A Simple Globally Convergent Algorithm for the Nonsmooth Nonconvex Single Source Localization Problem*

D. Russell Luke† Shoham Sabach‡ Marc Teboulle§ Kobi Zatlawey¶

January 25, 2017

Abstract

We study the single source localization problem which consists of minimizing the squared sum of the errors, also known as the maximum likelihood formulation of the problem. The resulting optimization model is not only nonconvex but is also nonsmooth. We first derive a novel equivalent reformulation as a smooth constrained nonconvex minimization problem. The resulting reformulation allows for deriving a delightfully simple algorithm that does not rely on smoothing or convex relaxations. The proposed algorithm is proven to generate bounded iterates which globally converge to critical points of the original objective function of the source localization problem. Numerical examples are presented to demonstrate the performance of our algorithm.

Keywords: Nonsmooth nonconvex minimization, Kurdyka-Łojasiewicz property, method of multipliers, alternating minimization, convergence in semialgebraic optimization, single source localization.

AMS Subject Classification: 90C26, 90C90, 49M37, 65K05.

*Supported by a GIF Grant G-1253-304.6/2014.

†Institut für Numerische und Angewandte Mathematik, Universität Göttingen, 37083 Göttingen, Germany. E-mail: `r.luke@math.uni-goettingen.de`.

‡Department of Industrial Engineering and Management, Technion—Israel Institute of Technology, Haifa 3200003, Israel. E-mail: `ssabach@ie.technion.ac.il`.

§School of Mathematical Sciences, Tel-Aviv University, Ramat-Aviv 69978, Israel. E-mail: `teboulle@post.tau.ac.il`. Partially supported by the Israel Science Foundation, ISF Grant 998/12.

¶School of Mathematical Sciences, Tel-Aviv University, Ramat-Aviv 69978, Israel. E-mail: `kobi.zatlawey@gmail.com`.

1 Introduction

We report on a study of the single source localization problem, which appears in a broad range of important and disparate applications including for instance: mobile communication and wireless networks [11, 20], acoustic/sound source localization [16], GPS localization [4], and brain activity identification [12], to mention just a few. The problem is based on range measurements from an array of sensors (also called anchors) which are actually "data suppliers" for estimating the ranges to the unknown location of the radiating source. The basic scenario of the source localization problem can be described as follows. The sensors send their range measurements data to a control center which estimates the source location according to the received data. There are several types of measurements that could be used in source localization such as time of arrival, signal strength and distance (which in many cases is not known directly and should be estimated).

The source localization problem can be modeled mathematically as a system of nonlinear equations, for which, each equation describes the estimated range between a specific sensor to the source as being the distance between the unknown source and each anchor contaminated with additive noise. Consider a group of m sensors which are denoted by $a_j \in \mathbb{R}^n$, $j = 1, 2, \ldots, m$. Each a_j contains the exact location of the j-th sensor. We denote by $\mathcal{A}$ the set of all sensors, that is, $\mathcal{A} = \{a_1, a_2, \ldots, a_m\}$. Let $x \in \mathbb{R}^n$, be the source which exact location is unknown, and let $d_j > 0$, $j = 1, 2, \ldots, m$, be a noisy observation of the range between the source x and the j-th sensor a_j described by the following equations:

$$d_j = \|x - a_j\| + \varepsilon_j, \quad j = 1, 2, \ldots, m, \tag{1.1}$$

where ε_j is the j-th unknown noise. The problem is then to find an adequate approximation of the unknown source x satisfying (1.1).

Two well-known optimization approaches for finding the location of the source consist of minimizing the least squared error in the squared domain, or by minimizing the squared sum of the errors; see *e.g.*, [5]. The first approach consists of solving the source localization problem via the smooth Squared Least Squares formulation given by:

$$\min_{x \in \mathbb{R}^n} \left\{ \sum_{j=1}^{m} \left(\|x - a_j\|^2 - d_j^2 \right)^2 \right\}. \tag{SLS}$$

The second approach for estimating the source location, and is the focus of our study, consists of minimizing the Least Squares noise error, *i.e.*,

$$\min_{x \in \mathbb{R}^n} \left\{ f(x) := \sum_{j=1}^{m} \left(\|x - a_j\| - d_j \right)^2 \right\}. \tag{LS}$$

Both formulations of the source localization problem are unconstrained nonconvex optimization problems. One remarkable feature of problem (SLS), besides having a smooth objective, is that, despite nonconvexity, it can be solved globally in some cases [5]. Still, it is well-known that the (SLS) formulation suffers from two major drawbacks: it has no statistical interpretation and the estimates produced by (SLS) are less accurate than those obtained by the (LS) approach [5]. The situation is much better in this respect for problem (LS) which has the advantage of a statistical interpretation as a maximum likelihood whenever the noise is assumed to be Gaussian (see, *e.g.*, [5, 6]). The apparent disadvantage of (LS), however, is that it is nonsmooth: the gradient of the objective function f is not defined for any $x \in \mathcal{A}$. The nonsmoothness of the objective in (LS) also does not permit the derivation of exact global solutions.

In this paper we focus on the more challenging nonconvex and nonsmooth (LS) formulation. For various classes of problems, recent analysis shows that nonconvexity does not always present a significant numerical challenge locally, nor in some cases globally [13, 17]. Nonsmoothness, on the other hand, is a more formidable challenge. One approach to handling nonsmoothness has been via smooth approximations. A popular approach has been to solve convex relaxations of the source localization problem, in particular of the conic types via semi-definite programs or second order conic programs, see for instance [5, 21] and references therein. A clear advantage of using such convex reformulations is that the resulting relaxed problems can be efficiently solved via interior point methods. However, as shown in [5], since the resulting approximate solution is the solution to a *relaxed* problem, the correspondence to solutions of the original problem could be quite poor, if indeed there is any quantifiable correspondence at all. An alternative approach is to seek methods that can tackle directly the original nonconvex and nonsmooth (LS) problem via adequate and simple iterative schemes; see for instance [6], where the authors propose two different schemes to tackle the least squares formulation of the problem in its original form. One of the proposed methods there is based on solving a sequence of certain weighted least squares problems, where convergence of subsequences to cluster points generated by this method to a critical point of the problem was proven, see more details in Section 5 for the advantages and drawbacks of this scheme.

In the present work we eschew convex relaxation techniques. Our objective is to develop and analyze a new simple algorithm for solving the original nonconvex and nonsmooth formulation (LS) of the source localization problem. As just mentioned, the nonsmoothness of the objective function in the (LS) formulation remains a challenging issue. This will be the starting point of our developments. Relying on the simple observation that the Euclidean norm of a given vector is nothing more than the support function of the unit ball (this is Cauchy-Schwartz inequality!), we first derive an equivalent reformulation of (LS), as a smooth constrained nonconvex minimization problem. As we shall see, this novel perspective paves the way to a method which has two advantages over the current state of the art. First, it is very simple: the iteration steps are given by a – closed, computationally inexpensive formula – second, we can show stronger convergence results for the new method than the results derived in [6]. Indeed, exploiting the nice structure of the proposed smooth constrained reformulation of (LS), by adapting some very recent convergence results for semialgebraic optimization [1, 3], and in particular on the general methodology developed in [9], we prove that the proposed method generates a bounded sequence of iterates which converges to a critical point of the original problem (LS) from any given starting point.

After presenting the equivalent smooth constrained reformulation of the problem in Section 2, we study the associated Lagrangian to the problem in Section 3. This is shown to have some very attractive properties that allow us to derive an explicit algorithm (Algorithm 1) based on the alternating direction method of multipliers. Section 4 is devoted to the analysis of our proposed method where we state and prove the main convergence result (Theorem 4.1). Finally, in Section 5 numerical examples are presented to demonstrate the performance of our algorithm relative to the current state of the art.

Our notation and basic definitions are standard [18]. For any vector $r_j \in \mathbb{R}^n$, $j = 1, 2, \ldots, m$, we use the notation $\mathbf{r} := \left(r_1^T, r_2^T, \ldots, r_m^T\right)^T \in \mathbb{R}^{nm}$. We denote the unit closed ball of $\mathbb{R}^n$ by $\mathcal{B} := \{u \in \mathbb{R}^n : \|u\| \leq 1\}$ and, the Cartesian product of m copies of the unit ball $\mathcal{B}$ we denote by $\mathcal{B}^m := \mathcal{B} \times \mathcal{B} \times \cdots \times \mathcal{B}$. The orthogonal projection onto the ball $\mathcal{B}$ is defined by $P_{\mathcal{B}}(u) := \operatorname{argmin}_{v \in \mathcal{B}} \|v - u\|^2$, and $\delta_{\mathcal{B}}$ stands for the indicator function of $\mathcal{B}$.

