol{s}_j\|_2^2\big)\big],\ i=1,\ldots,M,\end{aligned}\tag{17}$$

$$\frac{\mathrm{d}d_i^t}{\mathrm{d}t} = \sum_{j=1}^{L}-\frac{\mathrm{d}\psi(\hat{r}_{i,j}-d_i^t-d_j^s)}{\mathrm{d}d_i^t}-2\lambda_i d_i^t-2\rho\big[(d_i^t)^2-\|\boldsymbol{x}-\boldsymbol{t}_i\|_2^2\big]d_i^t,\ \ i=1,\ldots,M,\tag{18}$$

$$\frac{\mathrm{d}d_j^s}{\mathrm{d}t} = \sum_{i=1}^{M}-\frac{\mathrm{d}\psi(\hat{r}_{i,j}-d_i^t-d_j^s)}{\mathrm{d}d_j^s}-2\lambda_{M+j}d_j^s-2\rho\big[(d_j^s)^2-\|\boldsymbol{x}-\boldsymbol{s}_j\|_2^2\big]d_j^s,\ \ j=1,\ldots,L,\tag{19}$$

$$\frac{\mathrm{d}d_{i,j}}{\mathrm{d}t} = -\frac{\mathrm{d}\psi(\hat{d}_{i,j}-d_{i,j})}{\mathrm{d}d_{i,j}}-2\lambda_{M+L+M(j-1)+i}d_{i,j}-2\rho\big(d_{i,j}^2-\|\boldsymbol{t}_i-\boldsymbol{s}_j\|_2^2\big)d_{i,j},\ \ i=1,\ldots,M,$$

Table 1
Typical options for $\psi(\cdot)$ in range-based localization.

Discrepancy measure $\psi(y)$	Description	Parameter	Application scenario
y^2	ℓ_2	—	Generally applicable in zero-mean Gaussian noise environments
$\log((\exp(\gamma y)+\exp(-\gamma y))/2)/\gamma$	Smoothed ℓ_1	$\gamma > 0$	TDOA-based hyperbolic localization [48]
$1-\exp(-y^2/(2\sigma_G^2))$	C-loss	$\sigma_G > 0$	TOA-based circular localization [49]
$y^2/(y^2+\epsilon^2)$	GM	ϵ	TOA-based circular localization [50]
$\log(1+y^2/v)$	Cauchy	$v > 0$	Received signal strength based circular localization [51]

"Correntropy-induced loss" and "Geman-McClure" are abbreviated to "C-loss" and "GM", respectively.

$$j = 1, \ldots, L, \tag{20}$$

$$\frac{\mathrm{d}\lambda_i}{\mathrm{d}t} = (d_i^t)^2 - \|\boldsymbol{x} - \boldsymbol{t}_i\|_2^2, \ \ i = 1, \ldots, M, \tag{21}$$

$$\frac{\mathrm{d}\lambda_{M+j}}{\mathrm{d}t} = (d_j^s)^2 - \|\boldsymbol{x} - \boldsymbol{s}_j\|_2^2, \ \ j = 1, \ldots, L, \tag{22}$$

$$\frac{\mathrm{d}\lambda_{M+L+M(j-1)+i}}{\mathrm{d}t} = d_{i,j}^2 - \|\boldsymbol{t}_i - \boldsymbol{s}_j\|_2^2, \ \ i = 1, \ldots, M, \ \ j = 1, \ldots, L. \tag{23}$$

Here, Eqs. (16)–(20) are used for minimizing the objective function, whereas Eqs. (21)–(23) ensure that the constraints are satisfied at the equilibrium point. In this case, there are $3M + L + ML + 2$ variable neurons to hold the $3M + L + ML + 2$ decision variables and $M + L + ML$ Lagrangian neurons for the $M + L + ML$ Lagrange multiplier variables.

3.3. Possible objective functions

This subsection discusses the common choices for the objective function $\psi(\cdot)$ in range-based localization, as summarized in Table 1 and depicted in Fig. 2.

The l_2 loss $\psi(y) = y^2$ has been generally used in zero-mean Gaussian noise environments. However, it does not work well in the presence of outliers. The l_1 loss $\psi(y) = |y|$, in comparison, will be less affected by the impulsive components as it places a lower emphasis on large errors than its l_2 counterpart. Nonetheless, since the l_1 loss is non-differentiable, it is more common to utilize a smoothed version of it, namely:

$$\psi(y) = \frac{1}{\gamma}\log\left(\frac{e^{\gamma y} + e^{-\gamma y}}{2}\right), \tag{24}$$

where $\gamma > 0$ is a given constant. For a sufficiently large value of γ, Eq. (24) will well approximate the original ℓ_1 loss [36].

The correntropy-induced loss (C-loss), Geman-McClure (GM) loss, and the Cauchy loss are three other typical objective functions to mitigate the negative effects of outliers. Owing to their close relationship to the redescending M-estimators, these cost functions are known to have stronger robustness to heavy-tailed large errors.

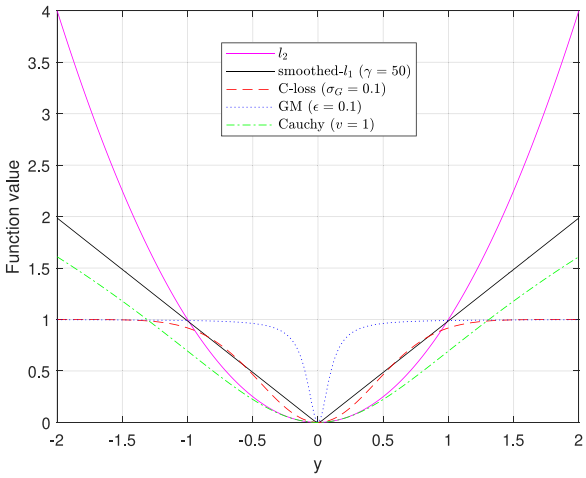


Fig. 2. Comparison of different objective functions.

