Chapter 3

Penalty Convex-Concave Procedure for Source Localization

In connection with the problem of localizing a single radiating source based on range measurements, in this chapter we explore special structure of the cost function of an unconstrained least squares (LS) formulation and show that it is well suited in a setting known as difference-of-convex-functions (DC) programming. In the literature, the DC programming is sometimes referred to as convex-concave procedure. Our focus in this chapter will be placed on the localization problem based on range measurements. We present an algorithm for solving the LS problem at hand based on a penalty convex-concave procedure (PCCP) [33] that accommodates infeasible initial points in solving a fairly large class of *nonconvex* constrained problems. Algorithmic details are provided to show that the PCCP-based formulation is tailored to the localization problem at hand. These include additional constraints that enforce the algorithms iteration path towards the LS solution, and several strategies to secure good initial points. Numerical results are presented to demonstrate that the proposed algorithm offers substantial performance improvement relative to some best known results from the literature.

3.1 Problem Statement and Review of Related Work

Typically non-survey based localization techniques compute the location estimates in two steps: range/angle estimation and tri-lateration/angulation [44]. In general, the

range estimates can be based on different types of measurements, e.g. received signal strength (RSS), or time of arrival (TOA). This chapter will focus on the problem of range-based localization given the TOA information. In the TOA method, the one-way propagation time of the signal traveling between radiating source and the sensor node is measured. Each TOA measurement then provides a circle centered at the sensor node on which the source of the signal must lie. With three or more sensor nodes the measurements are converted into a set of circular equations that, with knowledge of the geometry of the sensor network, allow to determine the unknown source position [44]. The accuracy of the positioning depends on the quality of the range measurements, network geometry, and the performace of the localization algorithm. In real-world situations, multipath and non line-of-sight (NLOS) propagation are two major sources of error, which can introduce large biases in the TOA measurements and result in unreliable position esitmation [24]. In fact, mitigation of the impairments due to multipath and/or NLOS is another key research topic in wireless location and recent works in this area have reported some promising results [38]. Ultra-wideband (UWB) technology has the potential to deliver very accurate range measurements, thus enabling accurate positioning [37, 38, 39]. As a result, we assume that the multipath and NLOS errors in the TOA measurements have been successfully mitigated.

The source localization problem discussed in this chapter involves a given array of m sensors placed in the $n = 2$ or 3 dimentional space with coordinates specified by $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m, \boldsymbol{a}_i \in R^n\}$. Each sensor measures its distance to a radiating source $\boldsymbol{x} \in R^n$. Throughout it is assumed that only noisy copies of the distance data are available, hence the *range measurements* obey the model

$$r_i = \|\boldsymbol{x} - \boldsymbol{a}_i\| + \varepsilon_i, \quad i = 1, \ldots, m. \tag{3.1}$$

where ε_i denotes the unknown noise that has occurred when the ith sensor measures its distance to source $\boldsymbol{x}$. Let $\boldsymbol{r} = [r_1\ r_2 \ldots r_m]^T$ and $\boldsymbol{\varepsilon} = [\varepsilon_1\ \varepsilon_2 \ldots \varepsilon_m]^T$. The source localization problem can be stated as to estimate the exact source location $\boldsymbol{x}$ from noisy range measurements $\boldsymbol{r}$.

The nonlinear least squares (NLLS) estimate refers to the solution of the problem

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad F(\boldsymbol{x}) = \sum_{i=1}^{m} (r_i - \|\boldsymbol{x} - \boldsymbol{a}_i\|)^2 \tag{3.2}$$

If ranging errors ε_i are i.i.d. random variables that follow Gaussian distribution with zero mean and covariance matrix proportional to the identity matrix, then the NLLS estimate becomes identical to the maximum likelihood estimate. The NLLS formulation is also geometrically meaningful and has been often used as a benchmark to compare new algorithms [15, 37].

In Sec.2.1 of the thesis, it is demonstrated that these problems are hard to solve globally. Many relaxation and approximation methods were developed that offer either lower computational complexity, robustness against positive bias in the distance estimates due to non line-of-site situations, or better performance compared to standard unconstrain optimization methods applied to the NLLS problem. Convex relaxation of a nonconvex problem in (3.2) to an SDP problem and solution methods for *squared* range LS problems are discussed in detail in [47] and Sec.2.1 and of the thesis. Another localization approach that has received considerable interest applies classical multidimensional scaling (MDS) algorithm or its variants to the problem at hand [24, 25, 26, 27].

Multidimensional scaling is a field of study concerning the search of points in a low dimensional space that represent the objects of interest and the pairwise distances between the points (objects) (as measure of dissimilarities) match a set of given values. As such MDS has been an attractive technique for analyzing experimental data in physical, biological, and behavioral science [24]. The classical MDS is a subset of MDS techniques where the relative coordinates of points are determined given only their pairwise Euclidean distances.

When applied to the localization problem at hand, classical MDS starts with constructing a multidimentional similarity matrix. Let $\boldsymbol{X}$ denote an m x n distance matrix

$$\boldsymbol{X} = \begin{pmatrix} (\boldsymbol{x} - \boldsymbol{a}_1)^T \\ (\boldsymbol{x} - \boldsymbol{a}_2)^T \\ \vdots \\ (\boldsymbol{x} - \boldsymbol{a}_m)^T \end{pmatrix}$$

The multidimentional similarity matrix is then defined by $\boldsymbol{D} = \boldsymbol{X}\boldsymbol{X}^T$ which can also be expressed in terms of pairwise distances between sensor nodes and error-free range measurements. Since the exact distances between the sourse $\boldsymbol{x}$ and sensors are not available, the noisy range measurements r_i are used to construct an approximation

of $\boldsymbol{D}$ as

$$\hat{\boldsymbol{D}} = \frac{1}{2}\begin{pmatrix} 2r_1^2 & r_1^2 + r_2^2 - \|\boldsymbol{a}_1 - \boldsymbol{a}_2\|^2 & \dots & r_1^2 + r_m^2 - \|\boldsymbol{a}_1 - \boldsymbol{a}_m\|^2 \\ r_1^2 + r_2^2 - \|\boldsymbol{a}_1 - \boldsymbol{a}_2\|^2 & 2r_2^2 & \dots & r_2^2 + r_m^2 - \|\boldsymbol{a}_2 - \boldsymbol{a}_m\|^2 \\ \vdots & \vdots & \ddots & \vdots \\ r_1^2 + r_m^2 - \|\boldsymbol{a}_1 - \boldsymbol{a}_m\|^2 & r_2^2 + r_m^2 - \|\boldsymbol{a}_1 - \boldsymbol{a}_m\|^2 & \dots & 2r_m^2 \end{pmatrix}$$

Because matrix $\hat{\boldsymbol{D}}$ is symmetric, it admits the orthogonal eigen-decomposition

$$\hat{\boldsymbol{D}} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$$

where $\boldsymbol{\Lambda} = \text{diag}\,(\lambda_1, \lambda_2, \dots, \lambda_m)$ is the diagonal matrix of eigenvalues of $\hat{\boldsymbol{D}}$ with $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m \geq 0$, and $\boldsymbol{U} = [\boldsymbol{u}_1 \quad \boldsymbol{u}_2 \dots \boldsymbol{u}_m]$ is an orthonormatl matrix whose columns are the corresponding eigenvectors. Since the rank of the ideal $\boldsymbol{D}$ is 2, an LS estimate of $\boldsymbol{X}$, denoted by $\boldsymbol{X}_r$, can be computed up to an arbitrary rotation as a solution to the following problem [24]

$$\boldsymbol{X}_r = \arg\min_{\tilde{\boldsymbol{X}}} \|\hat{\boldsymbol{D}} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{X}}^T\|_F^2 = \boldsymbol{U}_s\boldsymbol{\Lambda}_s^{(1/2)}$$

where $\tilde{\boldsymbol{X}}$ is the variable matrix for $\boldsymbol{X}$, $\|.\|_F$ represents the Frobenius norm, $\boldsymbol{U}_s = [\boldsymbol{u}_1 \quad \boldsymbol{u}_2]$ corresponds to the signal subspase, and $\boldsymbol{\Lambda}^{(1/2)} = \text{diag}(\lambda_1^{(1/2)}, \lambda_2^{(1/2)})$. In practical situations of nonzero range errors, the relationship between $\boldsymbol{X}_r$ and $\boldsymbol{X}$ is then

$$\boldsymbol{X} \approx \boldsymbol{X}_r\boldsymbol{\Omega}$$

where $\boldsymbol{\Omega}$ is an unknown rotation matrix to be determined. The estimate of the unknown rotation matrix $\boldsymbol{\Omega}$ and source location $\boldsymbol{x}$ can be obtained by solving an overdetermined system of linear equations [24]. In the absence of noise the symmetric $\hat{\boldsymbol{D}}$ is identical to $\boldsymbol{D}$, is positive semi-definite, and has a rank of 2. In the practical situations of nonzero range errors, $\hat{\boldsymbol{D}}$ will have a full rank.