2 A Smooth Constrained Reformulation of (LS)

We derive an equivalent *smooth constrained* re-formulation of (LS), which provides the key insight toward the main results of this paper, namely a simple algorithm and its convergence analysis. As described in the introduction, we focus our study on tackling the least squares noise error model which is the nonconvex and nonsmooth minimization problem:

$$\min_{x\in\mathbb{R}^n}\left\{f\left(x\right):=\sum_{j=1}^{m}\left(\left\|x-a_j\right\|-d_j\right)^2\right\}. \tag{LS}$$

Elementary algebra shows that (LS) reduces to the following problem (omitting the constant terms d_j^2, $j=1,2,\ldots,m$)

$$\min_{x\in\mathbb{R}^n}\sum_{j=1}^{m}\left(\left\|x-a_j\right\|^2-2d_j\left\|x-a_j\right\|\right). \tag{2.1}$$

Next we replace the nonsmooth term in the objective by a more fundamental representation of the norm. In particular, we note that, from the Cauchy-Schwartz inequality, the norm of any vector $v\in\mathbb{R}^n$ can be rewritten as

$$\left\|v\right\|=\max_{\left\|u\right\|\leq 1}\left\langle u,v\right\rangle. \tag{2.2}$$

Using (2.2) we have

$$-\left\|x-a_j\right\|=\min_{\left\|u_j\right\|\leq 1}-\left\langle u_j,x-a_j\right\rangle,\quad j=1,2,\ldots,m,$$

which yields the equivalent representation of Problem (2.1)

$$\min_{x\in\mathbb{R}^n}\sum_{j=1}^{m}\left(\left\|x-a_j\right\|^2+2d_j\min_{\left\|u_j\right\|\leq 1}-\left\langle u_j,x-a_j\right\rangle\right).$$

This can be conveniently written as a minimization problem of a *smooth function* over a simple convex constraint set, as follows:

$$\min_{(x,\mathbf{u})\in\mathbb{R}^n\times\mathcal{B}^m}\sum_{j=1}^{m}\left(\frac{1}{2}\left\|x-a_j\right\|^2-d_j\left\langle u_j,x-a_j\right\rangle\right).$$

Let $F:\mathbb{R}^n\times\mathbb{R}^{nm}\rightarrow(-\infty,+\infty]$ be a proper and lower semicontinuous function defined by

$$\begin{aligned}F\left(x,\mathbf{u}\right)&:=\sum_{j=1}^{m}\left(\frac{1}{2}\left\|x-a_j\right\|^2-d_j\left\langle u_j,x-a_j\right\rangle+\delta_{\mathcal{B}}\left(u_j\right)\right)\\&:=\Phi(x,\mathbf{u})+\sum_{j=1}^{m}\delta_{\mathcal{B}}\left(u_j\right).\end{aligned} \tag{2.3}$$

Using this notation, problem (2.1) is equivalent to:

$$\min\left\{F\left(x,\mathbf{u}\right):\ (x,\mathbf{u})\in\mathbb{R}^n\times\mathbb{R}^{nm}\right\}, \tag{LOCS}$$

and we clearly have

$$\min_{(x,\mathbf{u})\in\mathbb{R}^n\times\mathbb{R}^{nm}}F\left(x,\mathbf{u}\right)=\min_{x\in\mathbb{R}^n}f\left(x\right)-\sum_{j=1}^{m}d_j^2.$$

This formulation will be the starting point of our analysis. We are interested in designing a simple algorithm for solving (LOCS) which globally converges to a critical point of F, that is, to a pair $(x, \mathbf{u})$ which satisfies (see [19])[1]

$$(0, \mathbf{0}) \in \partial F(x, \mathbf{u}) = \{\nabla_x F(x, \mathbf{u})\} \times \{\partial_{\mathbf{u}} F(x, \mathbf{u})\},$$

which translates to:

$$\nabla_x F(x, \mathbf{u}) = \sum_{j=1}^{m} (x - a_j - d_j u_j) = 0, \tag{2.4}$$

$$\partial_{u_j} F(x, \mathbf{u}) = -d_j (x - a_j) + \partial \delta_{\mathcal{B}}(u_j) \ni 0, \tag{2.5}$$

for all $j = 1, 2, \ldots, m$, where $\partial_{u_j} F(x, \cdot)$ is the subdifferential with respect to u_j.

As we shall see now, we can further exploit the very special and separable structure of the problem to build a simple scheme based on the well-known alternating directions method of multipliers, see, *e.g.*, the recent survey [10] and reference therein.

3 A Simple Algorithm for Solving Problem (LOCS)

To construct a procedure for solving (LOCS), we first decompose the problem via the following equivalent formulation through the new variables v_j, $j = 1, 2, \ldots, m$,

$$\min_{x, \mathbf{u}, \mathbf{v}} \left\{ \sum_{j=1}^{m} \left(\frac{1}{2} \|x - a_j\|^2 - \langle v_j, x - a_j \rangle \right) : \ d_j u_j = v_j, \ u_j \in \mathcal{B}, \ j = 1, 2, \ldots, m \right\}. \tag{3.1}$$

The augmented Lagrangian for the above problem is then defined by

$$L_\rho(x, \mathbf{u}, \mathbf{v}; \mathbf{w}) := \sum_{j=1}^{m} L_{\rho_j}(x, u_j, v_j; w_j), \tag{3.2}$$

where, for each $j = 1, 2, \ldots, m$,

$$L_{\rho_j}(x, u_j, v_j; w_j) := \frac{1}{2} \|x - a_j\|^2 - \langle v_j, x - a_j \rangle + \langle w_j, d_j u_j - v_j \rangle + \frac{\rho_j}{2} \|d_j u_j - v_j\|^2 + \delta_{\mathcal{B}}(u_j). \tag{3.3}$$

[1]By the subdifferential $\partial \varphi$ we mean the *limiting* subdifferential, which is defined for proper and lower semicontinuous functions by

$$\partial \varphi(\bar{z}) := \left\{ v : \exists v^k \to v \text{ and } z^k \xrightarrow{\varphi} \bar{z} \text{ such that } \varphi(z) \geq \varphi\left(z^k\right) + \left\langle v^k, z - z^k \right\rangle + o(|z - z^k|) \right\}.$$

When φ is convex, the above definition coincides with the subdifferential of convex analysis. For more details see [19].

Here $\rho := (\rho_1, \rho_2, \ldots, \rho_m) \in \mathbb{R}^m_{++}$ is the penalty parameter and $w_j \in \mathbb{R}^n$, $j = 1, 2, \ldots, m$, is the multiplier corresponding to the constraint $d_j u_j - v_j = 0$.

The augmented Lagrangian L_ρ is separable in each of the variables (u_j, v_j, w_j), $j = 1, 2, \ldots, m$. This feature is key to the efficiency of our implementation and analysis. Fix $\mathbf{w} \in \mathbb{R}^{nm}$. By inspection one can see that L_{ρ_j}, $j = 1, 2, \ldots, m$, is strongly convex as a function separately of the primal variables $(x, \mathbf{u}, \mathbf{v})$ when the others are fixed. More precisely, we record this useful property in the following proposition.

Proposition 3.1. *For L_ρ defined via (3.2) and (3.3) Then the following hold*

1. *The function $x \to L_\rho(x, \mathbf{u}, \mathbf{v}; \mathbf{w})$ is m-strongly convex, for any fixed triple $(\mathbf{u}, \mathbf{v}, \mathbf{w})$.*

   In addition, for all $j = 1, 2, \ldots, m$, we have

2. *the function $u_j \to L_{\rho_j}(x, u_j, v_j; w_j)$ is $\rho_j d_j^2$-strongly convex, for any fixed triple (x, v_j, w_j);*

3. *the function $v_j \to L_{\rho_j}(x, u_j, v_j; w_j)$ is ρ_j-strongly convex, for any fixed triple (x, u_j, w_j).*

Therefore, each minimization step of the alternating direction of multipliers – which consists of minimizing the augmented Lagrangian L_ρ, in an alternating way for each primal variable, followed by a multiplier update – leads to a well-defined minimization of a *strongly convex* function for each primal step, *i.e.*, it generates the sequence $\left\{\left(x^k, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right)\right\}_{k\in\mathbb{N}}$ via the following basic scheme:

$$x^{k+1} = \operatorname{argmin}_x L_\rho\left(x, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right),$$

and for each $j = 1, 2, \ldots, m$,

$$\begin{aligned}
u_j^{k+1} &= \operatorname{argmin}_{u_j \in \mathcal{B}} L_{\rho_j}\left(x^{k+1}, u_j, v_j^k; w_j^k\right), \\
v_j^{k+1} &= \operatorname{argmin}_{v_j} L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j; w_j^k\right), \\
w_j^{k+1} &= w_j^k + \rho_j\left(d_j u_j^{k+1} - v_j^{k+1}\right).
\end{aligned}$$

It turns out that each minimization can be written in a closed form with a simple formula. Indeed, a straightforward computation yields

$$\nabla_x L_\rho(x, \mathbf{u}, \mathbf{v}; \mathbf{w}) = \sum_{j=1}^m (x - a_j - v_j), \tag{3.4}$$

and for each $j = 1, 2, \ldots, m$,

$$\partial_{u_j} L_{\rho_j}(x, u_j, v_j; w_j) = d_j w_j + \rho_j d_j (d_j u_j - v_j) + \partial \delta_{\mathcal{B}}(u_j), \tag{3.5}$$

$$\nabla_{v_j} L_{\rho_j}(x, u_j, v_j; w_j) = -(x - a_j) - w_j - \rho_j (d_j u_j - v_j), \tag{3.6}$$

$$\nabla_{w_j} L_{\rho_j}(x, u_j, v_j; w_j) = d_j u_j - v_j. \tag{3.7}$$

With the above expressions, the optimality conditions of each convex minimization step in $(x, \mathbf{u}, \mathbf{v})$ of the above basic, results in the following delightfully simple procedure.