4. Stability and complexity analyses

4.1. Local stability

Stability and convergence are perhaps the two most important facets of a certain neurodynamic approach. Due to the nonconvexity of the problem being solved, we are only able to investigate the local stability of the LPNN. The local stability ensures that the minimum of a constrained optimization problem is a stable point of the neurodynamics, which also guarantees the convergence of the network at the minimum point. Specifically, there are two sufficient conditions that a minimum point $(\boldsymbol{y}^*, \boldsymbol{\lambda}^*)$ should satisfy to guarantee the local stability of the LPNN [36,39]:

(i) $\nabla^2_{\boldsymbol{yy}}\mathcal{L}_\rho(\boldsymbol{y}^*, \boldsymbol{\lambda}^*) \succ \mathbf{0}$, i.e., the Hessian matrix of the Lagrangian should be positive definite at $(\boldsymbol{y}^*, \boldsymbol{\lambda}^*)$.
(ii) The gradient vectors of the constraints with respect to $\boldsymbol{y}$ are linearly independent at such a local minimum point.

The first condition is met by means of the augmented term, as it has been pointed out [39] that the Hessian matrix will be positive definite under mild conditions if ρ is sufficiently large. Next, we proceed to analyze the linear independence of the gradient vectors of the constraints (taking the 2-D case as an example).

Let $\{y^*, \lambda^*\}$[3] be a locally optimal solution of our constrained optimization problem Eq. (14) and q_i denote the ith constraint in Eq. (14). In this way, we obtain the following $M + L + ML$ rearranged equations:

$$q_m(\boldsymbol{y}) = (d_m^t)^2 - \|\boldsymbol{x} - \boldsymbol{t}_m\|_2^2, \tag{25}$$

$$q_{M+l}(\boldsymbol{y}) = (d_l^s)^2 - \|\boldsymbol{x} - \boldsymbol{s}_l\|_2^2, \tag{26}$$

[3] For notational convenience, we assume that the asterisk applies to every element of the corresponding vector by default.

$$q_{M+L+(m-1)l+l}(\boldsymbol{y}) = (d_{m,l})^2 - \|\boldsymbol{t}_m - \boldsymbol{s}_l\|_2^2, \tag{27}$$

where $m = 1, \ldots, M$ and $l = 1, \ldots, L$. The gradient vectors for Eqs. (25)–(27) at $\boldsymbol{y}^*$ are

$$\begin{aligned}
&\frac{\partial q_m(\boldsymbol{y})}{\partial \boldsymbol{y}}\bigg|_{\boldsymbol{y}=\boldsymbol{y}^*} \\
&= 2[-(\boldsymbol{x}^* - \boldsymbol{t}_m)^T, \mathbf{0}_{1\times 2(m-1)}, (\boldsymbol{x} - \boldsymbol{t}_m^*)^T, \mathbf{0}_{1\times 2(M-m)}, \mathbf{0}_{1\times(m-1)}, d_m^{t*}, \mathbf{0}_{1\times(M-m)}, \\
&\mathbf{0}_{1\times L}, \mathbf{0}_{1\times ML}]^T, \\
&m = 1, ..., M,
\end{aligned} \tag{28}$$

$$\begin{aligned}
&\frac{\partial q_{M+l}(\boldsymbol{y})}{\partial \boldsymbol{y}}\bigg|_{\boldsymbol{y}=\boldsymbol{y}^*} \\
&= 2[-(\boldsymbol{x}^* - \boldsymbol{s}_l)^T, \mathbf{0}_{1\times 2M}, \mathbf{0}_{1\times M}, \mathbf{0}_{1\times(l-1)}, d_l^{s*}, \mathbf{0}_{1\times(L-l)}, \mathbf{0}_{1\times ML}]^T, \\
&l = 1, \ldots, L,
\end{aligned} \tag{29}$$

$$\begin{aligned}
&\frac{\partial q_{M+L+(m-1)l+l}(\boldsymbol{y})}{\partial \boldsymbol{y}}\bigg|_{\boldsymbol{y}=\boldsymbol{y}^*} \\
&= 2[\mathbf{0}_{1\times 2}, \mathbf{0}_{1\times 2(m-1)}, -(\boldsymbol{t}_m^* - \boldsymbol{s}_l)^T, \mathbf{0}_{1\times 2(M-m)}, \mathbf{0}_{1\times M}, \mathbf{0}_{1\times L}, \mathbf{0}_{1\times(m-1)l+l-1)}, \\
&d_{m,l}^*, \mathbf{0}_{1\times(ML-(m-1)l-l)}]^T, \\
&m = 1, \ldots, M, l = 1, \ldots, L.
\end{aligned} \tag{30}$$

In Eq. (31), we provide a clearer illustration in terms of the gradient matrix.

$$\begin{aligned}
&\left[\frac{\partial q_1(\boldsymbol{y}^*)}{\partial \boldsymbol{y}}, ..., \frac{\partial q_M(\boldsymbol{y}^*)}{\partial \boldsymbol{y}}, \frac{\partial q_{M+1}(\boldsymbol{y}^*)}{\partial \boldsymbol{y}}, ..., \frac{\partial q_{M+L}(\boldsymbol{y}^*)}{\partial \boldsymbol{y}}, \frac{\partial q_{M+L+1}(\boldsymbol{y}^*)}{\partial \boldsymbol{y}}, ..., \frac{\partial q_{M+L+ML}(\boldsymbol{y}^*)}{\partial \boldsymbol{y}}\right] \\
&= \left[\begin{array}{c|c|c}
-2[\boldsymbol{x}^* - \boldsymbol{t}_1, ..., \boldsymbol{x}^* - \boldsymbol{t}_M] & -2[\boldsymbol{x}^* - \boldsymbol{s}_1, ..., \boldsymbol{x}^* - \boldsymbol{s}_L] & \mathbf{0}_{2\times ML} \\
\hline
2\text{blkdiag}(\boldsymbol{x} - \boldsymbol{t}_1^*, ..., \boldsymbol{x} - \boldsymbol{t}_M^*) & \mathbf{0}_{2M\times L} & -2\text{blkdiag}(\boldsymbol{t}_1^* - \boldsymbol{s}_1, ..., \boldsymbol{t}_1^* - \boldsymbol{s}_L, \boldsymbol{t}_2^* - \boldsymbol{s}_1, ..., \boldsymbol{t}_M^* - \boldsymbol{s}_L) \\
\hline
2\text{diag}(d_1^{t*}, ..., d_M^{t*}) & \mathbf{0}_{M\times L} & \mathbf{0}_{M\times ML} \\
\hline
\mathbf{0}_{L\times M} & 2\text{diag}(d_1^{s}*, ..., d_L^{s}*) & \mathbf{0}_{L\times ML} \\
\hline
\mathbf{0}_{ML\times M} & \mathbf{0}_{ML\times L} & 2\text{diag}(d_{1,1}^*, ..., d_{1,L}^*, d_{2,1}^*, ..., d_{M,L}^*)
\end{array}\right]
\end{aligned} \tag{31}$$