Other methods based on MDS include a generalized subspace approach by So and Chan [27], that performs position estimation based on the noise subspace. A subspase-based weighting Lagrangian multiplier estimator [25] reduces computational complexity by avoiding the process of eigendecomposition or inverse computation, but it requires some a priori knowledge about noise statistic to construct the weighting matrix. On the other hand, the distributed weighted-multidimentional scaling

(DW-MDS) [26] adds a penalty term to the standard MDS objective function which accounts for prior knowledge about node locations. Although these methods work well in general and can be efficient in terms of complexity, they are found to produce poor performance in certain sensor deployments [37].

In this chapter, we focus on the least squares formulation for the localization problem, where the l_2-norm of the residual errors is minimized in a setting known as difference-of-convex-functions programming. The problem at hand is then solved by applying a penalty convex-concave procedure (PCCP) in a succesive manner [48].

3.2 Fitting the Localization Problem into a CCP Framework

3.2.1 Basic Convex-Concave Procedure

The CCP refers to an effective heuristic method to deal with a class of difference of convex (DC) programming problems, which assume the form

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) - g(\boldsymbol{x}) && \text{(3.3a)} \\ \text{subject to:} \quad & f_i(\boldsymbol{x}) \leq g_i(\boldsymbol{x}) \quad \text{for: } i = 1, 2, \ldots, m && \text{(3.3b)} \end{aligned}$$

where $f(\boldsymbol{x}), g(\boldsymbol{x}), f_i(\boldsymbol{x}), g_i(\boldsymbol{x})$ for $i = 1, 2, \ldots, m$ are convex. A DC program is not convex unless the functions g and g_i are affine, and therefore is hard to solve in general. The class of DC functions is very broad. For example, any C^2 function can be expressed as a difference of convex functions [41]. Classes of problems that can be expressed as difference of convex programming among others include Boolean linear program, circle packing, circuit layout, and multi-matrix proncipal component analysis, examples of which can be found in [33] and references therein.

The basic CCP algorithm is an iterative procedure including two key steps (in the k-th iteration where iterate $\boldsymbol{x}_k$ is known):

(i) Convexification of the objective function and constraints by replacing $g(\boldsymbol{x})$ and $g_i(\boldsymbol{x})$, respectively, with their affine approximations

$$\hat{g}(\boldsymbol{x}, \boldsymbol{x}_k) = g(\boldsymbol{x}_k) + \bigtriangledown g(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k) \qquad \text{(3.4a)}$$

and

$$\hat{g}_i(\boldsymbol{x}, \boldsymbol{x}_k) = g_i(\boldsymbol{x}_k) + \bigtriangledown g_i(\boldsymbol{x}_k)^T(\boldsymbol{x} - \boldsymbol{x}_k) \quad \text{for: } i = 1, 2, \ldots, m \tag{3.4b}$$

(ii) Solving the convex problem

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) - \hat{g}(\boldsymbol{x}, \boldsymbol{x}_k) && \text{(3.5a)} \\ \text{subject to:} \quad & f_i(\boldsymbol{x}) - \hat{g}_i(\boldsymbol{x}, \boldsymbol{x}_k) \leq 0 && \text{(3.5b)} \\ & \text{for: } i = 1, 2, \ldots, m \end{aligned}$$

Because of the convexity of all the functions involved, it can be shown that the basic CCP is a descent algorithm and the iterates $\boldsymbol{x}_k$ converge to the critical point of the original problem [33]. In fact, the global convergence analysis for CCP has also been studied [34, 35]. Note that functions $g(\boldsymbol{x})$ and $g_i(\boldsymbol{x})$ for $i = 1, 2, \ldots, m$ are required to be convex but not necessarily differentiable. If any of $g(\boldsymbol{x})$ or $g_i(\boldsymbol{x})$ are not differentiable at some point $\tilde{\boldsymbol{x}}$ then the corresponding term $\bigtriangledown g(\tilde{\boldsymbol{x}})$ (or $\bigtriangledown g_i(\tilde{\boldsymbol{x}})$) is replaced by a subgradient of $g(\boldsymbol{x})$ (or $g_i(\boldsymbol{x})$) at point $\tilde{\boldsymbol{x}}$.

Let $\mathcal{D}$ be a nonempty set in R^n. A vector $\boldsymbol{h} \in R^n$ is said to be a subgradient of a convex function $f : \mathcal{D} \to R$ at $\boldsymbol{x} \in \mathcal{D}$ if

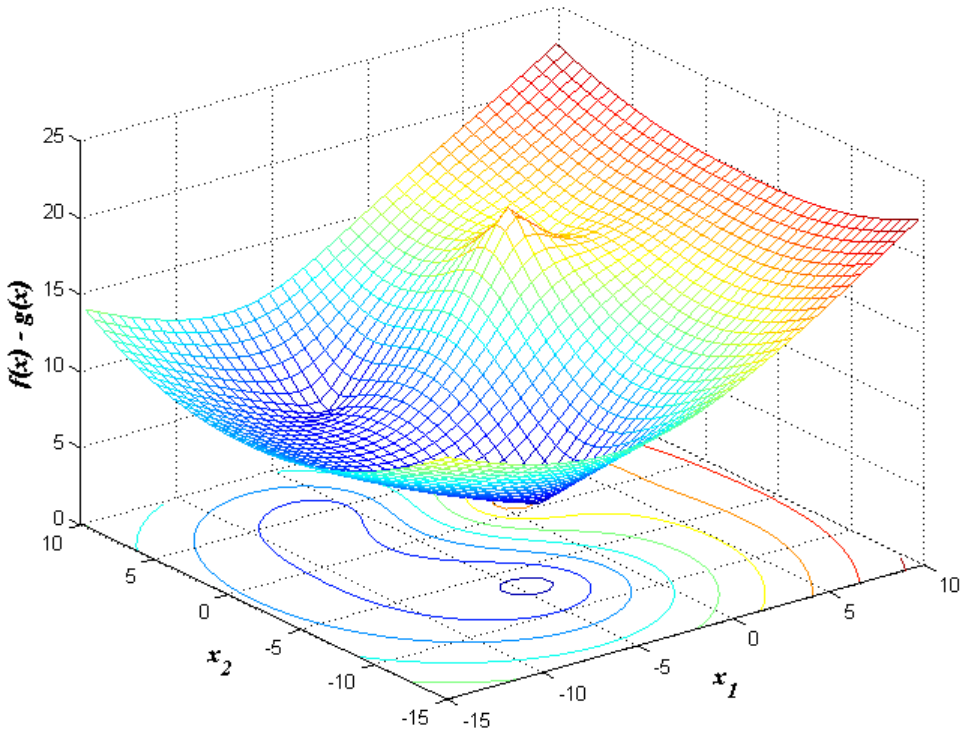
$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{h}^T(\boldsymbol{y} - \boldsymbol{x}) \text{ for all } \boldsymbol{y} \in \mathcal{D}$$

Geometrically, the subgradients at a point $\boldsymbol{x}$ for the case where the convex function $f(x)$ is not differentiable correspond to different tangent lines at $\boldsymbol{x}$ [42].

