OJMO

Open Journal of Mathematical Optimization

Eyal Gur & Shoham Sabach

Network Localization and Multi-Dimensional Scaling: Escaping Saddles and a Local Optimality Condition

Volume 6 (2025), article no. 6 (18 pages)

https://doi.org/10.5802/ojmo.42

Article submitted on March 24, 2024, revised on January 29, 2025,
accepted on April 22, 2025.

CENTRE MERSENNE

Open Journal of Mathematical Optimization is a member of the
Centre Mersenne for Open Scientific Publishing
http://www.centre-mersenne.org/
e-ISSN: 2777-5860

Network Localization and Multi-Dimensional Scaling: Escaping Saddles and a Local Optimality Condition

Eyal Gur
Faculty of Data and Decision Sciences, Technion – Israel Institute of Technology, Haifa 3200003, Israel
eyal.gur@campus.technion.ac.il

Shoham Sabach
Faculty of Data and Decision Sciences, Technion – Israel Institute of Technology, Haifa 3200003, Israel
ssabach@technion.ac.il

Abstract

In this paper, we focus on a class of problems characterized by solving a non-linear least squares minimization, for approximating a norm of a linear transformation. These problems are characterized by their non-convex and non-smooth nature, presenting challenges in finding (locally) optimal solutions. While existing optimization algorithms mostly concentrate on finding critical points of the associated least squares objective function, these functions often possess multiple non-global local minima and saddle points. These problems find wide applications in various domains, and we focus our attention on two challenging problems: Wireless Sensor Network Localization and Multi-Dimensional Scaling. We establish that non-differentiable points correspond to maximum or saddle points, and we provide a constructive approach to determine descent directions at these points. Leveraging this, we propose a straightforward procedure to escape non-differentiable saddle points that is applicable in either centralized or distributed computational setting. Furthermore, we develop a necessary condition for differentiable points to be local minimizers, by exploiting the structure of the objective function of these problems.

Digital Object Identifier 10.5802/ojmo.42

Keywords sensor network localization; multi-dimensional scaling; criticality; optimality condition; saddle point.

Acknowledgments The work of the authors was supported by the Israel Science Foundation, Israel, ISF 2480-21.

1 Introduction

In this paper, we address the task of minimizing a non-linear, non-convex and non-smooth least squares function $\mathcal{F}\colon \mathbb{R}^q \to [0,\infty)$, defined as

$$\underset{\mathbf{x}\in\mathbb{R}^q}{\text{minimize}} \quad \mathcal{F}(\mathbf{x}) \equiv \sum_{l=1}^{N} (\|\mathbf{x}_{i_l} - \mathbf{x}_{j_l}\| - \delta_l)^2, \tag{1}$$

where $\|\cdot\|$ represents the Euclidean norm, $\mathbf{x} \in \mathbb{R}^q$ $(q = n \cdot N)$ is the vector variable, $\mathbf{x}_{i_l}, \mathbf{x}_{j_l} \in \mathbb{R}^n$ $(l = 1, 2, \ldots, N)$ are some sub-vectors of vector $\mathbf{x} \in \mathbb{R}^q$, and $\delta_l \in \mathbb{R}$ are given scalars. Optimization problems of this form arise in various applied contexts, such as signal processing and unsupervised learning. Notable examples include the widely studied applications of Sensor Network Localization (SNL) and Multi-Dimensional Scaling (MDS), which are further discussed below.

The optimization model in (1) is non-convex and non-smooth (i.e., non-differentiable), which presents challenges in finding (locally) optimal solutions. Recent advancements in non-convex optimization have mainly concentrated on finding critical points [4, 5, 10], which could technically be (locally) optimal solutions but also saddle points (or even maximum points). Therefore, a grand challenge of optimization theory and practice is to avoid saddle points. This work aims at advancing the research on this challenging question by studying non-convex optimization problems as described in (1) and exploiting the more specific structure.

We note that while this paper focuses on the objective function $\mathcal{F}$ formulated in (1), it is a particular instance of a more general least squares minimization problem, where the norm term is replaced with $\|\mathbf{A}_l\mathbf{x} + \mathbf{b}_l\|$ for $\mathbf{A}_l \in \mathbb{R}^{q_l \times q}$ and $\mathbf{b}_l \in \mathbb{R}^{q_l}$ given data matrices and vectors.