In general, we can consider that the transmitter and receiver positions are different from the target position, which implies that none of d_m^{t*}, d_l^{s*} and $d_{m,l}^*$ are equal to 0. Hence, the gradient matrix in Eq. (31) is linear independent (viz., the gradient vectors of the constraints in Eq. (14) are linearly independent) at $\boldsymbol{y} = \boldsymbol{y}^*$. Combining these two conditions, we deduce that the dynamics of the LPNN for solving Eq. (14) are stable at a local minimum point.

4.2. Implementation complexity

The LPNN was originally designed to be an analog computational technique [39]. Nonetheless, it can also be implemented numerically either by using an ordinary differential equation (ODE) solver [34] or by assigning a certain user-defined step size [22]. Here, we take the former way for realizing our LPNN approach. For this reason, measuring only its circuit complexity may not be the appropriate way to characterize the complexity of the overall methodology.

The computational cost of the numerical implementation of LPNN, on the other hand, is dominated by that of the neurodynamic operations at each iteration. Involving mostly derivative calculations, our LPNN has linear computational complexity of

$\mathcal{O}((H+2))M+2L+2ML+H)$. By comparison, the existing approach [26] built upon convex relaxation requires $\mathcal{O}((3M+2)((M+1)H+3M+2)^{3.5})$. In Section 5, we will also show the CPU times of different methods to give a more intuitive comparison.

5. Numerical results

This section evaluates the proposed algorithm in different PEP scenarios based on both random synthetic and real experimental data. Comparisons are made to the existing SDP method [26] for tackling Eq. (6), and the benchmarking estimator Eq. (3) implemented using the perfectly initialized MATLAB command for unconstrained minimization, `fminunc`. The two competitors and our LPNN algorithm are termed `SDP`, `MLE`, and `L-`*** for short, respectively, where the asterisks "***" will be substituted with the name of the corresponding loss in particular scenarios. The parameters used in `SDP` are the same as in the original literature. For the LPNN-type location estimators, based on our practical experience, we set the parameters in Table 1 as $\gamma=50$, $\sigma_G=5$, $\epsilon=20$, and $v=10$ for better stability and convergence speed. Based on the observation in [36] and our experience, the coefficient of the augmented term is fixed as $\rho=20$ to guarantee that equilibrium can be attained in a stable and efficient manner. In our simulations, varying the value of ρ within an appropriate range demonstrated only a tiny amount of influence on the convergence rate. The initial values of the variables $\boldsymbol{x}$, $\boldsymbol{t}_i$ (for $i=1,\ldots,M$), d_i^t (for $i=1,\ldots,M$), d_j^r (for $j=1,\ldots,L$), $d_{i,j}$ (for $i=1,\ldots,M$, $j=1,\ldots,L$), and $\boldsymbol{\lambda}=[\lambda_1,\ldots,\lambda_{M+L+ML}]^T$ are all randomly generated. For simplicity, we only consider the case of $H=2$ as the results of $H=3$ are in general similar. The SDP program and ODEs are handled using the MATLAB CVX package [52] and `ode15s` solver [46], respectively.

To assess the positioning accuracy, we use the root-mean-square error (RMSE), defined as

$$\mathrm{RMSE}=\sqrt{\frac{1}{N_{\mathrm{MC}}}\sum_{j=1}^{N_{\mathrm{MC}}}\|\tilde{\boldsymbol{x}}^{\{j\}}-\boldsymbol{x}^{\{j\}}\|_2^2}, \tag{32}$$

where N_{MC} denotes the number of Monte Carlo (MC) runs (fixed as $N_{\mathrm{MC}}=200$ in our tests unless otherwise specified), and $\tilde{\boldsymbol{x}}^{\{j\}}$ is the estimate of the true target location $\boldsymbol{x}^{\{j\}}$ in the jth MC run.

5.1. Results of random synthetic data

For simulation purposes, we use the Gaussian mixture model (GMM) to take into account the possible occurrence of outliers [34,53]. Specifically, $\delta_{m,l}$ and $\epsilon_{m,l}$ in Eqs. (1) and (2) are modeled as a two-component bivariate Gaussian mixture distribution, with the probability density function:

$$\begin{aligned}p\big(e_{m,l}|\beta_{m,l},\mu_{m,l},\widetilde{\mu}_{m,l},\sigma_{m,l},\widetilde{\sigma}_{m,l}\big)=&\frac{\beta_{m,l}}{\sigma_{m,l}\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{e_{m,l}-\mu_{m,l}}{\sigma_{m,l}}\right)^2\right]\\&+\frac{(1-\beta_{m,l})}{\widetilde{\sigma}_{m,l}\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{e_{m,l}-\widetilde{\mu}_{m,l}}{\widetilde{\sigma}_{m,l}}\right)^2\right],\end{aligned} \tag{33}$$