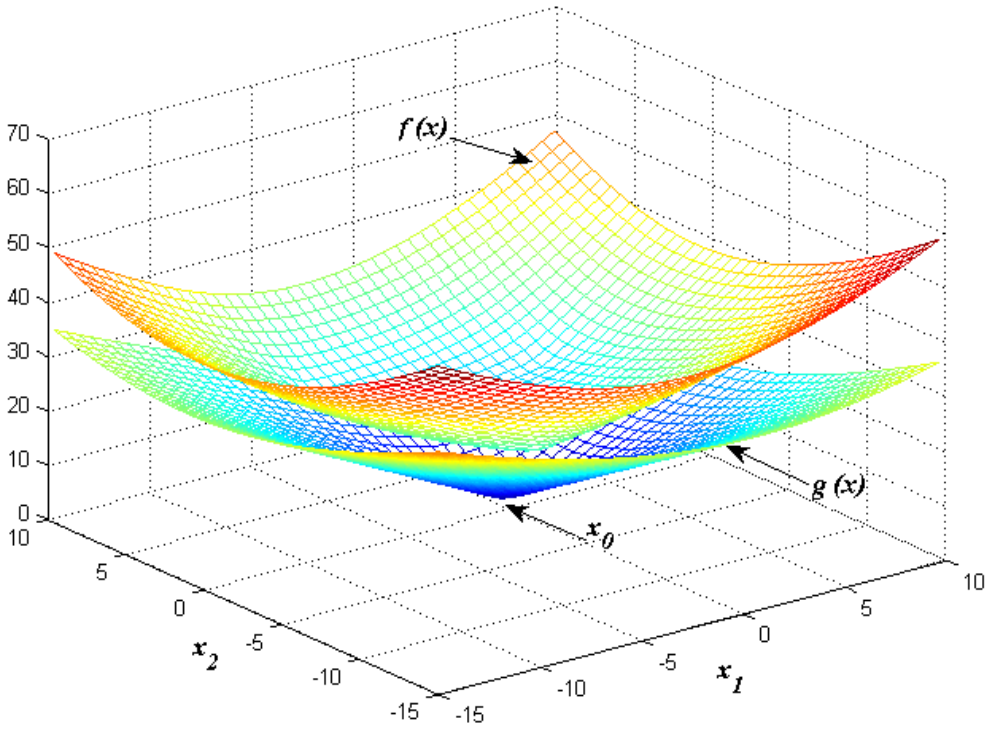
Figure 3.1 shows an example of the CCP approach for an unconstrained DC problem in the form of (3.3a), where $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ are given by:

$$\begin{aligned} f(\boldsymbol{x}) &= \|\boldsymbol{x} - \boldsymbol{a}_1\| + \|\boldsymbol{x} - \boldsymbol{a}_2\| + \|\boldsymbol{x} - \boldsymbol{a}_3\| \\ g(\boldsymbol{x}) &= \|2\boldsymbol{x} - \boldsymbol{a}_4\| \end{aligned}$$

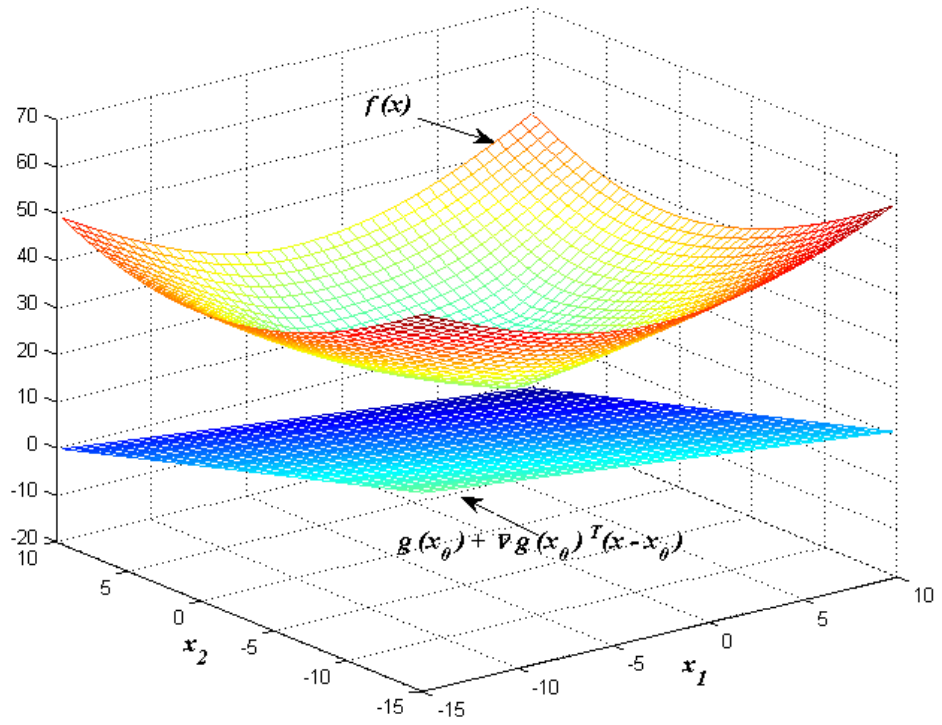
with $\boldsymbol{a}_1 = [3\ 2]^T$, $\boldsymbol{a}_2 = [6\ 5]^T$, $\boldsymbol{a}_3 = [4\ 7]^T$, and $\boldsymbol{a}_4 = [1\ 2]^T$. In this figure, the original nonconvex problem is transferred to a convex problem by replacing a nonconvex part $(-g(\boldsymbol{x}))$ by its affine approximation around the point $\boldsymbol{x}_0 = [0\ 0]^T$.



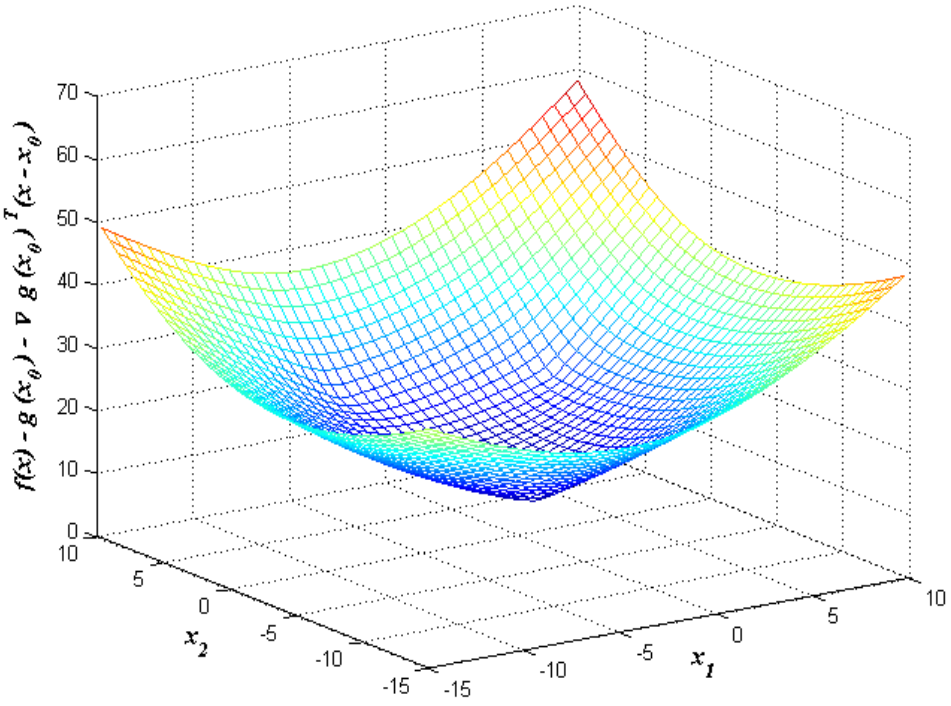
(a) A nonconvex function in the form of the difference of two convex functions and its contour plot.