Algorithm 1.

Initialization. Start with any $\left(x^0, \mathbf{u}^0, \mathbf{v}^0; \mathbf{w}^0\right) \in \mathbb{R}^n \times \mathbb{R}^{nm} \times \mathbb{R}^{nm} \times \mathbb{R}^{nm}$, and $\rho_j > 0$, $j = 1, 2, \ldots, m$.

Main Loop. For each $k = 0, 1, \ldots$ generate a sequence $\left\{\left(x^k, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right)\right\}_{k \in \mathbb{N}}$ as follows:

- Compute
$$x^{k+1} = \frac{1}{m} \sum_{j=1}^{m} \left(a_j + v_j^k\right). \tag{3.8}$$

- Compute, for each $j = 1, 2, \ldots, m$,
$$u_j^{k+1} = P_{\mathcal{B}}\left(\frac{v_j^k - \rho_j^{-1} w_j^k}{d_j}\right), \tag{3.9}$$
$$v_j^{k+1} = \frac{1}{\rho_j}\left(\rho_j d_j u_j^{k+1} + x^{k+1} - a_j + w_j^k\right), \tag{3.10}$$
$$w_j^{k+1} = w_j^k + \rho_j\left(d_j u_j^{k+1} - v_j^{k+1}\right). \tag{3.11}$$

The u_j step given in (3.9) which is the projection onto the unit ball simply reduces to the formula:

$$u_j^{k+1} = \frac{p_j^k}{\max\{1, \left\|p_j^k\right\|\}}, \text{ where } p_j^k := \frac{v_j^k - \rho_j^{-1} w_j^k}{d_j}, \quad j = 1, 2, \ldots, m.$$

Note that the auxiliary variables v_j^k and w_j^k, $j = 1, 2, \ldots, m$, can be eliminated to produce an algorithm with iterates only on $(x, \mathbf{u})$ thanks to the relation

$$w_j^k = a_j - x^k, \quad \forall\, k \in \mathbb{N},\ \forall\, j \in \{1, 2, \ldots, m\}, \tag{3.12}$$

which can be easily deduced from the algorithms steps. This is presented in Section 5.

3.1 Properties of the Augmented Lagrangian L_ρ

The following elementary, but key fact relates the critical points of the augmented Lagrangian L_ρ (3.2) with those of the objective function F (2.3) of problem (LOCS).

Proposition 3.2. *Let $(x^*, \mathbf{u}^*, \mathbf{v}^*; \mathbf{w}^*)$ be a critical point of L_ρ. Then the pair $(x^*, \mathbf{u}^*)$ is a critical point of F, that is, satisfies (2.4) and (2.5).*

Proof. Since $(x^*, \mathbf{u}^*, \mathbf{v}^*; \mathbf{w}^*)$ is a critical point of L_ρ, we have that $\mathbf{0} \in \partial L_\rho (x^*, \mathbf{u}^*, \mathbf{v}^*; \mathbf{w}^*)$. Thus, it follows from (3.4) and (3.7) that

$$\sum_{j=1}^{m} \left(x^* - a_j - d_j u_j^*\right) = 0,$$

which is exactly (2.4). On the other hand, multiplying (3.6) by d_j and adding to (3.5) yields that

$$0 \in -d_j \left(x^* - a_j\right) + \partial \delta_{\mathcal{B}} \left(u_j^*\right).$$

This shows that (2.5) also holds true for all $j = 1, 2, \ldots, m$. □

The above result suggests that one might try to use the augmented Lagrangian L_ρ as a merit function to measure and analyze the progress of the algorithm. For this idea to work, one must have a *sufficient decrease property* of L_ρ at each iteration. Since our algorithm is of a primal-dual type, at first glance it does not appear possible (*e.g.*, due to the increase in the dual variable $\mathbf{w}$). The trick to guaranteeing sufficient decrease in L_ρ (and hence enabling its use as a merit function) lies in controlling the effect of the dual sequence $\left\{\mathbf{w}^k\right\}_{k\in\mathbb{N}}$. We do this via a simple and adequate choice of the penalty parameter ρ_j, $j = 1, 2, \ldots, m$.

Before showing how this can be done, we first recall the following well-known result (see [19]).

Proposition 3.3. *Let $\varphi : \mathbb{R}^d \to (-\infty, +\infty]$ be proper, lower semicontinuous and σ-strongly convex function. Then, for all $p, q \in \mathbb{R}^d$, we have*

$$\varphi(q) - \varphi(p) \leq \langle \xi, q - p \rangle + \frac{\sigma}{2} \|p - q\|^2, \quad \xi \in \partial\varphi(q). \tag{3.13}$$

Remark 3.1. If φ is a "pure" quadratic function, *i.e.*, of the form $\varphi(p) = (\sigma/2)\|p\|^2 + b^T p + c$ for some $b \in \mathbb{R}^d$ and $c \in \mathbb{R}$, then $\operatorname{dom}\varphi = \mathbb{R}^d$, $\varphi \in C^1$ and equality holds in (3.13) with $\xi = \nabla\varphi(q)$.

We can now prove the sufficient decrease property of the augmented Lagrangian L_ρ. For convenience, we use the notation $\mathbf{y}^k = \left(x^k, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right)$, $k \in \mathbb{N}$, for the sequence generated by Algorithm 1.

Proposition 3.4. *Let* $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ *be a sequence generated by Algorithm 1. Then, for all* $k \in \mathbb{N}$,

$$L_\rho\left(\mathbf{y}^{k+1}\right) - L_\rho\left(\mathbf{y}^k\right) \leq -\frac{\alpha}{2}\left\|x^{k+1}-x^k\right\|^2 - \frac{1}{2}\sum_{j=1}^m \rho_j\left\|v_j^{k+1}-v_j^k\right\|^2 - \frac{1}{2}\sum_{j=1}^m \rho_j d_j^2\left\|u_j^{k+1}-u_j^k\right\|^2,$$

where $\alpha := \sum_{j=1}^m \frac{\rho_j-2}{\rho_j}$. *Therefore, with* $\rho_j > 2$ *for all* $j = 1, 2, \ldots, m$,

$$L_\rho\left(\mathbf{y}^{k+1}\right) - L_\rho\left(\mathbf{y}^k\right) \leq -C_1\left(\left\|x^{k+1}-x^k\right\|^2 + \sum_{j=1}^m\left\|v_j^{k+1}-v_j^k\right\|^2\right), \tag{3.14}$$

where $C_1 = \min\left\{\alpha, \rho_1, \rho_2, \ldots, \rho_m\right\}$.

Proof. First note that by the definition of algorithm (Steps (3.8)–(3.10)) we have that

$$0 = \nabla_x L_\rho\left(x^{k+1}, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right), \tag{3.15}$$

and for each $j = 1, 2, \ldots, m$,

$$0 \in \partial_{u_j} L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^k; w_j^k\right) + \partial_{u_j}\delta_{\mathcal{B}}\left(u_j^{k+1}\right),$$
$$0 = \nabla_{v_j} L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^{k+1}; w_j^k\right).$$

These facts, together with Proposition 3.3 and the strong convexity of L_ρ (*cf.* Proposition 3.1) applied to

$$\varphi(x) = L_\rho\left(x, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right) \text{ at the points } q = x^{k+1} \text{ and } p = x^k,$$
$$\varphi(u_j) = L_{\rho_j}\left(x^{k+1}, u_j, v_j^k; w_j^k\right) + \delta_{\mathcal{B}}(u_j) \text{ at the points } q = u_j^{k+1} \text{ and } p = u_j^k,$$
$$\varphi(v_j) = L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^{k+1}; w_j^k\right) \text{ at the points } q = v_j^{k+1} \text{ and } p = v_j^k,$$

yield

$$L_\rho\left(x^{k+1}, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right) - L_\rho\left(\mathbf{y}^k\right) = -\frac{m}{2}\left\|x^{k+1}-x^k\right\|^2, \tag{3.16}$$

$$L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^k; w_j^k\right) - L_{\rho_j}\left(x^{k+1}, u_j^k, v_j^k; w_j^k\right) \leq -\frac{\rho_j d_j^2}{2}\left\|u_j^{k+1}-u_j^k\right\|^2, \tag{3.17}$$

$$L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^{k+1}; w_j^k\right) - L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^k; w_j^k\right) = -\frac{\rho_j}{2}\left\|v_j^{k+1}-v_j^k\right\|^2, \tag{3.18}$$

where equalities (3.16) and (3.18) follow from Remark 3.1. The representation of the augmented Lagrangian in (3.3) together with (3.11) and (3.12) gives

$$\begin{aligned} L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^{k+1}; w_j^{k+1}\right) - L_{\rho_j}\left(x^{k+1}, u_j^{k+1}, v_j^{k+1}; w_j^k\right) &= \left\langle w_j^{k+1}-w_j^k, v_j^{k+1}-a_j-d_j u_j^{k+1}\right\rangle \\ &= \frac{1}{\rho_j}\left\|w_j^{k+1}-w_j^k\right\|^2, \\ &= \frac{1}{\rho_j}\left\|x^{k+1}-x^k\right\|^2. \end{aligned} \tag{3.19}$$