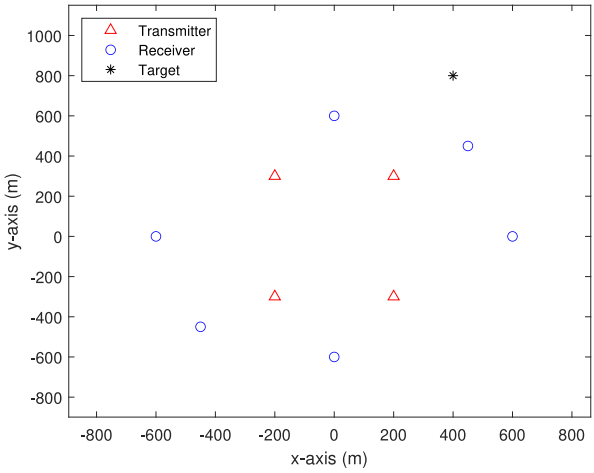


Fig. 3. Geometry of sensors and target.

where $e_{m,l}$ is the random variable being modeled, $\beta_{m,l}$ the mixture weight, $\mu_{m,l}$ the mean of the first GMM component, $\widetilde{\mu}_{m,l}$ the mean of the second GMM component, and $\sigma_{m,l}$ and $\widetilde{\sigma}_{m,l}$ the standard deviations of the first and second GMM components, respectively.

Consider a deterministically deployed PEP system with four transmitters and six receivers, namely, $M = 4$ and $L = 6$. The sensor positions are $\boldsymbol{t}_1 = [-200, -300]^T$ m, $\boldsymbol{t}_2 = [-200, 300]^T$ m, $\boldsymbol{t}_3 = [200, 300]^T$ m, $\boldsymbol{t}_4 = [200, -300]^T$ m, $\boldsymbol{s}_1 = [-450, -450]^T$ m, $\boldsymbol{s}_2 = [450, 450]^T$ m, $\boldsymbol{s}_3 = [0, 600]^T$ m, $\boldsymbol{s}_4 = [600, 0]^T$ m, $\boldsymbol{s}_5 = [-600, 0]^T$ m, and $\boldsymbol{s}_6 = [0, 600]^T$ m, whereas the target is located at $\boldsymbol{x} = [400, 800]^T$ m. Fig. 3 shows the geometry of the target and sensors.

In the first experiment, the RMSE performance of different PEP techniques in pure Gaussian noise environments (viz., without any outliers) is studied. The measurement errors are modeled as independent and identically distributed two-component Gaussian mixture processes with $\beta = 1$, $\mu = 0$, $\widetilde{\mu} = 0$, and $\widetilde{\sigma} = 0$. We vary σ from 10 m to 50 m and plot the RMSE in Fig. 4. It can be observed that `L-Closs`, `L-GM`, and `L-Cauchy` have the best performance and deliver comparable RMSE results to `MLE`. A possible reason is the shape resemblance between the corresponding robust cost functions and the ℓ_2 loss in small error regions (see Fig. 2). The `L-smoothed-`l_1 estimator performs slightly worse but still better than the `SDP` approach. This confirms the effectiveness of our LPNN method in Gaussian noise environments.

In the second experiment, we simulate the mild NLOS environment with one transmitter subject to outliers. All the range measurements associated with it are affected by NLOS errors that follow the Gaussian mixture distributions with $\beta = 0.5$, $\mu = 0$, $\widetilde{\mu} = 20$, and $\sigma = 1$, whereas the rest are under Gaussian noise of unit standard deviation. The RMSE results of varying $\widetilde{\sigma}$ from 0 m to 10 m are shown in Fig. 5. We see that `L-smoothed-`l_1 demonstrates excellent robustness against the presence of sporadic NLOS outliers, and the RMSE almost remains unchanged as $\widetilde{\sigma}$ grows. `L-Closs`, `L-GM` and `L-Cauchy` have similar RMSE performance to `MLE` and are slightly better than `SDP` when $\widetilde{\sigma}$ is not large enough.

Finally, we evaluate the performance of our proposal in an environment with a large number of outliers, where the Gaussian mixture errors are added to the BR and direct range measurements for every transmitter-receiver pair. This may be regarded as a generally contaminated

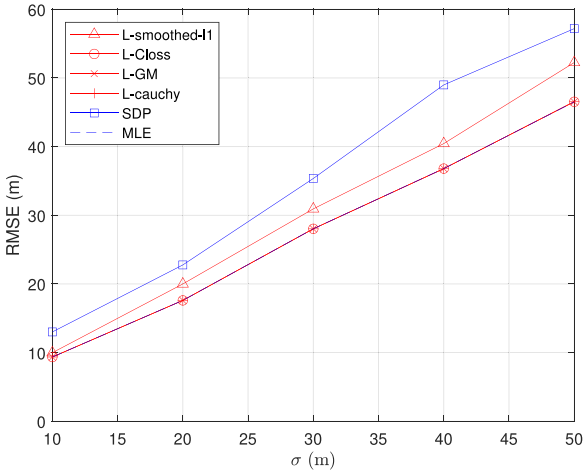


Fig. 4. RMSE versus σ in Gaussian noise environment.

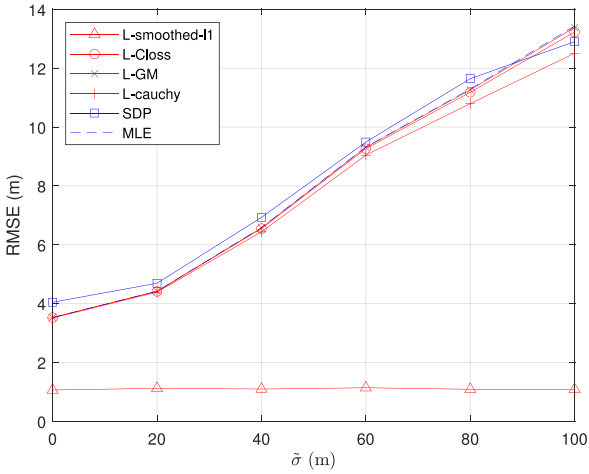


Fig. 5. RMSE versus $\widetilde{\sigma}$ in mild NLOS environment. In this experiment, one transmitter is subject to NLOS propagation.

severe NLOS scenario. The GMM-related parameters are kept the same as in the last experiment. The results are shown in Fig. 6. We observe that in the presence of a large number of outliers, the robustness of `L-smoothed-`l_1 decreases. Nevertheless, it is still capable of outperforming `SDP` for the whole range under investigation and the other approaches when $\widetilde{\sigma}$ is a larger number. The remaining three objective functions, in contrast, lead to performance comparable to `MLE` and better than `SDP`.