(b) Separation of the nonconvex function into two convex functions $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$.



(c) First order approximation of $g(\boldsymbol{x})$.



(d) A convex approximation of the original nonconvex function at $\boldsymbol{x}_0 = [0\ 0]^T$.

Figure 3.1: An example of the CCP procedure (re-generated based on [36]).

The basic CCP requires a *feasible* initial point $\boldsymbol{x}_0$ (in the sense that $\boldsymbol{x}_0$ satisfies (3.5b) for $i = 1, 2, \ldots, m$) to start the procedure. By introducing additional slack variables, a penalty CCP has been adopted to accept infeasible initial points. In what follows, we reformulate our localization problem to fit it into the basic CCP framework. Bounds on squared measurement errors as well as penalty terms are then imposed, and a PCCP-based algorithm is developed for solving the problem.

3.2.2 Problem Reformulation

We begin by re-writing the NLLS objective function in (3.2) up to a constant as

$$F(\boldsymbol{x}) = m\boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\sum_{i=1}^{m}\boldsymbol{a}_i - 2\sum_{i=1}^{m} r_i\|\boldsymbol{x} - \boldsymbol{a}_i\| \tag{3.6}$$

The objective in (3.6) is not convex. This is because, for points $\boldsymbol{x}$ that are not coincided with $\boldsymbol{a}_i$ for $1 \leq i \leq m$, the Hessian of $F(\boldsymbol{x})$ is given by

$$\bigtriangledown^2 F(\boldsymbol{x}) = 2m\boldsymbol{I} + 2\sum_{i=1}^{m}\frac{r_i}{\|\boldsymbol{x} - \boldsymbol{a}_i\|^3}\cdot\Big((\boldsymbol{x} - \boldsymbol{a}_i)(\boldsymbol{x} - \boldsymbol{a}_i)^T - \|\boldsymbol{x} - \boldsymbol{a}_i\|^2\boldsymbol{I}\Big)$$

which is obviously not always positive semidefinite. On the other hand, by defining

$$\begin{aligned} f(\boldsymbol{x}) &= m\boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\sum_{i=1}^{m}\boldsymbol{a}_i \\ g(\boldsymbol{x}) &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x} - \boldsymbol{a}_i\| \end{aligned} \tag{3.7}$$

the objective in (3.6) can be expressed as

$$F(\boldsymbol{x}) = f(\boldsymbol{x}) - g(\boldsymbol{x})$$

with both $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ convex, hence it fits naturally into (3.3). Note that $g(\boldsymbol{x})$ in (3.7) is not differentiable at the point where $\boldsymbol{x} = \boldsymbol{a}_i$ for some $1 \leq i \leq m$, thus we replace the term $\bigtriangledown g(\boldsymbol{x}_k)$ in (3.4b) by a subgradient [43] of $g(\boldsymbol{x})$ at $\boldsymbol{x}_k$, denoted by $\partial g(\boldsymbol{x}_k)$, as

$$\partial g(\boldsymbol{x}_k) = 2\sum_{i=1}^{m} r_i\partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\|$$

where

$$\partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| = \begin{cases} \dfrac{\boldsymbol{x}_k - \boldsymbol{a}_i}{\|\boldsymbol{x}_k - \boldsymbol{a}_i\|}, & \text{if } \boldsymbol{x}_k \neq \boldsymbol{a}_i \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

Hence $\hat{g}(\boldsymbol{x}, \boldsymbol{x}_k)$ in (3.4b) is given by

$$\begin{aligned} \hat{g}(\boldsymbol{x}, \boldsymbol{x}_k) &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| + 2\left(\boldsymbol{x} - \boldsymbol{x}_k\right)^T \sum_{i=1}^{m} r_i \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| \\ &= 2\boldsymbol{x}^T \sum_{i=1}^{m} r_i \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| + c \end{aligned}$$

where c is a constant given by

$$\begin{aligned} c &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\boldsymbol{x}_k^T \sum_{i=1}^{m} r_i \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| \\ &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\sum_{i=1}^{m} r_i \boldsymbol{x}_k^T \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| \\ &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\sum_{i=1}^{m} r_i \left(\boldsymbol{x}_k^T - \boldsymbol{a}_i + \boldsymbol{a}_i\right)^T \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| \\ &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\sum_{i=1}^{m} r_i \left(\boldsymbol{x}_k^T - \boldsymbol{a}_i\right)^T \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\sum_{i=1}^{m} r_i \boldsymbol{a}_i^T \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| \\ &= 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\sum_{i=1}^{m} r_i\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - 2\sum_{i=1}^{m} r_i \boldsymbol{a}_i^T \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| \\ &= -2\sum_{i=1}^{m} r_i \boldsymbol{a}_i^T \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\|. \end{aligned}$$

The convex approximation of the objective in (3.6) can now be derived as

$$\begin{aligned} \hat{F}(\boldsymbol{x}) &= f(\boldsymbol{x}) - \hat{g}(\boldsymbol{x}, \boldsymbol{x}_k) \\ &= m\boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T \sum_{i=1}^{m} \boldsymbol{a}_i - 2\boldsymbol{x}^T \sum_{i=1}^{m} r_i \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\| + c \end{aligned}$$

It follows that, up to a multiplicative factor $1/m$ and an additive constant term, the convex objective function in (3.5b) can be written as

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \hat{F}(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{v}_k \tag{3.8}$$

where

$$\boldsymbol{v}_k = \bar{\boldsymbol{a}} + \frac{1}{m}\sum_{i=1}^{m} r_i \partial \|\boldsymbol{x}_k - \boldsymbol{a}_i\|, \quad \bar{\boldsymbol{a}} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{a}_i \tag{3.9}$$

It is rather straightforward to see that given $\boldsymbol{x}_k$ (in the k-th iteration) the solution of the quadratic problem (3.8) can be obtained as

$$\boldsymbol{x}_{k+1} = \bar{\boldsymbol{a}} + \frac{1}{m}\sum_{i=1}^{m} r_i \partial \|\boldsymbol{x}_k - \boldsymbol{a}_i\|. \tag{3.10}$$

3.2.3 Imposing Error Bounds and Penalty Terms

The algorithm being developed can be enhanced by imposing a bound on each squared measurement error, namely

$$\left(\|\boldsymbol{x} - \boldsymbol{a}_i\| - r_i\right)^2 \leq \delta_i^2 \tag{3.11}$$

which leads to

$$\|\boldsymbol{x} - \boldsymbol{a}_i\| - r_i - \delta_i \leq 0 \tag{3.12a}$$

$$r_i - \delta_i \leq \|\boldsymbol{x} - \boldsymbol{a}_i\| \tag{3.12b}$$

for $1 \leq i \leq m$. Placing such bounds means that as iterations proceed the new iterates (coordinates of possible source locations) are restricted to lie within a physically meaningfull region determined by parameters δ_i.

Note that the constraints in (3.12a) are convex and fit into the form of basic CCP in (3.5b) with $f_i(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{a}_i\| - r_i - \delta_i$ and $g_i(\boldsymbol{x}) = 0$, while those in (3.12b) are in the form of (3.3) with $f_i(\boldsymbol{x}) = r_i - \delta_i$ and $g_i(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{a}_i\|$. Following CCP (see (3.4b)), $g_i(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{a}_i\|$ is linearized around iterate $\boldsymbol{x}_k$ to

$$\hat{g}_i(\boldsymbol{x}, \boldsymbol{x}_k) = \|\boldsymbol{x}_k - \boldsymbol{a}_i\| + \partial \|\boldsymbol{x}_k - \boldsymbol{a}_i\|^T (\boldsymbol{x} - \boldsymbol{x}_k)$$

and (3.12b) is convexified as

$$r_i - \delta_i \leq \|\boldsymbol{x}_k - \boldsymbol{a}_i\| + \partial \|\boldsymbol{x}_k - \boldsymbol{a}_i\|^T (\boldsymbol{x} - \boldsymbol{x}_k)$$

which now fits into (3.5b), or equivalently

$$-\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\|^T (\boldsymbol{x} - \boldsymbol{x}_k) + r_i - \delta_i \leq 0 \tag{3.13}$$