Optimization problems of this form arise in signal processing tasks that involve solving non-linear inverse problems [28]. While focusing on Problem (1) instead of the more general case may initially appear limiting, our theoretical framework can be readily extended to other problems. By thoroughly examining the structure in (1), we provide a clear and convenient framework for analysis and facilitate insightful discussions, enabling the extension and generalization of our findings to a broader range of related problems encompassed by the general class.

The main goal of this paper it to deepen our understanding of optimization problems with structure as given in (1). We fully characterize its set of critical points and leverage this knowledge to also advance the algorithmic and practical fronts. Our main contributions are now summarized and categorized into two areas: the theoretical analysis of the function's landscape, and computational aspects in both centralized and distributed settings.

i. In Section 3, we prove that all non-differentiable points of $\mathcal{F}$ have an *explicit and easy-to-find* descent direction that can be utilized to decrease the objective value.
ii. In Section 4, we develop a procedure that escapes non-differentiable saddle points, thereby preventing minimization algorithms from getting trapped in a subset of the non-optimal critical points. This escape procedure can be implemented in both centralized and distributed computational settings.
iii. In Section 5, we utilize the classical second-order necessary optimality condition to formulate a condition for a differentiable critical point to be a local minimum point. This condition can be easily verified in both centralized and distributed computational settings.

2 Motivating Applications and Literature Review

In this section, we will first discuss two prominent applications that provide the motivation for studying the problem of solving the non-linear composite norm equations with the structure as given in Problem (1). Then, we will survey some relevant existing literature.

2.1 The Problems of MDS and SNL

The first application is Multi-Dimensional Scaling (MDS), which is a popular tool for dimensionality reduction and data visualization [38]. Formally, given a symmetric matrix $\mathbf{D} \in \mathbb{R}^{K\times K}$, where $\mathbf{D}_{ij} = \mathbf{D}_{ji}$ denotes the dissimilarity between two data points $\mathbf{o}_i, \mathbf{o}_j \in \mathbb{R}^p$ defined mathematically as $\mathbf{D}_{ij} = \|\mathbf{o}_i - \mathbf{o}_j\|$. MDS aims to find lower-dimensional representations $\mathbf{x}_i \in \mathbb{R}^n$, $n < p$, for each data point $\mathbf{o}_i \in \mathbb{R}^p$. These representations should satisfy the condition that the distances $\|\mathbf{x}_i - \mathbf{x}_j\|$ approximate the original dissimilarity $\mathbf{D}_{ij}$ within a certain tolerance $\epsilon_{ij} \in \mathbb{R}$. In essence, MDS seeks to find K vectors $\mathbf{x}_i \in \mathbb{R}^n$ such that [13, 14]

$$\delta_{ij} \equiv \mathbf{D}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| + \epsilon_{ij}, \quad i, j \in \{1, 2, \ldots, K\}.$$

We easily see that MDS fits into the framework of (1) by representing $\mathbf{x} \in \mathbb{R}^{nK}$ as the concatenation of all unknowns $\mathbf{x}_i \in \mathbb{R}^n$, $i = 1, 2, \ldots, K$, into a single vector. We mention that in the context of the MDS problem, the objective function (1) is called the *stress function* [13].

In the second application, Sensor Network Localization (SNL), the goal is to find the location of each sensor in a deployed sensor network, utilizing distance measurements between neighboring sensors [3]. Formally, we consider a set of K sensors, each located at an unknown location $\mathbf{x}_i \in \mathbb{R}^n$, $i = 1, 2, \ldots, K$. Given a set $\mathcal{E}$ comprising pairs of neighboring sensors i and j, with positive noisy distance measurements $\delta_{ij} > 0$ between them, the SNL problem is typically formulated as finding K vectors $\mathbf{x}_i \in \mathbb{R}^n$ such that [31, 33]

$$\delta_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| + \epsilon_{ij}, \quad (i, j) \in \mathcal{E},$$

where ϵ_{ij} denotes a distance measurement noise. Notice that we result with a similar set of equations as in MDS, and therefore it fits into the setting of equations given in (1).