Now, recalling that $L_\rho = \sum_{j=1}^m L_{\rho_j}$, equations (3.16), (3.18) and (3.19) together with (3.17) yield

$$L_\rho\left(\mathbf{y}^{k+1}\right) - L_\rho\left(\mathbf{y}^k\right) \leq -\sum_{j=1}^m \frac{\rho_j - 2}{2\rho_j}\left\|x^{k+1} - x^k\right\|^2 - \frac{1}{2}\sum_{j=1}^m \rho_j\left\|v_j^{k+1} - v_j^k\right\|^2 - \frac{1}{2}\sum_{j=1}^m \rho_j d_j^2\left\|u_j^{k+1} - u_j^k\right\|^2,$$

which proves the first statement. The second statement (3.14) follows immediately from the first one whenever $\rho_j > 2$. This completes the proof. $\square$

Standing assumption. From now on, we assume that

$$\rho_j > 2, \quad \forall\ j = 1, 2, \ldots, m.$$

Finally, once again thanks to the nice structure of L_ρ, we show that at each iteration $k \in \mathbb{N}$, $L_\rho\left(\mathbf{y}^k\right)$ is bounded from below.

Proposition 3.5. *Let $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1. Then, for each $k \in \mathbb{N}$, we have*

$$L_\rho\left(\mathbf{y}^k\right) \geq -\frac{1}{2}\sum_{j=1}^m d_j^2.$$

Proof. We first re-write the augmented Lagrangian L_{ρ_j}, $j = 1, 2, \ldots, m$, as follows

$$\begin{aligned} L_{\rho_j}\left(x, u_j, v_j; w_j\right) &= \frac{1}{2}\left\|x - a_j\right\|^2 - \left\langle v_j, x - a_j\right\rangle + \left\langle w_j, d_j u_j - v_j\right\rangle + \frac{\rho_j}{2}\left\|d_j u_j - v_j\right\|^2 + \delta_{\mathcal{B}}\left(u_j\right) \\ &= \frac{1}{2}\left\|x - a_j - d_j u_j\right\|^2 - \frac{d_j^2}{2}\left\|u_j\right\|^2 + \left\langle d_j u_j - v_j, x - a_j + w_j\right\rangle + \frac{\rho_j}{2}\left\|d_j u_j - v_j\right\|^2 \\ &\quad + \delta_{\mathcal{B}}\left(u_j\right). \end{aligned}$$

Note also that, for all $k \in \mathbb{N}$, $x^k - a_j + w_j^k = 0$ (*cf.* (3.12)) and $u_j^k \in \mathcal{B}$ for all $j = 1, 2, \ldots, m$, hence

$$L_{\rho_j}\left(x^k, u_j^k, v_j^k; w_j^k\right) = \frac{1}{2}\left\|x^k - a_j - d_j u_j^k\right\|^2 - \frac{d_j^2}{2}\left\|u_j^k\right\|^2 + \frac{\rho_j}{2}\left\|d_j u_j^k - v_j^k\right\|^2 \geq -\frac{d_j^2}{2}\left\|u_j^k\right\| \geq -\frac{d_j^2}{2}.$$

Using the fact that $L_\rho\left(\mathbf{y}^k\right) = \sum_{j=1}^m L_{\rho_j}\left(x^k, u_j^k, v_j^k; w_j^k\right)$ then yields the result. $\square$

4 Convergence Analysis

The main goal of this section is to derive convergence properties of Algorithm 1. To establish the main convergence result, namely global convergence of the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ generated by Algorithm 1 to a critical point of the function F (*cf.* (2.3)), we follow a general convergence mechanism first described in [1] and more recently extended and simplified in [9]. Here we follow the general approach developed in [9], whereby a systematic mechanism and "recipe" was derived to prove global convergence of any sequence (satisfying certain conditions) produced by a given algorithm, independently of the algorithms used. See details in [9, Section 3.2], and in particular conditions (i), (ii) and (iii) in [9, p. 470]. The first two bring conditions on the sequence; while the last one is on problem's data information requiring the objective function to satisfy the so-called Kurdyka-Łojasiewicz (KL) property [15, 14]. For recent advances and impact of the KL property in optimization we refer the reader to [7, 1, 8].

Unfortunately, this general mechanism cannot be directly applied, since conditions (i) and (ii) stated in [9, p. 470] do not hold for our method. Nevertheless, thanks to the generous structure of the problem (3.1) and of the augmented Lagrangian L_ρ (3.2) we will be able to prove convergence of sequences $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ generated by Algorithm 1 to critical points of F from any initial point $\mathbf{y}^0$.

To begin, note first that both the objective function F of (LOCS) and the corresponding augmented Lagrangian L_ρ are *semi-algebraic* functions, and hence, thanks to a result established in [7], any proper and lower semicontinuous function which is semi-algebraic satisfies the KL property at any point of the domain.

Next, we show that $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is a bounded sequence.

Proposition 4.1. *Let $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1. Then*

1. *the sequence $\left\{\left\|\mathbf{y}^{k+1}-\mathbf{y}^k\right\|\right\}_{k\in\mathbb{N}}$ converges to zero as $k\to\infty$ and*
2. *the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is bounded.*

Proof. 1. From Proposition 3.4 we have, for all $k\in\mathbb{N}$, that

$$\sum_{j=1}^{m}\left(\frac{\rho_j-2}{2\rho_j}\left\|x^{k+1}-x^k\right\|^2+\frac{\rho_j}{2}\left\|v_j^{k+1}-v_j^k\right\|^2+\frac{\rho_j d_j^2}{2}\left\|u_j^{k+1}-u_j^k\right\|^2\right)\le L_\rho\left(\mathbf{y}^k\right)-L_\rho\left(\mathbf{y}^{k+1}\right).$$

Since $\rho_j>2$ for all $j=1,2,\ldots,m$, summing this inequality for $k=1,2,\ldots,N$ we obtain that

$$\sum_{k=1}^{N}\sum_{j=1}^{m}\left(\frac{\rho_j-2}{2\rho_j}\left\|x^{k+1}-x^k\right\|^2+\frac{\rho_j}{2}\left\|v_j^{k+1}-v_j^k\right\|^2+\frac{\rho_j d_j^2}{2}\left\|u_j^{k+1}-u_j^k\right\|^2\right)\le L_\rho\left(\mathbf{y}^1\right)-L_\rho\left(\mathbf{y}^N\right).$$

Thus, since by Proposition 3.5 $\left\{L_\rho\left(\mathbf{y}^k\right)\right\}_{k\in\mathbb{N}}$ is bounded from below, we let $N\to\infty$ and obtain, for all $j=1,2,\ldots,m$, that

$$\sum_{k=1}^{\infty}\left\|x^{k+1}-x^k\right\|^2<\infty,\quad \sum_{k=1}^{\infty}\left\|v_j^{k+1}-v_j^k\right\|^2 \quad\text{and}\quad \sum_{k=1}^{\infty}\left\|u_j^{k+1}-u_j^k\right\|^2<\infty.$$

This shows that the sequences $\left\{\left\|x^{k+1}-x^k\right\|\right\}_{k\in\mathbb{N}}$, $\left\{\left\|\mathbf{u}^{k+1}-\mathbf{u}^k\right\|\right\}_{k\in\mathbb{N}}$ and $\left\{\left\|\mathbf{v}^{k+1}-\mathbf{v}^k\right\|\right\}_{k\in\mathbb{N}}$ all converge to zero as $k\to\infty$. Now, in order to complete the proof of the first item, all we need is to show that $\left\{\left\|\mathbf{w}^{k+1}-\mathbf{w}^k\right\|\right\}_{k\in\mathbb{N}}$ also converges to zero as $k\to\infty$. This immediately follows from (3.12). Combining all these facts yields 1.