We remark that constraint (3.13) is not only convex but also tighter than (3.12b). As a matter of fact, the convexity of the norm $\|\boldsymbol{x} - \boldsymbol{a}_i\|$ implies that it obeys the property

$$\|\boldsymbol{x} - \boldsymbol{a}_i\| \geq \|\boldsymbol{x}_k - \boldsymbol{a}_i\| + \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\|^T (\boldsymbol{x} - \boldsymbol{x}_k)$$

Therefore, a point $\boldsymbol{x}$ satisfying (3.13) automatically satisfies (3.12b). Summarizing, the convexified problem in the k-th iteration can be stated as

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{v}_k && \text{(3.14a)} \\ \text{subject to:} \quad & \|\boldsymbol{x} - \boldsymbol{a}_i\| - r_i - \delta_i \leq 0 && \text{(3.14b)} \end{aligned}$$

$$-\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - \partial\|\boldsymbol{x}_k - \boldsymbol{a}_i\|^T (\boldsymbol{x} - \boldsymbol{x}_k) + r_i - \delta_i \leq 0 \tag{3.14c}$$

A technical problem making the formulation in (3.14) difficult to implement is that it requires a feasible initial point $\boldsymbol{x}_0$. The problem can be overcome by introducing nonnegative slack variables $s_i \geq 0, \hat{s}_i \geq 0$, for $i = 1, \ldots, m$ into the constraints in (3.14b) and (3.14c) to replace their right-hand sides (which are zeros) by relaxed upper bounds (as these new bounds themselves are nonnegative variables). This leads to a *penalty* CCP (PCCP) based formulation as follows:

$$\begin{aligned} \underset{\boldsymbol{x},\boldsymbol{s},\hat{\boldsymbol{s}}}{\text{minimize}} \quad & \boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{v}_k + \tau_k \sum_{i=1}^{m} (s_i + \hat{s}_i) && \text{(3.15a)} \\ \text{subject to:} \quad & \|\boldsymbol{x} - \boldsymbol{a}_i\| - r_i - \delta_i \leq s_i && \text{(3.15b)} \end{aligned}$$

$$-\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - \frac{(\boldsymbol{x}_k - \boldsymbol{a}_i)^T}{\|\boldsymbol{x}_k - \boldsymbol{a}_i\|} (\boldsymbol{x} - \boldsymbol{x}_k) + r_i - \delta_i \leq \hat{s}_i \tag{3.15c}$$

$$s_i \geq 0, \hat{s}_i \geq 0, \text{ for: } i = 1, 2, \ldots, m \tag{3.15d}$$

where the weight $\tau_k \geq 0$ increases as iterations proceed until it reaches an upper limit τ_{max}. By using a monotonically increasing τ_k for the penalty term in (3.15a), the algorithm reduces the slack variables s_i and $\hat{s}_i$ very quickly. As a result, new iterates quickly become feasible as s_i and $\hat{s}_i$ vanish. The upper limit τ_{max} is imposed to avoid numerical difficulties that may occur if τ_k becomes too large and to ensure convergence

if a feasible region is not found [9]. Consequently, while formulation (3.15) accepts *infeasible* initial points, the iterates obtained by solving (16) are practically identical to those obtained by solving (3.14).

3.2.4 The Algorithm

The input parameters for the algorithm include the bounds δ_i on the measurement error. Setting δ_i to a lower value leads to a "tighter" solution. On the other hand, a larger δ_i would make the algorithm less sensitive to outliers. However, some a priori knowledge about noise statistic, if available, or sensor geometry can be used to derive reasonable values for δ_i. For example, if measurement noise $\boldsymbol{\varepsilon}$ obeys a Gaussian distribution with zero mean and known covariance $\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \ldots, \sigma_m^2)$, then δ_i can be expressed as $\delta_i = \gamma\sigma_i$, where γ is a parameter that determines the width of the confidence interval. For example, for $\gamma = 3$ we have the probability $Pr\{|\varepsilon_i| \leq 3\sigma_i\} \approx 0.99$. Other input parameters are initial point $\boldsymbol{x}_0$, maximum number of iterations K_{max}, initial weight τ_0, and upper limit of weight τ_{max} (to avoid numerical problems that may occur if τ_i becomes too large).

As mentioned in Sec. 3.2 of the thesis, the original LS objective is highly nonconvex with many local minimums even for small-scale systems. Consequently, it is of critical importance to select a good initial point for the proposed PCCP-based algorithm because PCCP is essentially a local procedure. Several techniques are available, these include:

(i) Select the initial point uniformly randomly over the same region as the unknown radiating source;

(ii) Set the initial point to the origin;

(iii) Run the algorithm from a set of candidate initial points and identify the solution as the one with lowest LS error. Typically, comparing the results from n distinct initial points shall suffice. For the planar case ($n = 2$), for example, it is sufficient to compare the two intersection points of the two circles that are associated with the two smallest distance readings as the target is very likely to be in the vicinity of these sensors;

(iv) Apply a global localization algorithm such as those in [15] to generate an approximate LS solution, then take it as the initial point to run the proposed

algorithm.

The algorithm can be now outlined as follows.

Algorithm 3. PCCP-based LS Algorithm for Source Localization

1) Input data: Sensor locations $\{\boldsymbol{a}_i, i = 1, \ldots, m\}$, range measurements $\{r_i, i = 1, \ldots, m\}$, initial point $\boldsymbol{x}_0$, maximum number of iterations K_{max}, initial weight τ_0 and upper limit of weight τ_{max}, weight increment $\mu > 0$, error bounds δ_i. Set iteration count to $k = 0$.

2) Form $\boldsymbol{v}_k$ as

$$\boldsymbol{v}_k = \bar{\boldsymbol{a}} + \frac{1}{m}\sum_{i=1}^{m} r_i \partial \|\boldsymbol{x}_k - \boldsymbol{a}_i\|, \quad \bar{\boldsymbol{a}} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{a}_i$$

and solve

$$\begin{aligned} \underset{\boldsymbol{x},\boldsymbol{s},\hat{\boldsymbol{s}}}{\text{minimize}} \quad & \boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{v}_k + \tau_k \sum_{i=1}^{m}(s_i + \hat{s}_i) \\ \text{subject to:} \quad & \|\boldsymbol{x} - \boldsymbol{a}_i\| - r_i - \delta_i \le s_i \\ & -\|\boldsymbol{x}_k - \boldsymbol{a}_i\| - \frac{(\boldsymbol{x}_k - \boldsymbol{a}_i)^T}{\|\boldsymbol{x}_k - \boldsymbol{a}_i\|}(\boldsymbol{x} - \boldsymbol{x}_k) + r_i - \delta_i \le \hat{s}_i \\ & s_i \ge 0, \hat{s}_i \ge 0, \text{ for: } i = 1, 2, \ldots, m \end{aligned}$$

Denote the solution as $(\boldsymbol{s}^*, \hat{\boldsymbol{s}}^*, \boldsymbol{x}^*)$.

3) Update $\tau_{k+1} = \min(\mu\tau_k, \tau_{max})$, set $k = k + 1$.

4) If $k = K_{max}$, terminate and output $\boldsymbol{x}^*$ as the solution; otherwise, set $\boldsymbol{x}_k = \boldsymbol{x}^*$ and repeat from Step 2.