In summary, our proposed algorithms are superior to `SDP` in various situations. Especially, `L-smoothed-`l_1 shows excellent robustness in the mild NLOS environment.

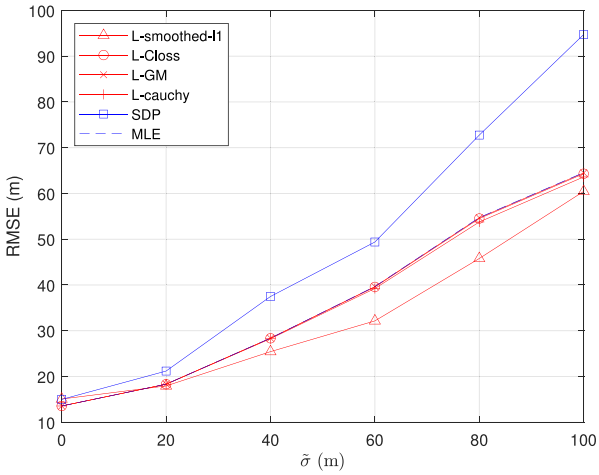


Fig. 6. RMSE versus $\tilde{\sigma}$ in severe NLOS environment. In this experiment, all transmitters and receivers are subject to NLOS propagation.

5.2. Results of real experimental data

In addition to computer simulations, onsite PEP experiments were carried out by employing an acoustic localization system. It is composed of multiple spatially separated sound-making speakers and a Huawei Mate 20 signal-receiving device with Android 10.0 system, Hisilicon Kirin 980 CPU, and 4 GB memory (i.e., a middle-level configuration commercial off-the-shelf smartphone). We modulated the chirp signals, whose frequencies were sweeping from 19 kHz to 21 kHz, to estimate the TOA-based speaker-smartphone distances using the well-known generalized cross correlation algorithm [54]. We conducted ranging $N_{\mathrm{MC}} = 40$ times for each path. To introduce NLOS errors into a certain communication channel, we positioned an experimenter on the associated transmission path, acting as a barrier and deliberately obstructing the signal. With the obtained speaker-smartphone distance observations, we constructed their BR and direct range counterparts for our purpose of PEP performance evaluation.

For simplicity, the smartphone (taking the role of the target to be localized) was always placed at the frame's origin (i.e., $\boldsymbol{x} = [0, 0]^T$ m). The PEP system under test that we constructed consists of one transmitter and four receivers, whose ground-truth positions $\boldsymbol{t}_1 = [4, 0]^T$ m, $\boldsymbol{s}_1 = [1.375, 1.452]^T$ m, $\boldsymbol{s}_2 = [2.625, 1.452]^T$ m, $\boldsymbol{s}_3 = [5, 0]^T$ m, and $\boldsymbol{s}_4 = [1.375, -1.452]^T$ m were measured via a laser range finder. Fig. 7 depicts the geometric setting.

Invoking the MATLAB routine `fitdist`, Gaussian mixture distributions were fitted to the sample data according to Eqs. (1) and (2). Table 2 summarizes the details. We see that the $\boldsymbol{x}$–$\boldsymbol{s}_3$ path is subject to NLOS propagation, whereas the others are all under lower-level Gaussian disturbances.

We then input these range data to different PEP estimators for estimating $\boldsymbol{x}$. To better align with the smaller scale ranges in the real experiments, the parameters in Table 1 were reset as $\gamma = 1$, $\sigma_G = 5$, $\epsilon = 5$, and $\nu = 10$. Table 3 reports the results of RMSE and average CPU time for `L-smoothed-`l_1,`L-Closs`,`L-GM`, `L-Cauchy`, and `SDP`. We see that compared to `SDP`,

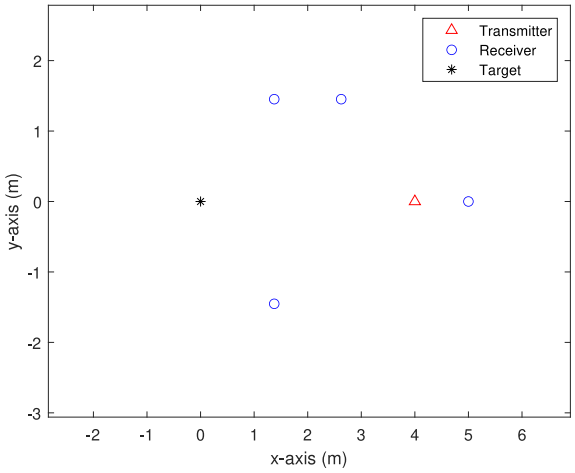


Fig. 7. Geometry of sensors in onsite experiment.

Table 2
Characteristics of Gaussian mixture distributions.