2. First of all, it is clear from the algorithm that $\left\{\mathbf{u}^k\right\}_{k\in\mathbb{N}}\subset\mathcal{B}^m$ which implies directly that $\left\{\mathbf{u}^k\right\}_{k\in\mathbb{N}}$ is bounded. In addition, from part 1 it is also true that $\left\{\left\|w_j^{k+1}-w_j^k\right\|\right\}_{k\in\mathbb{N}}$ is bounded. Thus, using Step (3.11), for all $j=1,2,\ldots,m$

$$\left\|d_j u_j^k-v_j^k\right\|=\frac{1}{\rho_j}\left\|w_j^k-w_k^{k-1}\right\|,$$

and therefore

$$\left\|v_j^k\right\|\leq\frac{1}{\rho_j}\left\|w_j^k-w_k^{k-1}\right\|+d_j\left\|u_j^k\right\|.$$

Since the right-hand side of the above inequality is bounded, it follows that $\left\{v_j^k\right\}_{k\in\mathbb{N}}$ is bounded for all $j=1,2,\ldots,m$, and therefore $\left\{\mathbf{v}^k\right\}_{k\in\mathbb{N}}$ is bounded. Using now Step (3.8) we immediately obtain that $\left\{x^k\right\}_{k\in\mathbb{N}}$ is bounded, and from (3.12) that $\left\{\mathbf{w}^k\right\}_{k\in\mathbb{N}}$ is also bounded. It follows then that $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is bounded. $\square$

Now we derive a lower bound on the steps in the primal variables $\left\{\left(x^k,\mathbf{v}^k\right)\right\}_{k\in\mathbb{N}}$ in terms of an element from the subdifferential.

Proposition 4.2. *Let $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1. Then, for each $k\in\mathbb{N}$, there exist positive constants M and M_j, $j=1,2,\ldots,m$, such that*

$$\left\|\mathbf{p}^k\right\|\leq M\left\|x^k-x^{k-1}\right\|+\sum_{j=1}^{m}M_j\left\|v_j^k-v_j^{k-1}\right\|,$$

where

$$M=m+\sum_{j=1}^{m}\left(d_j+\frac{1}{\rho_j}\right)\qquad M_j=1+\rho_j d_j,\quad j=1,2,\ldots,m, \tag{4.1}$$

and $\mathbf{p}^k=\left(p_1^k,\mathbf{p}_2^k,\mathbf{p}_3^k,\mathbf{p}_4^k\right)\in\partial L_\rho\left(\mathbf{y}^k\right)$ is given by

$$p_1^k=\sum_{j=1}^{m}\left(v_j^{k-1}-v_j^k\right),$$

and for all $j = 1, 2, \ldots, m$,

$$\begin{aligned}
\left(p_2^k\right)_j &= d_j \left(w_j^k - w_j^{k-1}\right) + \rho_j d_j \left(v_j^{k-1} - v_j^k\right), \\
\left(p_3^k\right)_j &= w_j^{k-1} - w_j^k, \\
\left(p_4^k\right)_j &= \frac{1}{\rho_j} \left(w_j^k - w_j^{k-1}\right).
\end{aligned}$$

In particular,

$$\left\|\mathbf{p}^k\right\| \leq C_2 \left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\|, \tag{4.2}$$

where $C_2 = \sqrt{m+1} \cdot \max\{M, M_1, M_2, \ldots, M_m\}$ *and* $\mathbf{z}^k = \left(x^k, \mathbf{v}^k\right)$, $k \in \mathbb{N}$.

Proof. We first verify that $\mathbf{p}^k \in \partial L_\rho\left(\mathbf{y}^k\right)$ for all $k \in \mathbb{N}$. Equation (3.4) together with the update formula (3.8) for x^k yields $p_1^k = \sum_{j=1}^m \left(v_j^{k-1} - v_j^k\right) = \nabla_x L_\rho\left(\mathbf{y}^k\right)$. By the update rule given in (3.9), applied at iteration k, we have $\rho_j d_j v_j^{k-1} - d_j w_j^{k-1} \in \rho_j d_j^2 u_j^k + \partial \delta_{\mathcal{B}}\left(u_j^k\right)$. Hence

$$\begin{aligned}
\left(p_2^k\right)_j &= d_j \left(w_j^k - w_j^{k-1}\right) + \rho_j d_j \left(v_j^{k-1} - v_j^k\right) \\
&= \left(d_j w_j^k - \rho_j d_j v_j^k\right) + \left(\rho_j d_j v_j^{k-1} - d_j w_j^{k-1}\right) \\
&\in \partial_{u_j} L_{\rho_j}\left(\mathbf{y}^k\right),
\end{aligned}$$

where the inclusion follows from (3.5). A straightforward application of (3.6) and (3.7) together with the update formulas (3.10) and (3.11) for v^k, yields

$$\begin{aligned}
\left(p_3^k\right)_j &= w_j^{k-1} - w_j^k \\
&= -\left(x^k - a_j\right) - w_j^k - \rho_j d_j u_j^k + \left(\rho_j d_j u_j^k + x^k - a_j + w_j^{k-1}\right) \\
&= -\left(x^k - a_j\right) - w_j^k - \rho_j d_j u_j^k + \rho_j v_j^k \\
&= \nabla_{v_j} L_{\rho_j}\left(\mathbf{y}^k\right),
\end{aligned}$$

and

$$\left(p_4^k\right)_j = \frac{1}{\rho_j}\left(w_j^k - w_j^{k-1}\right) = d_j u_j^k - v_j^k = \nabla_{w_j} L_{\rho_j}\left(\mathbf{y}^k\right).$$

for all $j = 1, 2, \ldots, m$, as claimed.

Now using the expressions for $\mathbf{p}^k$, we obtain the following bound for $k \in \mathbb{N}$,

$$\left\|\mathbf{p}^k\right\| \leq \sum_{j=1}^m \left(d_j + 1 + \frac{1}{\rho_j}\right)\left\|w_j^k - w_j^{k-1}\right\| + \sum_{j=1}^m \left(1 + d_j \rho_j\right)\left\|v_j^k - v_j^{k-1}\right\|.$$

Using (3.12) we obtain that

$$\left\|\mathbf{p}^k\right\| \leq M\left\|x^k - x^{k-1}\right\| + \sum_{j=1}^{m} M_j\left\|v_j^k - v_j^{k-1}\right\|,$$

where M and M_j, $j = 1, 2, \ldots, m$, are given in (4.1). This proves the first result. Now, we can use the fact that for any vector $\mathbf{a} \in \mathbb{R}^{m+1}$ we have that $\|\mathbf{a}\|_1 \leq \sqrt{m+1}\,\|\mathbf{a}\|_2$. Thus

$$\left\|\mathbf{p}^k\right\| \leq C_2\left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\|,$$

where $C_2 = \sqrt{m+1}\max\{M, M_1, M_2, \ldots, M_m\}$. □

Subsequent results make use of the following notation. The set of all cluster points of sequences $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ generated by Algorithm 1 with initial point $\mathbf{y}^0$ is denoted $\omega\left(\mathbf{y}^0\right)$ and defined by

$$\left\{\overline{\mathbf{y}} \in \mathbb{R}^d : \exists \text{ an increasing sequence of integers } \{k_l\}_{l\in\mathbb{N}} \text{ such that } \mathbf{y}^{k_l} \to \overline{\mathbf{y}} \text{ as } l \to \infty\right\}.$$

The set of all critical points of L_ρ is denoted and defined by $\operatorname{crit} L_\rho := \left\{\mathbf{y} \in \mathbb{R}^d : \mathbf{0} \in \partial L_\rho(\mathbf{y})\right\}$. The next result establishes that $\omega\left(\mathbf{y}^0\right)$ consists entirely of critical points of L_ρ, and therefore of F, thanks to Proposition 3.2.

Lemma 4.1. *Let $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1. The set of cluster points $\omega\left(\mathbf{y}^0\right)$ is nonempty, compact and satisfies the following two properties:*

1. $\omega\left(\mathbf{y}^0\right) \subset \operatorname{crit} L_\rho$;
2. $\lim_{k\to\infty} \operatorname{dist}(\mathbf{y}^k, \omega\left(\mathbf{y}^0\right)) = 0$.

Moreover, L_ρ is finite and constant on $\omega\left(\mathbf{y}^0\right)$.

Proof. Since $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is a bounded sequence by Proposition 4.12, it is obvious that $\omega\left(\mathbf{y}^0\right)$ is nonempty. In addition, since $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is bounded it follows that $\omega\left(\mathbf{y}^0\right)$ is bounded too. By the definition of $\omega\left(\mathbf{y}^0\right)$ it is also closed and therefore compact.

Now we will prove the properties 1 and 2. Let $\mathbf{y}^* = (x^*, \mathbf{u}^*, \mathbf{v}^*; \mathbf{w}^*)$ be a limit point of $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ which exists since the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is bounded as we proved in Proposition 4.12. This means that there is a subsequence $\left\{\left(x^{k_q}, \mathbf{u}^{k_q}, \mathbf{v}^{k_q}; \mathbf{w}^{k_q}\right)\right\}_{q\in\mathbb{N}}$ for which $\left(x^{k_q}, \mathbf{u}^{k_q}, \mathbf{v}^{k_q}; \mathbf{w}^{k_q}\right) \to (x^*, \mathbf{u}^*, \mathbf{v}^*; \mathbf{w}^*)$ as $q \to \infty$. Therefore from the continuity of L_ρ, it follows that

$$\lim_{q\to\infty} L_\rho\left(x^{k_q}, \mathbf{u}^{k_q}, \mathbf{v}^{k_q}; \mathbf{w}^{k_q}\right) = L_\rho\left(x^*, \mathbf{u}^*, \mathbf{v}^*; \mathbf{w}^*\right).$$

On the other hand from Propositions 3.4, 4.12 and 4.2, we know that $\mathbf{p}^k \in \partial L_\rho\left(\mathbf{y}^k\right)$ and $\mathbf{p}^k \to \mathbf{0}$ as $k \to \infty$. The closedness property of ∂L_ρ (see [9, Remark 1(ii), p. 464]) implies that $\mathbf{0} \in \partial L_\rho\left(\mathbf{y}^*\right)$. This proves that $\mathbf{y}^*$ is a critical point of L_ρ and completes the proof of the first item. The second item follows immediately from the definition of limit points.