3.3 Numerical Results

For illustration purposes, the proposed algorithm was applied to a network with five sensors, and its performance was evaluated and compared with existing state-of-the-art methods by Monte Carlo simulations with a set-up similar to that of [15]. SR-LS solutions were used as performance benchmarks for the PCCP-based LS Algorithm. The system consisted of 5 sensors $\{\boldsymbol{a}_i, i = 1, 2, \ldots, 5\}$ randomly placed in the planar region in $[-15; 15] \times [-15; 15]$, and a radiating source $\boldsymbol{x}_s$, located randomly in the region $\{\boldsymbol{x} = [x_1; x_2], -10 \leq x_1, x_2 \leq 10\}$. The coordinates of the source and sensors were generated for each dimension following a uniform distribution. Measurement noise $\{\varepsilon_i, i = 1, \ldots, m\}$ was modelled as independent and identically distributed (i.i.d) random variables with zero mean and variance σ^2, with σ being one of four possible levels $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$. The range measurements $\{r_i, i = 1, 2, \ldots, 5\}$ were calculated using (3.1). Accuracy of source location estimation was evaluated in terms of average of the squared position error error in the form $\|\boldsymbol{x}^* - \boldsymbol{x}_s\|^2$, where $\boldsymbol{x}_s$ denotes the exact source location and $\boldsymbol{x}^*$ is its estimation obtained by SR-LS and PCCP methods, respectively. In our simulations parameter γ was set to 3 and the number of iterations was set to 20. The proposed method was implemented by using CVX [45] and implementation of SR-LS followed [15]. The PCCP algorithm was initialized with intersection points of the two circles that are associated with the two smallest distance readings. A candidate solution point with lowest LS error in (3.2) was chosen as a PCCP solution. In cases when the circles did not intersect due to high noise level, the initial point was set as a midpoint between the centers of the two circles.

Table 3.1 provides comparisons of the PCCP with SR-LS and MLE, where each entry is averaged squared error over 1,000 Monte Carlo runs of the method. The MLE was implemented using Matlab function *lsqnonlin* [46], initialized with the same point as PCCP. It is observed that, comparing with SR-LS, the estimates produced by the proposed algorithm are found to be closer to the true source locations in MSE sense. The last column of the table represents relative improvement of the proposed method over SR-LS solutions in percentage.

Figure 3.2 illustrates the location estimation errors for the SR-LS solution and the proposed algorithm for various numbers of sensors. Each entry is a squared error of the given method averaged over 50 random initializations of the system consisting of m sensors, with m being one of $\{4, 5, 7, 10\}$. The MLE was implemented using Matlab

Table 3.1: Averaged MSE for SR-LS and PCCP methods

σ	MLE	SR - LS	PCCP	R.I.
1e-03	6.0159e-01	1.3394e-06	**9.5243e-07**	29%
1e-02	3.5077e-01	1.4516e-04	**9.5831e-05**	34%
1e-01	3.7866e-01	1.2058e-02	**8.7107e-03**	28%
1e+0	1.4470e+00	1.3662e+00	**1.2346e+00**	10%

function *lsqnonlin*, initialized with the same point as PCCP-based LS. It is observed that at all noise levels the proposed approach performes well compared to SR-LS, especially for a low number of reference nodes. From the figure, it is noted that for the high standard deviation of the noise, as the number of sensor nodes increases, the PCCP-based LS and SR-LS show similar performance.

Finally, we study the convergence of the PCCP-based LS Algorithm. As an example, consider an instance of the source localization problem on the plane ($n = 2$) as discussed in Sec. 2.1 of the thesis. The system consists of five sensors ($m = 5$) located at $(6, 4)^T, (0, -10)^T, (5, -3)^T, (1, -4)^T$ and $(3, -3)^T$ with the source emmitting the signal at $\boldsymbol{x}_s = (-2, 3)^T$. Figure 3.3 shows the iteration path the PCCP-based LS follows when initialized with a randomly selected point $\boldsymbol{x}_0 = (-20, -10)^T$. It took 8 iterations for the PCCP-based LS to converge to a point $\boldsymbol{x}^* = (-1.9914, 3.1632)^T$ which is a good estimate of the R-LS approximation of the exact source location $\boldsymbol{x}_s$.

Figure 3.4 depicts the convergence speed of the proposed algorithm for 50 random initializations of the system with a fixed number of sensors (m). In the simulations, we set $\sigma = 10^{-1}$ and consider systems with 4, 5, 7 and 10 sensors randomly placed in the planar region in $[-15; 15] \times [-15; 15]$. For every estimate of the PCCP-based LS, we compute the value of the objective in (3.6). It is observed that PCCP-based LS converges fairly fast, in most cases after 10 iterations. For systems with large numbers of sensor nodes ($m = 10$) PCCP-based LS converges approximately in three sequential updates.

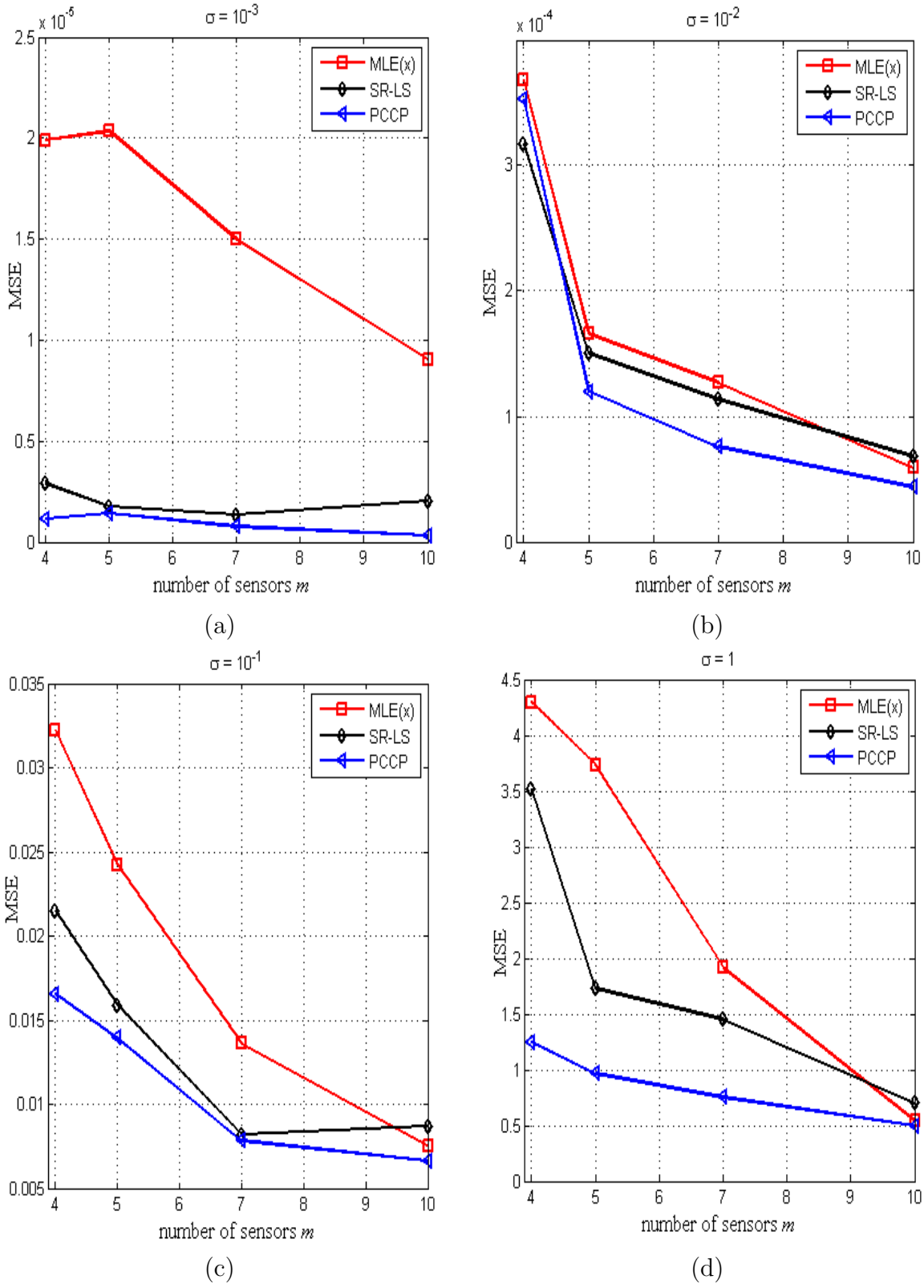


Figure 3.2: MSE for different methods, various number of sensor nodes m and different noise levels with (a) $\sigma = 10^{-3}$, (b) $\sigma = 10^{-2}$, (c) $\sigma = 10^{-1}$, and (d) $\sigma = 1$.