Path	β	μ	$\widetilde{\mu}$	σ	$\widetilde{\sigma}$
$\boldsymbol{t}_1 - \boldsymbol{x}$	0.495	−0.028	0.015	5.21×10^{-4}	5.8×10^{-4}
$\boldsymbol{x} - \boldsymbol{s}_1$	0.417	−0.018	−0.047	7.98×10^{-4}	4.68×10^{-4}
$\boldsymbol{x} - \boldsymbol{s}_2$	0.553	−0.036	0.009	3.68×10^{-4}	7.81×10^{-4}
$\boldsymbol{x} - \boldsymbol{s}_3$	0.900	0.442	1.795	0.1003	0.0119
$\boldsymbol{x} - \boldsymbol{s}_4$	0.422	−0.027	−0.043	7.31×10^{-4}	4.66×10^{-4}
$\boldsymbol{t}_1 - \boldsymbol{s}_1$	0.525	−0.008	−0.018	8.18×10^{-4}	0
$\boldsymbol{t}_1 - \boldsymbol{s}_2$	0.150	−0.001	−0.04	6.13×10^{-5}	2.24×10^{-4}
$\boldsymbol{t}_1 - \boldsymbol{s}_3$	0.539	−0.046	−0.007	3.57×10^{-4}	9.33×10^{-4}
$\boldsymbol{t}_1 - \boldsymbol{s}_4$	0.713	−0.019	−0.041	0.001	3.24×10^{-5}

Table 3
Performance of different PEP methods in real experiments.

Algorithm	RMSE (m)	Average CPU time (s)
L-smoothed-l_1	0.1280	0.0862
L-C-loss	0.1484	0.1824
L-GM	0.1436	0.1718
L-Cauchy	0.1423	0.1240
SDP	0.3818	0.3434

the four proposed approaches take less time and yield more accurate location estimates. This is consistent with the analyses and simulation results above.

6. Conclusion

In this contribution, we have studied the problem of robust PEP with unknown transmitter locations. Our proposal is based on the neurodynamic approach of LPNN, and we adopt a statistically robustified differentiable objective function in order to reduce the impact of outliers.

Furthermore, we have analyzed the local stability of the neural model and the implementation complexity of the proposed LPNN-based algorithms. Through both numerical simulations and acoustic localization experiments, it is demonstrated that our LPNN scheme is superior to the state-of-the-art SDP method in terms of outlier-robustness and computational efficiency.

Fixed-parameter objective functions were observed to behave differently at different noise levels, implying that the choice of the objective function parameters can also be crucial. One of the possible future research directions is the adaptive selection of such parameters.

Ethical Statement for Solid State Ionics

Hereby, we, Keyuan Hu, Wenxin Xiong, Yuwei Wang, Zhang-Lei Shi, Ge Cheng, Hing Cheung So, and Zhi Wang, consciously assure that for the manuscript "Lagrange programming neural network for robust passive elliptic positioning" the following is fulfilled:

(1) This material is the authors' own original work, which has not been previously published elsewhere.
(2) The paper is not currently being considered for publication elsewhere.
(3) The paper reflects the authors' own research and analysis in a truthful and complete manner.
(4) The paper properly credits the meaningful contributions of co-authors and co-researchers.
(5) The results are appropriately placed in the context of prior and existing research.
(6) All sources used are properly disclosed (correct citation). Literally copying of text must be indicated as such by using quotation marks and giving proper reference.
(7) All authors have been personally and actively involved in substantial work leading to the paper, and will take public responsibility for its content.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix.

In the following, we show that the inequality constraints in Eq. (12) are all removable.

Let us denote the optimal solution of Eqs. (12) by $\boldsymbol{y}^* = [\boldsymbol{x}^{*T}, \boldsymbol{t}_1^{*T}, \ldots, \boldsymbol{t}_M^{*T}, d_1^{t*}, \ldots, d_M^{t*}, d_1^{s*}, \ldots, d_L^{s*} d_{1,1}^*, \ldots, d_{M,1}^*, d_{1,2}^*, \ldots, d_{M,L}^*]^T$. It follows straightforwardly from Eqs. (12a)–(12c) that $[\boldsymbol{x}^{*T}, |\boldsymbol{t}_1^{*T}|, \ldots, |\boldsymbol{t}_M^{*T}|, |d_1^{t*}|, \ldots, |d_M^{t*}|, |d_1^{s*}|, \ldots, |d_L^{s*}|, |d_{1,1}^*|, \ldots, |d_{M,1}^*|, |d_{1,2}^*|, \ldots, |d_{M,L}^*|]^T$ is a feasible solution. On the other hand, by definition we have $\hat{r}_{m,l} \geq 0, \hat{d}_{m,l} \geq 0, d_m^t \geq 0, d_l^s \geq 0$, and $d_{m,l} \geq 0$, based on which the mild assumption that $\hat{r}_{m,l} \geq d_m^{t*}$ and $\hat{r}_{m,l} \geq d_l^{s*}$ can be fulfilled in the majority of cases of practical significance.

We start our analysis from the direct path components, the case of which may be relatively simpler. For Eq. (12f), we consider the following reverse triangle inequality:

$$\left|\hat{d}_{m,l} - d_{m,l}^*\right| \geq \left|\hat{d}_{m,l} - |d_{m,l}^*|\right| \tag{A.1}$$

Obviously, $d^*_{m,l}$ will no longer be the optimal solution if the inequality holds strictly. In other words, $\geq$ in Eq. (A.1) degrades into $=$, which means $d^*_{m,l} = |d^*_{m,l}| \ \forall m, l$. Therefore, the inequality constraints in Eq. (12f) can be discarded.

The inequality for the indirect path, in a similar manner, is constructed as

$$\left|\hat{r}_{m,l} - d^{t*}_m - d^{s*}_l\right| \geq \left|\hat{r}_{m,l} - |d^{t*}_m| - |d^{s*}_l|\right|. \tag{A.2}$$

However, this time we would need to validate Eq. (A.2) before making use of it in our discussion of optimality. To do so, we calculate the squared distance for the two associated terms as

$$\begin{aligned} R = \left|\hat{r}_{m,l} - d^{t*}_m - d^{s*}_l\right|^2 - \left|\hat{r}_{m,l} - |d^{t*}_m| - |d^{s*}_l|\right|^2 = 2\hat{r}_{m,l}\left(|d^{t*}_m| + |d^{s*}_l| - d^{t*}_m - d^{s*}_l\right) \\ + 2\left(d^{t*}_m d^{s*}_l - |d^{t*}_m||d^{s*}_l|\right). \end{aligned} \tag{A.3}$$

It can be easily obtained that no matter whether d^{t*}_m and d^{s*}_l take positive or negative values, $R \geq 0$ holds. For the same justification as above, it follows that $d^{t*}_m = |d^{t*}_m| \ \forall m \in \{1, \ldots, M\}$ and $d^{s*}_l = |d^{s*}_l| \ \forall l \in \{1, \ldots, L\}$. That is to say, we also remove the inequality constraints in Eqs. (12d) and (12e).