To complete the proof of the lemma we have to show that L_ρ is finite and constant on $\omega\left(\mathbf{y}^0\right)$. The sequence $\left\{L_\rho\left(\mathbf{y}^k\right)\right\}_{k\in\mathbb{N}}$ decreases by Proposition 3.4 and by Proposition 3.5 is bounded from below which implies convergence to some finite limit l. It is also follows that L_ρ is constant on $\omega\left(\mathbf{y}^0\right)$. □

For $\eta \in (0, +\infty]$ we denote by Φ_η the set of all concave and continuous functions $\varphi : [0, \eta) \to \mathbb{R}_+$ which satisfy $\varphi(0) = 0$, $\varphi \in C^1$ on $(0, \eta)$ and $\varphi'(s) > 0$ for all $s \in (0, \eta)$. We recall next the following key result obtained in [9, Lemma 6, p. 478].

Lemma 4.2 (uniform KL property)**.** *Let Ω be a compact set and let $\sigma : \mathbb{R}^d \to (-\infty, \infty]$ be a proper and lower semicontinuous function. Assume that σ is constant on Ω and satisfies the KL property at each point of Ω. Then there exist $\varepsilon > 0$, $\eta > 0$ and $\varphi \in \Phi_\eta$ such that for all $\overline{u}$ in Ω and all u in the following intersection*

$$\left\{u \in \mathbb{R}^d : \operatorname{dist}(u, \Omega) < \varepsilon\right\} \cap [\sigma(\overline{u}) < \sigma(u) < \sigma(\overline{u}) + \eta], \tag{4.3}$$

one has,

$$\varphi'(\sigma(u) - \sigma(\overline{u}))\operatorname{dist}(0, \partial\sigma(u)) \geq 1. \tag{4.4}$$

We can now prove our main result.

Theorem 4.1 (global convergence to critical values)**.** *For any initial point $\mathbf{y}^0$, the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ generated by Algorithm 1 applied to problem* (LOCS) *converges to a critical point $\mathbf{y}^*$ of the augmented Lagrangian L_ρ. In particular, the pair $(x^*, \mathbf{u}^*)$ is a critical point of F.*

Proof. Our main task is to show that the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}} := \left\{\left(x^k, \mathbf{u}^k, \mathbf{v}^k; \mathbf{w}^k\right)\right\}_{k\in\mathbb{N}}$ has finite length. For convenience, we also use the notation $\mathbf{z}^k = \left(x^k, \mathbf{v}^k\right)$, $k \in \mathbb{N}$. Since by Proposition 4.12 the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is bounded, there exists a convergent sub-sequence $\left\{\mathbf{y}^{m_k}\right\}_{k\in\mathbb{N}}$ such that $\mathbf{y}^{m_k} \to \bar{\mathbf{y}}$ as $k \to \infty$. From the continuity of L_ρ, it follows that

$$\lim_{k\to\infty} L_\rho\left(\mathbf{y}^k\right) = L_\rho(\bar{\mathbf{y}}). \tag{4.5}$$

If there exists an integer $\bar{k} \in \mathbb{N}$ for which $L_\rho\left(\mathbf{y}^{\bar{k}}\right) = L_\rho(\bar{\mathbf{y}})$ then the decreasing property obtained in Proposition 3.4 would imply that $\mathbf{y}^{\bar{k}+1} = \mathbf{y}^{\bar{k}}$. A simple induction shows then that the sequence $\left\{\mathbf{y}^k\right\}_{k\in\mathbb{N}}$ is stationary and the claim follows.

Since $\left\{L_\rho\left(\mathbf{y}^k\right)\right\}_{k\in\mathbb{N}}$ is a decreasing sequence by Proposition 3.4, it is clear from (4.5) that $L_\rho\left(\bar{\mathbf{y}}\right) < L_\rho\left(\mathbf{y}^k\right)$ for all $k \in \mathbb{N}$. Again from (4.5), for any $\eta > 0$ there exists $k_0 \in \mathbb{N}$ such that

$$L_\rho\left(\mathbf{y}^k\right) < L_\rho\left(\bar{\mathbf{y}}\right) + \eta, \quad \forall\ k > k_0.$$

From Lemma 4.12 we know that $\lim_{k\to\infty} \operatorname{dist}\left(\mathbf{y}^k, \omega\left(\mathbf{y}^0\right)\right) = 0$. This means that for any $\varepsilon > 0$ there exists a positive integer k_1 such that $\operatorname{dist}\left(\mathbf{y}^k, \omega\left(\mathbf{y}^0\right)\right) < \varepsilon$ for all $k > k_1$. Summing up all these facts, we get that $\mathbf{y}^k$ belongs to the intersection in (4.3) for all $k > l := \max\left\{k_0, k_1\right\}$.

It follows from Lemma 4.1 that $\omega\left(\mathbf{y}^0\right)$ is nonempty and compact, L_ρ is finite and constant on $\omega\left(\mathbf{y}^0\right)$. Now, as we previously noticed, L_ρ is semi-algebraic (a polynomial function) and satisfies the KL property. Therefore we can apply Lemma 4.2 with $\Omega = \omega\left(\mathbf{y}^0\right)$. Therefore for any $k > l$ we have

$$\varphi'\left(L_\rho\left(\mathbf{y}^k\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right) \cdot \operatorname{dist}\left(0, \partial L_\rho\left(\mathbf{y}^k\right)\right) \geq 1. \tag{4.6}$$

This makes sense since we know that $L_\rho\left(\mathbf{y}^k\right) > L_\rho\left(\bar{\mathbf{y}}\right)$ for any $k > l$. From (4.2) of Proposition 4.2 we also have that

$$\left\|\mathbf{p}^k\right\| \leq C_2\left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\|, \quad C_2 > 0. \tag{4.7}$$

Combining this fact with (4.6) yields that

$$\varphi'\left(L_\rho\left(\mathbf{y}^k\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right) \geq \frac{1}{\operatorname{dist}\left(0, \partial L_\rho\left(\mathbf{y}^k\right)\right)} \geq \left(C_2\left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\|\right)^{-1}. \tag{4.8}$$

On the other hand, from the concavity of φ we get that

$$\varphi\left(L_\rho\left(\mathbf{y}^k\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right) - \varphi\left(L_\rho\left(\mathbf{y}^{k+1}\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right) \geq \varphi'\left(L_\rho\left(\mathbf{y}^k\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right)\left(L_\rho\left(\mathbf{y}^k\right) - L_\rho\left(\mathbf{y}^{k+1}\right)\right). \tag{4.9}$$

For convenience, we define for all $p, q \in \mathbb{N}$ and $\bar{\mathbf{y}}$ the following quantities

$$\Delta_{p,q} := \varphi\left(L_\rho\left(\mathbf{y}^p\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right) - \varphi\left(L_\rho\left(\mathbf{y}^q\right) - L_\rho\left(\bar{\mathbf{y}}\right)\right).$$

Combining (4.8) and (4.9) while using (3.14) yields for any $k > l$ that

$$\Delta_{k,k+1} \geq \frac{C_1\left\|\mathbf{z}^{k+1} - \mathbf{z}^k\right\|^2}{C_2\left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\|}, \tag{4.10}$$

and hence

$$\left\|\mathbf{z}^{k+1} - \mathbf{z}^k\right\|^2 \leq \gamma\Delta_{k,k+1}\left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\|,$$

where $\gamma = C_2/C_1$. Using the fact that $2\sqrt{ab} \leq a + b$ for all $a, b \geq 0$, we infer

$$2\left\|\mathbf{z}^{k+1} - \mathbf{z}^k\right\| \leq \left\|\mathbf{z}^k - \mathbf{z}^{k-1}\right\| + \gamma\Delta_{k,k+1}. \tag{4.11}$$

Let us now prove that for any $k > l$ the following inequality holds

$$\sum_{i=l+1}^{k} \left\| \mathbf{z}^{i+1} - \mathbf{z}^{i} \right\| \leq \left\| \mathbf{z}^{l+1} - \mathbf{z}^{l} \right\| + \gamma \Delta_{l+1,k+1}.$$