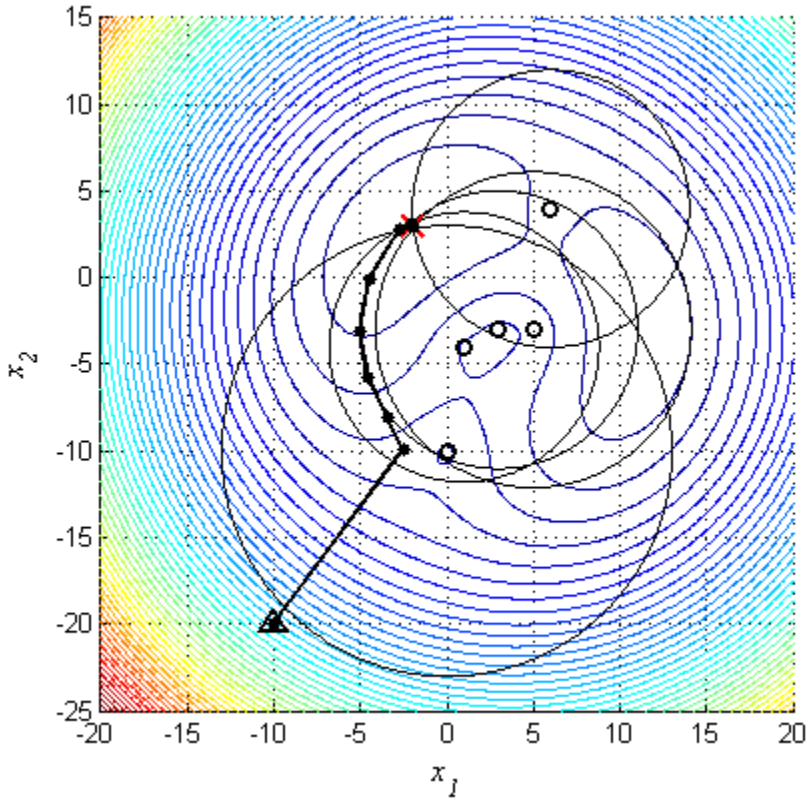


Figure 3.3: Iteration path of the PCCP-based LS Algorithm and contours of the R-LS objective function over the region $\Re = \{\boldsymbol{x} : -15 \leq x_1 \leq 15, -25 \leq x_2 \leq 15\}$. The red cross indicates the location of the signal source. Sensors are located at $(6,4)^T$, $(0,-10)^T$, $(5,-3)^T$, $(1,-4)^T$ and $(3,-3)^T$. Large circles denote possible source locations given the noisy range reading at a particular sensor.

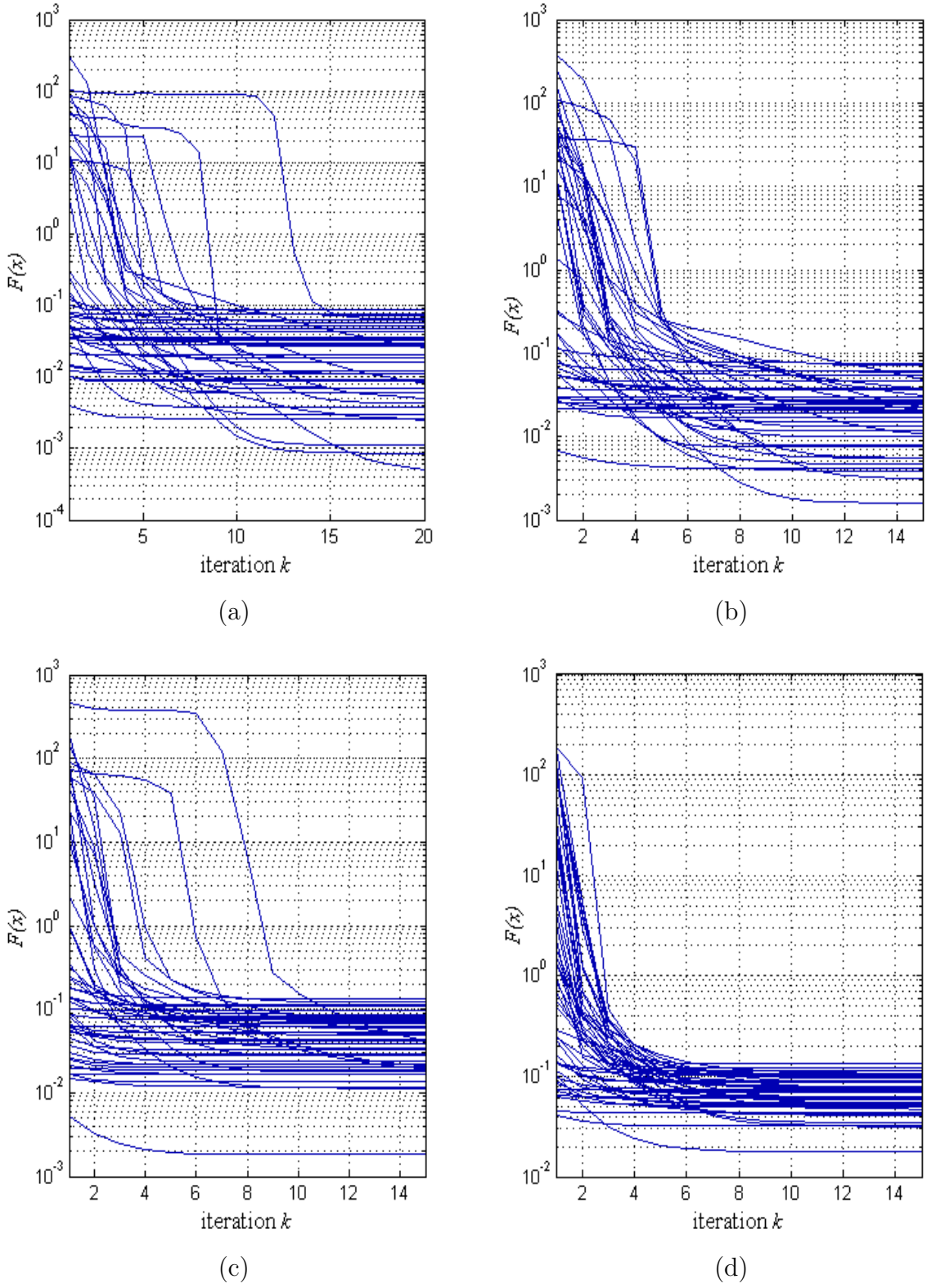


Figure 3.4: Convergernce of the proposed PCCP-based LS for random initializations with $\sigma = 10^{-1}$ for (a) 4 sensor nodes, (b) 5 sensor nodes, (c) 7 sensor nodes, and (d) 10 sensor nodes.

Bibliography

[1] J. O. Smith and J. S. Abel, "Closed-form least-squares source location estimation from range-difference measurements," *IEEE Trans. on Acoustic, Speech Signal Proc.*, vol. 12, pp. 1661–1669, Dec. 1987.

[2] H. Schau and A. Robinson, "Passive source localization employing intersecting spherical surfaces from time-of-arrival differences," *IEEE Trans. on Acoustic, Speech Signal Proc.*, vol. ASSP–35, pp. 1223–1225, Aug. 1987.

[3] K. Yao, R. Hudson, C. Reed, D. Chen, and F. Lorenzelli, "Blind beamforming on a randomly distributed sensor array system," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1555-1567, Oct. 1998.

[4] M. A. Sprito, "On the accuracy of cellular mobile station location estimation," *IEEE Trans. on Veh. Technol.*, vol. 50, pp. 674-685, May 2001.

[5] Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereau, "Realtime passive source localization: A practical linear correction least-squares approach," *IEEE Trans. on Speech Audio Proc.*, vol. 9, no. 8, pp. 943-956, Nov. 2002.