CRediT authorship contribution statement

Keyuan Hu: Methodology, Writing – original draft, Software, Validation. **Wenxin Xiong:** Conceptualization, Investigation. **Yuwei Wang:** Data curation. **Zhang-Lei Shi:** Formal analysis. **Ge Cheng:** Visualization. **Hing Cheung So:** Writing – review & editing. **Zhi Wang:** Conceptualization.

References

[1] Y. Huang, J. Benesty, G.W. Elko, Source localization, in: Audio Signal Processing for Next Generation Multimedia Communication Systems,, Springer, US, 2004, pp. 229–253.
[2] A. Jagoe, Mobile location services: the definitive guide, in: Upper Saddle River, Prentice-Hall, NJ, 2003.
[3] D.J. Peters, A Bayesian method for localization by multistatic active sonar, IEEE J. Ocean. Eng., 42 (1) (2017) 135–142.
[4] M. Ilyas, I. Mahgoub, L. Kelly, Handbook of sensor networks, in: Compact Wireless and Wired Sensing Systems, CRC, Boca Raton, FL, 2004.
[5] H. Godrich, A.M. Haimovich, R.S. Blum, Target localization accuracy gain in MIMO radar-based systems, IEEE Trans. Inf. Theory., 56 (6) (2010) 2783–2803.
[6] Q. He, R.S. Blum, A.M. Haimovich, Noncoherent MIMO radar for location and velocity estimation: more antennas means better performance, IEEE Trans. Signal Process. 58 (7) (2010) 3661–3680.
[7] S. Gogineni, A. Nehorai, Target estimation using sparse modeling for distributed MIMO radar, IEEE Trans. Signal Process. 59 (11) (2011) 5315–5325.
[8] P. Wang, H. Li, Target detection with imperfect waveform separation in distributed MIMO radar, IEEE Trans. Signal Process. 68 (2020) 793–807.
[9] L. Rui, K.C. Ho, Elliptic localization: performance study and optimum receiver placement, IEEE Trans. Signal Process. 62 (18) (2014) 4673–4688.
[10] H.C. So, Source localization: algorithms and analysis, in: Handbook of Position Location: Theory, Practice, and Advances, Wiley, Hoboken, NJ, USA, 2011, pp. 25–66.
[11] M. Einemo, H.C. So, Weighted least squares algorithm for target localization in distributed MIMO radar, Signal Process. 115 (2015) 144–150.
[12] S. Coraluppi, Multistatic sonar localization, IEEE J. Ocean Eng. 31 (4) (2006) 4745–4760.
[13] J. Shen, A.F. Molisch, J. Salmi, Accurate passive location estimation using TOA measurements, IEEE Trans. Wireless Commun. 11 (6) (2012) 2182–2192.