Summing up (4.11) for $i = l+1, l+2, \ldots, k$ yields

$$\begin{aligned} 2 \sum_{i=l+1}^{k} \left\| \mathbf{z}^{i+1} - \mathbf{z}^{i} \right\| &\leq \sum_{i=l+1}^{k} \left\| \mathbf{z}^{i} - \mathbf{z}^{i-1} \right\| + \gamma \sum_{i=l+1}^{k} \Delta_{i,i+1} \\ &\leq \sum_{i=l+1}^{k+1} \left\| \mathbf{z}^{i} - \mathbf{z}^{i-1} \right\| + \gamma \sum_{i=l+1}^{k} \Delta_{i,i+1} \\ &= \sum_{i=l+1}^{k} \left\| \mathbf{z}^{i+1} - \mathbf{z}^{i} \right\| + \left\| \mathbf{z}^{l+1} - \mathbf{z}^{l} \right\| + \gamma \Delta_{l+1,k+1}, \end{aligned}$$

where the last inequality follows from the fact that $\Delta_{p,q} + \Delta_{q,r} = \Delta_{p,r}$ for all $p, q, r \in \mathbb{N}$. Since $\varphi \geq 0$, we thus have for any $k > l$ that

$$\sum_{i=l+1}^{k} \left\| \mathbf{z}^{i+1} - \mathbf{z}^{i} \right\| \leq \left\| \mathbf{z}^{l+1} - \mathbf{z}^{l} \right\| + \gamma \varphi \left(L_{\rho} \left(\mathbf{y}^{l+1} \right) - L_{\rho} \left(\bar{\mathbf{y}} \right) \right).$$

Since the right hand-side of the inequality above does not depend on k at all, it follows immediately that the sequence $\left\{ \mathbf{z}^{k} \right\}_{k \in \mathbb{N}}$ has finite length, that is,

$$\sum_{k=1}^{\infty} \left\| \mathbf{z}^{k+1} - \mathbf{z}^{k} \right\| < \infty, \tag{4.12}$$

which implies that both $\left\{ x^{k} \right\}_{k \in \mathbb{N}}$ and $\left\{ \mathbf{v}^{k} \right\}_{k \in \mathbb{N}}$ also have finite length. In addition, from (3.12) we have for all $j = 1, 2, \ldots, m$, that

$$w_j^{k+1} - w_j^{k} = x^{k+1} - x^{k},$$

which implies that $\left\{ \mathbf{w}^{k} \right\}_{k \in \mathbb{N}}$ also has finite length. Now, using (3.9) we obtain that

$$\begin{aligned} \left\| u_j^{k+1} - u_j^{k} \right\| &= \left\| P_{\mathcal{B}} \left(\frac{v_j^{k} + \rho_j^{-1} w_j^{k}}{d_j} \right) - P_{\mathcal{B}} \left(\frac{v_j^{k-1} + \rho_j^{-1} w_j^{k-1}}{d_j} \right) \right\| \\ &\leq \frac{1}{d_j} \left\| v_j^{k} + \rho_j^{-1} w_j^{k} - \left(v_j^{k-1} + \rho_j^{-1} w_j^{k-1} \right) \right\| \\ &\leq \frac{\rho_j}{d_j} \left\| v_j^{k} - v_j^{k-1} \right\| + \frac{1}{d_j} \left\| w_j^{k} - w_j^{k-1} \right\|, \end{aligned}$$

where the first inequality uses the nonexpansivity of the projection $P_{\mathcal{B}}$, and hence the latter inequality implies that $\left\{ \mathbf{u}^{k} \right\}_{k \in \mathbb{N}}$ also has finite length. Therefore $\left\{ \mathbf{y}^{k} \right\}_{k \in \mathbb{N}}$ has finite length which means that it is a Cauchy sequence and hence a convergent sequence. Let $\mathbf{y}^{*}$ be a limit point of $\left\{ \mathbf{y}^{k} \right\}_{k \in \mathbb{N}}$. From Lemma 4.1 it is clear that $\mathbf{y}^{*}$ is a critical point of L_{ρ}, as asserted. Thanks to Proposition 3.2, we then obtain that $(x^{*}, \mathbf{u}^{*})$ is a critical point of F. $\square$

5 Numerical Examples

The structure of (LOCS) opens the door to consider possibly different types of iterative schemes for solving (LS). Before turning to the implementation of Algorithm 1, we first briefly discuss two known algorithms that are not unreasonable candidates for tackling (LOCS). However, as we shall explain below, both schemes are not competitive.

First, note that the objective function Φ given by (2.1) is nonconvex and continuously differentiable (a nonconvex quadratic) in $(x, \mathbf{u})$, and hence admits a Lipschitz continuous gradient with Lipschitz constant L. Thus one scheme that easily could be applied to (LOCS) is the projected gradient method. Projected gradients has been studied thoroughly in [2] where, under suitable assumptions satisfied by (LOCS), it is shown to converge globally to critical points. More recently it has been shown that, on a local neighborhood of the critical points, assuming these are isolated points, projected gradients in fact converges linearly for any problem with this structure [17, Proposition 6.8]. Our numerical experiments indicate, however, that despite the analytical guarantees, projected gradient is not competitive *globally* to Algorithm 1. Moreover, its performance deteriorates rapidly as the number of anchors m increases. This is explained by the inverse dependence of the Lipschitz constant L on m: the larger m, the smaller the step sizes. (The same behavior was observed when using a backtracking procedure for L.) We therefore do not report on numerical testing for this scheme.

Another natural scheme that could be applied to (LOCS) is the alternating minimization (Gauss-Seidel) method. Observe that Φ is convex in each of its arguments separately, then the alternating minimization method applied to (LOCS) would generate iterates by solving the two convex programs

$$x^{k+1} = \operatorname{argmin}\left\{\Phi\left(x, \mathbf{u}^k\right) : x \in \mathbb{R}^n\right\},$$

and

$$\mathbf{u}^{k+1} \in \operatorname{argmin}\left\{\Phi\left(x^{k+1}, \mathbf{u}\right) : \mathbf{u} \in \mathcal{B}^m\right\}.$$

It is easy to see that both minimization steps can be explicitly solved, yielding a very simple algorithm. It turns out (after some algebra left to the reader) that the resulting alternating minimization scheme reduces to a fixed point algorithm already proposed in [6] where it is called SFP. The SFP method has two drawbacks. First, global convergence to critical points can be guaranteed under the assumption that these are isolated points (see [6, Theorem 2.2]). Second, as established in that work, the numerical performance of SFP is significantly worse than the SWLS method also developed in [6], and which we now consider in our comparisons below.

We present a numerical comparison of Algorithm 1 for solving (LOCS) against the SWLS algorithm proposed in [6], which appears to be the state of the art for this class of problems. We will not provide here the exact description of the SWLS algorithm, but it is important to note that this is a quite involved algorithm, since at each iteration of the SWLS, a generalized trust region subproblem must be solved. SWLS is thus a nested optimization algorithm which is quite expansive to implement as we will see in the numerical examples below. Before presenting the numerical experiments and findings, we first discuss a few theoretical aspects of the SWLS algorithm.

In [6], the authors prove that if $\left\{x^k\right\}_{k\in\mathbb{N}}$ is a sequence generated by SWLS, then every subsequence of it converges to a critical point of the original objective function f (see [6, Theorem 3.1, p. 1410]). Algorithm 1, in comparison, is guaranteed to generate a convergent sequence, not just a subsequence, from any initial point. In addition to this global convergence property, we would like to emphasize two further drawbacks of the SWLS algorithm (see [6, p. 1408]). The analysis is based on two assumptions which are not needed when we analyze Algorithm 1. The first assumption is on the problem's data, that is, on the set of anchors a_j, $j = 1, 2, \ldots, m$, and requires that the matrix

$$\begin{bmatrix} 1 & a_1^T \\ 1 & a_2^T \\ \vdots & \vdots \\ 1 & a_m^T \end{bmatrix}$$

has full column rank. Such an assumption is quite common when dealing with localization problems and seems to hold in many cases. The second assumption, which seems to be more demanding, comes about because the iterates of the SWLS algorithm could coincide with anchor points a_j for some $j = 1, 2, \ldots, m$, at which the algorithm is not defined. To deal with this scenario the authors of [6] show that if x^0 belongs to the following set

$$\mathcal{R} = \left\{ x \in \mathbb{R}^n : \; f\left(x\right) < \frac{\min_{1\leq j\leq m} d_j^2}{4} \right\},$$

then $x^k \notin \mathcal{A}$ for all $k \in \mathbb{N}$. The price to be paid for this stratagem is in finding such a starting point x^0, which increases computational cost of the SWLS. See [6] for more details about how to find a starting point x^0 which satisfies this assumption.

We have mentioned in Section 3 that Algorithm 1 can be written as an algorithm with steps on $(x, \mathbf{u})$, that is, to eliminate the auxiliary variables $\mathbf{v}$ and $\mathbf{w}$. Indeed, Algorithm 1 can be written equivalently as follows

Algorithm 1 (simplified).