[6] K. W. Cheung, H. C. So, W. K. Ma, and Y. T. Chan, "Least squares algorithms for time-of-arrival-based mobile location," *IEEE Trans. on Signal Proc.*, vol. 52, no. 4, pp. 1121–1228, Apr. 2004.

[7] D. Li and H. Hu, "Energy-Based Collaborative Source Localization Using Acoustic Microsensor Array," in *EURASIP Journal on Applied Signal Proc.*, vol. 4, 321–337, 2003.

[8] X. Sheng and Y.-H. Hu, "ML Multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Trans. on Signal Proc.*, vol. 53, no.1, pp. 44–53, Jan. 2005.

[9] Z. M. Saric, D. D. Kukolj, and N. D. Teslic, "Acoustic source localization in wireless sensor network", *Circuits Syst Signal Proc.*, vol. 29, pp. 837–856, 2010.

[10] L.Lu, H.-C. Wu, K.Yan, and S.Iyengar, "Robust expectation maximization algorithm for multiple wideband acoustic source localization in the presence of nonuniform noise variances," *IEEE Sensors J.*, vol. 11, no. 3, pp. 536–544, Mar. 2011.

[11] K.W. Cheung, W.K. Ma, and H.C. So, "Accurate approximation algorithm for TOA-based maximum-likelihood mobile location using semidefinite programming," in *Proc. ICASSP*, vol. 2, pp. 145–148, 2004.

[12] A. H. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location," *IEEE Signal Pro. Mag.*, vol. 22, no. 4, pp. 24–40, July 2005.

[13] Y. T. Chan, H. Y. C. Hang, and P. C. Ching, "Exact and approximate maximum likelihood localization algorithms," *IEEE Trans. on Veh. Technol.*, vol. 55, no. 1, pp. 10–16, Jan. 2006.

[14] P. Stoica and J. Li, "Source localization from range-difference measurements," *IEEE Signal Proc. Mag.*, vol. 23, pp. 63–65,69, Nov. 2006.

[15] A. Beck, P. Stoica and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Trans. on Signal Proc.*, vol. 56, no. 5, pp. 1770–1777, May 2008.

[16] A. Beck, M. Teboulle, and Z. Chikishev, "Iterative minimization schemes for solving the single source localization problem," *SIAM J. Optim.*, vol. 19, no. 3, pp. 1397–1416, Nov. 2008.

[17] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Comm. Pure Appl. Math.*, vol. 63, pp. 1–38, 2010.

[18] A. Beck, and D. Pan, "On the solution of the GPS localization and circle fitting problems," *SIAM J. Optim.*, vol. 22, no. 1, pp. 108–134, Jan. 2012.

[19] A. Beck, "On the convergence of alternating minimization for convex programming with applications to iteratively re-weighted least squares and decomposition schemes," *SIAM J. Optim.*, vol. 25, no. 1, pp. 185-209, Jan. 2015.

[20] J.J. More, "Generalizations of the trust region subproblem," *Optim. Methods Softw.*, vol. 2, pp. 189–209, 1993.

[21] C. Fortin and H. Wolkowicz, "The trust region subproblem and semidefinite programming," *Optim. Methods Softw.*, vol. 19, no.1, pp. 41–67, 2004.

[22] D.P. O'Leary, "Robust regression computation using iteratively reweighted least squares," *SIAM J. Matrix Anal. Appl.*, vol. 11, no. 3, pp. 466–480, 1990.

[23] N. Bissantz, L. Dümbgen, A. Munk, and B. Stratmann, "Convergence analysis of generalized iteratively reweighted least squares algorithms on convex function spaces," *SIAM J. Optim.*, vol. 19, no. 4, pp 1828–1845, 2009.

[24] K. W. Cheung and H. C. So, "A multidimensional scaling framework for mobile location using time-of-arrival measurements," *IEEE Trans. on Signal Proc.*, vol. 53, no. 2, pp. 460–470, Feb. 2005.

[25] S. Qin, Q. Wan, and L.-F. Duan, "Fast and efficient multidimentional scaling algorithm for mobile positioning," *IET Signal Processing*, vol. 6, no. 9, pp. 857–861, March 2012.

[26] J.A. Costa, N. Patwari, and A. O. Hero, "Distributed weighted-multidimentional scaling for node localization in sensor networks,"ACD Trans. Sens. Netw., vol. 2, no. 1, pp. 39–64, 2006.

[27] H.C. So and F.K.W. Chan, "Generalized Subspace Approach for Mobile Positioning With Time-of-Arrival Measurements," *IEEE Trans. on Signal Proc.*, vol. 55, no. 10, pp. 5103–5107, October 2007.

[28] H. Liu, H. Darabi, P. B, and J. Liu, "Survey of Wireless Indoor Positioning Techniques and Systems," *IEEE Trans. on Systems, Man and Cybernetics. Part C: Applications and Revies*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[29] Y. Liu, Y.H. Hu, and Q. Pan, "Distributed, Robust Acoustic Source Localization in a Wireless Sensor Network," *IEEE Trans. on Signal Proc.*, vol. 60, no. 8, pp. 4350–4359, Aug. 2012.

[30] W. Kim, J. Lee, and G. Jee, "The interior-point method for an optimal treatment of bias in trilateration location," *IEEE Trans. Veh. Technol.*, vol. 55, no. 4, pp. 1291–1301, Jul. 2006.

[31] T. Qiao, S. Redfield, A. Abbasi, Z. Su, and H. Liu, "Robust coarse position estimation for TDOA localization," *IEEE Wireless Commun. Lett.*, vol. 2, no. 6, pp. 623–626, Dec. 2013.

[32] A. L. Yuille and A. Rangarajan, "The concave-convex procedure,' *Neural Computation*, vol. 15, no. 4, pp. 915–936, 2003.

[33] T. Lipp and S. Boyd, "Variations and extensions of the convex-concave procedure," *Research Report*, Stanford University, Aug. 2014.

[34] G. R. Lanckreit and B. K. Sriperumbudur, "On the convergence of the concave-convex procedure," in *Advances in Neural Information Proc. Systems*, pp. 1759–1767, 2009.

[35] Y. Kang, Z. Zhang, and W.-J. Li, "On the global convergence of majorization minimization algorithms for nonconvex optimization problems," in *arXiv: 1504.07791v2*, 2015.

[36] M. R. Gholami, S. Gezici, and E. G. Ström, "TW-TOA based positioning in the presence of clock imperfections," *Digital Signal Processing*, vol. 59, pp. 19–30, 2016.

[37] M. R. Gholami, E. G. Ström, F. Sottile, D. Dardari, A. Conti, S. Gezici, M. Rydström, and M. A. Spirito, "Static positioning using UWB range measurements," in *Future Network and Mobile Summit Conference Proceedings*, pp. 1–10, 2010.

[38] D. Dardari, A.Conti, U.J. Ferner, A. Giorgetti, and M.Z. Win, "Ranging with ultrawide bandwidth signals in multipath environments," *Proceedings of the IEEE*, vol. 97, pp. 427–450, 2009.

[39] M. R. Gholami, E. G. Ström, and M. Rydström, "Indoor sensor node positioning using UWB range measurements," in *17th European Signal Processing Conference (Eusipco)*, pp. 1943–1947, 2009.

[40] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, no. 1, pp. 40–95, Mar. 1996.

[41] P. Hartman, "On functions representable as a difference of convex functions," *Pacific Journal of Math*, vol. 9, no. 3, pp. 707–713, 1959.

[42] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*, Springer, New-York, 2007.

[43] Y. Nesterov, *Introductory Lectures on Convex Optimization*, Kluwer Academic Publishers, Boston, 2004.

[44] C. Gentile, N. Alsindi, R. Raulefs, and C. Teolis, *Geolocation Techniques: Principles and Applications*, Springer, New-York, 2013.

[45] CVX Research, http://cvx.com/cvx, August 2012.

[46] The Mathworks Inc., http://mathworks.com, 2015.

[47] "Iterative re-weighting"

[48] "Penalty Convex-Concave Procedure of Source Localization Problem,"