[14] D. Fortin-Simard, J.-S. Bilodeau, K. Bouchard, S. Gaboury, B. Bouchard, A. Bouzouane, Exploiting passive RFID technology for activity recognition in smart homes, IEEE Intell. Syst., 30 (4) (2015) 7–15.
[15] M. Dianat, M.R. Taban, J. Dianat, V. Sedighi, Target localization using least squares estimation for MIMO radars with widely separated antennas, IEEE Trans. Aerosp. Electron. Syst. 49 (4) (2013) 2730–2741.
[16] R. Amiri, F. Behnia, An efficient weighted least squares estimator for elliptic localization in distributed MIMO radars, IEEE Signal Process. Lett. 24 (6) (2017) 902–906.
[17] A. Noroozi, M.A. Sebt, S.M. Hosseini, R. Amiri, M.M. Nayebi, Closed-form solution for elliptic localization in distributed MIMO radar systems with minimum number of sensors, IEEE Trans. Aerosp. Electron. Syst. 56 (4) (2020) 3123–3133.
[18] R. Amiri, F. Behnia, M.A.M. Sadr, Exact solution for elliptic localization in distributed MIMO radar systems, IEEE Trans. Veh. Technol. 67 (2) (2018) 1075–1086.
[19] R. Amiri, F. Behnia, M.A.M. Sadr, Positioning in MIMO radars based on constrained least squares estimation, IEEE Commun. Lett. 21 (10) (2017) 2222–2225.
[20] Z. Zheng, H. Zhang, W.Q. Wang, Target localization in distributed MIMO radars via improved semidefinite relaxation, J. Franklin Inst. 358 (10) (2021) 5588–5598.
[21] J. Liang, Y. Chen, H.C. So, Y. Jing, Circular/hyperbolic/elliptic localization via euclidean norm elimination, Signal Process. 148 (2018) 102–113.
[22] J. Liang, C.S. Leung, H.C. So, Lagrange programming neural network approach for target localization in distributed MIMO radar, IEEE Trans. Signal Process. 64 (6) (2016) 1574–1585.
[23] Z. Zhou, J.-H. Cui, S. Zhou, Localization for large-scale underwater sensor networks, in: Proc. Int. Conf. Res. Netw., Springer-Verlag, Berlin, Germany, 2007, pp. 108–119.
[24] Y. Zhang, K.C. Ho, Multistatic localization in the absence of transmitter position, IEEE Trans. Signal Process. 67 (18) (2019) 4745–4760.
[25] F. Khelifi, A. Bradai, A. Benslimane, P. Rawat, M. Atri, A survey of localization systems in internet of things, Mobile Netw. Appl. 24 (3) (2019) 761–785.
[26] R. Zheng, G. Wang, K.C. Ho, Accurate semidefinite relaxation method for elliptic localization with unknown transmitter position, IEEE Trans. Wirel. Commun. 20 (4) (2021) 2746–2760.
[27] X. Wu, Y. Liu, X. Zhu, L. Mo, Efficient solutions for MIMO radar localization under unknown transmitter positions and offsets, IEEE Trans. Wirel. Commun. 21 (1) (2022) 505–518.
[28] Y. Li, G. Wang, Multistatic localization with unknown transmitter position and signal propagation speed, IEEE Signal Process. Lett. 29 (2022) 1427–1431.
[29] I. Guvenc, C.C. Chong, A survey on TOA based wireless localization and NLOS mitigation techniques, IEEE Commun. Surveys Tuts. 11 (3) (2009) 107–124.
[30] Y. Gu, L.-T. Hsu, S. Kamijo, GNSS/Onboard inertial sensor integration with the aid of 3-D building map for lane-level vehicle self-localization in urban canyon, IEEE Trans. Veh. Technol. 65 (6) (2016) 4274–4287.
[31] E. Olson, J.J. Leonard, S. Teller, Robust range-only beacon localization, IEEE J. Ocean. Eng. 31 (4) (2006) 949–958.
[32] A.M. Zoubir, V. Koivunen, E. Ollila, M. Muma, Robust Statistics for Signal Processing, Cambridge Univ. Press, Cambridge, U.K., 2018.
[33] W. Xiong, C. Schindelhauer, H.C. So, Error-reduced elliptic positioning via joint estimation of location and a balancing parameter, IEEE Signal Process. Lett. 29 (2022) 2447–2451.
[34] W. Xiong, J. Liang, Z. Wang, H.C. So, Elliptic target positioning based on balancing parameter estimation and augmented lagrange programming neural network, Digital Signal Process. 136 (104004) (2023).
[35] K. Panwar, P. Babu, Robust multistatic target localization in the presence of NLOS errors and outliers, IEEE Signal Process. Lett. 29 (2022) 2632–2636.
[36] Z. Shi, H. Wang, C.S. Leung, H.C. So, Robust MIMO radar target localization based on lagrange programming neural network, Signal Process. 174 (2020) 107574.
[37] Z. Yu, J. Li, Q. Guo, T. Sun, Message passing based robust target localization in distributed MIMO radars in the presence of outliers, IEEE Signal Process. Lett. 27 (2020) 2168–2172.
[38] J. Liang, D. Wang, L. Su, B. Chen, H. Chen, H.C. So, Robust MIMO radar target localization via nonconvex optimization, Signal Process. 122 (2016) 33–38.
[39] S. Zhang, A.G. Constantinides, Lagrange programming neural networks, IEEE Trans. Circuits Syst. II: Anal. Digit. Signal Process. 39 (7) (1992) 441–452.
[40] S.M. Kay, Fundamentals of statistical signal processing, in: Estimation Theory, vol. 2, Prentice-Hall, Cliffs, NJ, USA, 1998.
[41] K.W.K. Lui, W.-K. Ma, H.C. So, F.K.W. Chan, Semi-definite programming algorithms for sensor network node

localization with uncertainties in anchor positions and/or propagation speed, IEEE Trans. Signal Process. 57 (2) (2009) 752–763.
[42] Z.-Q. Luo, W.-K. Ma, A.M.-C. So, Y. Ye, S. Zhang, Semidefinite relaxation of quadratic optimization problems, IEEE Signal Process. Mag. 27 (3) (2010) 20–34.
[43] K.W.K. Lui, F.K.W. Chan, H.C. So, Semidefinite programming approach for range-difference based source localization, IEEE Trans. Signal Process. 57 (4) (2009) 1630–1633.
[44] G. Wang, W. Zhu, N. Ansari, Robust TDOA-based localization for IoT via joint source position and NLOS error estimation, IEEE Internet Things J. 6 (5) (2019) 8529–8541.
[45] P. Biswas, T.-C. Lian, T.-C. Wang, Y. Ye, Semidefinite programming based algorithms for sensor network localization, ACM Trans. Sens. Netw. 2 (2) (2006) 188–220.
[46] M. Grant, S. Boyd, CVX: MATLAB software for disciplined convex programming, version 2.1, 2021. Accessed: Sep. 11,. [Online]. Available: http://cvxr.com/cvx.
[47] M.C. Grant, S.P. Boyd, The CVX users' guide, in: Release 2.2, CVX Res. Inc. Tech. Rep., 2020. [Online]. Available: http://cvxr.com/cvx/doc/CVX.pdf.
[48] W. Xiong, C. Schindelhauer, H.C. So, J. Bordoy, A. Gabbrielli, J. Liang, TDOA-based localization with NLOS mitigation via robust model transformation and neurodynamic optimization, Signal Process. 178 (2021a) 107774.
[49] W. Xiong, C. Schindelhauer, H.C. So, Z. Wang, Maximum correntropy criterion for robust TOA-based localization in NLOS environments, Circuits Syst. Signal Process. 40 (12) (2021b) 6325–6339.
[50] A. Zaeemzadeh, M. Joneidi, B. Shahrasbi, N. Rahnavard, Robust target localization based on squared range iterative reweighted least squares, in: Proc. IEEE 14th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS), 2017, pp. 380–388. Orlando, FL.
[51] X. Sun, Y. Zhuang, J. Huai, L. Hua, D. Chen, Y. Li, Y. Cao, R. Chen, RSS-based visible light positioning using nonlinear optimization, IEEE Internet Things J. 9 (15) (2022) 14137–14150.
[52] L.F. Shampine, M.W. Reichelt, The MATLAB ODE suite, SIAM J. Sci. Comput. 18 (1) (1997) 1–22.
[53] D. Reynolds, Gaussian mixture models, in: Encyclopedia of Biometrics, Springer, Berlin, Germany, 2009, pp. 827–832.
[54] C. Knapp, G. Carter, The generalized correlation method for estimation of time delay, IEEE Trans. Acoust., Speech, Signal Process. ASSP–24 (4) (1976) 320–327.