Initialization. Start with any $\left(x^0, \mathbf{u}^0, \mathbf{v}^0; \mathbf{w}^0\right) \in \mathbb{R}^n \times \mathbb{R}^{nm} \times \mathbb{R}^{nm} \times \mathbb{R}^{nm}$, and $\rho_j > 0$, $j = 1, 2, \ldots, m$. Compute

$$x^1 = \frac{1}{m} \sum_{j=1}^{m} \left(a_j + v_j^0\right),$$

and, for each $j = 1, 2, \ldots, m$,

$$u_j^1 = P_{\mathcal{B}} \left(\frac{v_j^0 + \rho_j^{-1} w_j^0}{d_j} \right).$$

Main Loop. For each $k = 1, \ldots$ generate the sequence $\left\{\left(x^k, \mathbf{u}^k\right)\right\}_{k \in \mathbb{N}}$ as follows:

- Compute

$$x^{k+1} = \frac{1}{m} \sum_{j=1}^{m} \frac{1}{\rho_j} \left(\rho_j a_j + \rho_j d_j u_j^k + x^k - x^{k-1} \right). \tag{5.1}$$

- For each $j = 1, 2, \ldots, m$, compute

$$u_j^{k+1} = P_{\mathcal{B}} \left(u_j^k + \frac{1}{\rho_j d_j} \left(a_j - x^{k-1} \right) \right) \tag{5.2}$$

We present now some numerical comparison to the SWLS algorithm. Following [6], we will use the following setting to what we call an “experiment” (for $n = 2$ and $m \in \{3, 5, 7, 10\}$):

- Generate randomly the sensor locations a_j, $j = 1, 2, \ldots, m$, and the true source location x, from a uniform distribution over the box $[-1000, 1000] \times [-1000, 1000]$.

- Compute the ranges d_j, $j = 1, 2, \ldots, m$, using the relation (1.1), that is,

$$d_j = \|x - a_j\| + \varepsilon_j,$$

  where ε_j, $j = 1, 2, \ldots, m$, being generated from a normal distribution with zero mean and standard deviation 20.

- Generate random starting point again form uniform distribution over the box $[-1000, 1000] \times [-1000, 1000]$, $x^0 \in \mathbb{R}^2$ for both methods and $\mathbf{u}^0, \mathbf{v}^0, \mathbf{w}^0 \in \mathbb{R}^{2m}$ for Algorithm 1.

For both methods, we use the stopping criteria $\left\|\nabla f\left(x^k\right)\right\| < 10^{-6}$, where k is the iteration index. In our theoretical results we assumed that $\rho_j > 2$ for all $j = 1, 2, \ldots, m$, and in the experiments we took $\rho_j = 2.00001$ for all $j = 1, 2, \ldots, m$. All the experiments were conducted in MATLAB.

For each value of $m \in \{3, 5, 7, 10\}$, which is the number of sensors, we have conducted 1000 experiments and the results are presented in Table 1. In the second row we have recorded the number of experiments (out of 1000) for which SWLS achieved a lower function value f than Algorithm 1, while in the third row we record the opposite cases, that is, when Algorithm 1 achieved a lower function value than SWLS. The last two rows indicate the mean CPU time of each method. We will denote by $\hat{x}_S$ and $\hat{x}_A$ the solutions obtained by the SWLS method and Algorithm 1, respectively.

m	3	5	7	10
$\#(f(\hat{x}_S) < f(\hat{x}_A))$	264	114	61	18
$\#(f(\hat{x}_A) < f(\hat{x}_S))$	32	1	1	0
CPU - SWLS	0.0359	0.0365	0.0372	0.0376
CPU - Algorithm 1	0.0175	0.0155	0.0158	0.0179

Table 1: Comparison between the SWLS algorithm and Algorithm 1.

As can be clearly seen from Table 1, SWLS achieved lower function values in more experiments than Algorithm 1. Even though, it should be noted that the superiority of SWLS becomes very minor as m increases. We obtained that for $m \geq 20$, the two methods perform equally and produce the same solutions. On the other hand, a clear advantage of Algorithm 1 is the running time, which for any value of m, is at least two times faster than SWLS.

To test the robustness of Algorithm 1 against the choice of initial points we have repeated the 1000 experiments above but this time we initialized Algorithm 1 from two different, randomly selected starting points. The results are summarized in Table 2. It should be noted that we compared the function value in the solution obtained by the SWLS algorithm with the smaller function value obtained by the two runs of Algorithm 1.

m	3	5	7	10
$\#(f(\hat{x}_S) < f(\hat{x}_A))$	145	63	17	7
$\#(f(\hat{x}_A) < f(\hat{x}_S))$	44	5	3	0
CPU - SWLS	0.0361	0.0366	0.0376	0.0379
CPU - Algorithm 1	0.0353	0.0309	0.0316	0.0361

Table 2: Comparison between the SWLS algorithm and Algorithm 1.

As can be seen now, the mean CPU time of both methods is of the same order and the performance of Algorithm 1 is much more robust than SWLS (see Table 1).

Another interesting phenomena that we would like to point out here relates to the penalty parameter ρ. We have proved that convergence can be guaranteed for $\rho_j > 2$, $j = 1, 2, \ldots, m$ and in the previous experiments we used parameter values that satisfy this requirement. It should be noted that the algorithm is well-defined for any $\rho_j > 0$, $j = 1, 2, \ldots, m$. We set $m = 3$ and conducted 1000 experiments with different values for $\rho_j > 0$, $j = 1, 2, \ldots, m$. For each value, we compared the performance of Algorithm 1 to the SWLS algorithm in terms of the objective function value. In the following table we see that if $\rho_j = 1$, which does not satisfy the theoretical requirement, then we get slightly better results, but for $\rho_j \geq 3$ we observe that the performance deteriorates.

ρ	$\#(f(\hat{x}_S) < f(\hat{x}_A))$	$\#(f(\hat{x}_A) < f(\hat{x}_S))$
1	249	33
3	294	31
4	323	28
6	374	27
8	414	27

Table 3: Comparison between the SWLS algorithm and Algorithm 1 (different values of ρ).

From Table 3 we see that violating the theoretical requirement on the penalty parameter ρ does not appear to affect the numerical performance of the algorithm, however in few experiments we do obtained a better solution in terms of lower objective function value. It should be noted that we observed the same phenomena for larger values of m, that is, to larger number of sensors.

References

[1] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Math. Program.*, 116(1-2, Ser. B):5–16, 2009.

[2] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality. *Math. Oper. Res.*, 35(2):438–457, 2010.

[3] H. Attouch, J. Bolte, and B. F. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Math. Program.*, 137(1-2, Ser. A):91–129, 2013.

[4] A. Beck and D. Pan. On the solution of the GPS localization and circle fitting problems. *SIAM J. Optim.*, 22(1):108–134, 2012.

[5] A. Beck, P. Stoica, and J. Li. Exact and approximate solutions of source localization problems. *IEEE Trans. Signal Process.*, 56(5):1770–1778, 2008.

[6] A. Beck, M. Teboulle, and Z. Chikishev. Iterative minimization schemes for solving the single source localization problem. *SIAM J. Optim.*, 19(3):1397–1416, 2008.

[7] J. Bolte, A. Daniilidis, and A. Lewis. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM J. Optim.*, 17(4):1205–1223, 2006.

[8] J. Bolte, A. Daniilidis, O. Ley, and L. Mazet. Characterizations of Łojasiewicz inequalities: subgradient flows, talweg, convexity. *Trans. Amer. Math. Soc.*, 362(6):3319–3363, 2010.

[9] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Math. Program.*, 146(1-2, Ser. A):459–494, 2014.

[10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, 2011.

[11] 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. Signal Process.*, 52(4):1121–1128, 2004.

[12] F. Freschi. Localization of sources of brain activity: A MILP approach. *IEEE Trans. Magn.*, 46(8):3429–3432, 2010.

[13] R. Hesse, D. R. Luke, and P. Neumann. Alternating projections and Douglas-Rachford for sparse affine feasibility. *IEEE Trans. Signal Process.*, 62(18):4868–4881, 2014.

[14] K. Kurdyka. On gradients of functions definable in o-minimal structures. *Ann. Inst. Fourier (Grenoble)*, 48(3):769–783, 1998.

[15] S. Łojasiewicz. Une propriété topologique des sous-ensembles analytiques réels. In *Les Équations aux Dérivées Partielles (Paris, 1962)*, pages 87–89. Éditions du Centre National de la Recherche Scientifique, Paris, 1963.

[16] A. Lombard, Y. Zheng, H. Buchner, and W. Kellermann. Tdoa estimation for multiple sound sources in noisy and reverberant environments using broadband independent component analysis. *IEEE T. Audio Speech*, 19(6):1490–1503, 2011.

[17] D. R. Luke, N. H. Thao, and M. T. Tam. Quantitative convergence analysis of iterated expansive, set-valued mappings. arXiv:1563306, 2016.

[18] R. T. Rockafellar. *Convex analysis.* Princeton Mathematical Series, No. 28. Princeton University Press, Princeton, N.J., 1970.

[19] R. T. Rockafellar and R. J.-B. Wets. *Variational Analysis*, volume 317 of *Grundl. Math. Wiss.* Springer-Verlag, Berlin, 1998.

[20] P. Stoica and J. Li. Source localization from range-difference measurements. *IEEE Signal Processing Magazine*, 23(6):63–66, 2006.

[21] P. Tseng. Second-order cone programming relaxation of sensor network localization. *SIAM J. Optim.*, 18:156–185, 2007.