A few other well-studied applications that fall under non-linear least-squares formulation (1) are the Source Localization problem [9, 18, 26] and the Phase Retrieval problem [15, 37]. Before presenting our study on this optimization model, we review existing works that are relevant for the task of understanding the critical points of the function $\mathcal{F}$.

2.2 First-Order Criticality

Here, we survey some existing literature about the notion of critical points, focusing on non-smooth functions. To this end, we recall that a first-order *critical point* for a non-convex and non-smooth function $\mathcal{F}$ is a point $\mathbf{x} \in \mathbb{R}^q$

for which the zero vector belongs to its limiting sub-differential set [27], denoted by $\partial\mathcal{F}(\mathbf{x})$, i.e., $\mathbf{0}_q \in \partial\mathcal{F}(\mathbf{x})$. In the context of non-smooth analysis, a critical point can be either a differentiable or a non-differentiable point. When a point $\mathbf{x} \in \mathbb{R}^q$ is differentiable, then the condition $\mathbf{0}_q \in \partial\mathcal{F}(\mathbf{x})$ reduces to the equation $\mathbf{0}_q = \nabla\mathcal{F}(\mathbf{x})$, where $\nabla\mathcal{F}(\mathbf{x})$ is the gradient of $\mathcal{F}$ at $\mathbf{x} \in \mathbb{R}^q$.

We mention that, even though the problem is non-convex and non-smooth, criticality serves as a necessary condition for local optimality, implying that any global or local minimum point must also be a first-order critical point of the objective function. Therefore, in this paper, we first aim at characterizing the set of critical points of $\mathcal{F}$ and explore their relationship with (locally) optimal minimum points.

In order to better understand the critical points of the given function at hand, beyond the definition of the sub-differential set, it will be beneficial to exploit different structural properties of the function $\mathcal{F}$. One line of research that can be relevant for our case is from the domain of Difference-of-Convex (DC) programming. A general DC programming problem can be expressed in the following form

$$\min_{\mathbf{x}\in\mathbb{R}^q} \{\Psi(\mathbf{x}) \equiv \varphi(\mathbf{x}) - \psi(\mathbf{x})\}, \tag{DC}$$

where $\varphi, \psi\colon \mathbb{R}^q \to (-\infty, \infty]$ are convex and (possibly) non-smooth. Indeed, we easily see that Problem (1) can be formulated in the form by defining [39]

$$\varphi(\mathbf{x}) = \sum_{l=1}^{N}\left(\|\mathbf{x}_{i_l} - \mathbf{x}_{j_l}\|^2 + \delta_l^2\right) \quad \text{and} \quad \psi(\mathbf{x}) = 2\sum_{l=1}^{N}\delta_l\|\mathbf{x}_{i_l} - \mathbf{x}_{j_l}\|.$$

By framing Problem (1) as a DC programming problem, we can leverage existing insights from this domain to shed some light on the critical points of the function $\mathcal{F}$. For instance, in DC programming problems, a necessary condition for optimality is known as a *DC-critical point* [29], which is a point $\mathbf{x} \in \mathbb{R}^q$ satisfying $\partial\varphi(\mathbf{x}) \cap \partial\psi(\mathbf{x}) \neq \emptyset$, where here ∂ denotes the sub-differential set of a convex function[1]. The concept of criticality in DC programming has been studied in several papers, see for instance [34, 35]. See also [24, 30] for a concise introduction to this notion.

However, when it comes to characterizing the minimum points of Problem (1), it is important to note that the notion of DC-criticality does not provide a better understanding than the classical notion of criticality, as DC-criticality can not differentiate between minimum and maximum points. For example, we consider the one-dimensional DC function $\Psi(x) = x^2 - |x|$. In this case, $x = 0$ is a DC-critical point that is also a maximum point.

A more restrictive concept than DC-criticality is a *directional-stationary point* [29], often referred to as a d-stationary point, which is a point $\mathbf{x} \in \mathbb{R}^q$ where no feasible descent directions exist. We point out that being a d-stationary point is a necessary condition for optimality but not a sufficient one. For instance, consider the function $\left(y - x^2\right)^2 + x^5$ at the point $(0, 0)$. Although there are no descent directions at this point, it is not a local minimizer, as the function decreases along the curve $\left(t, t^2\right)$ for $t < 0$.[2]

We mention here that a *descent direction* of a function f at a point $\mathbf{x}$ is any direction $\mathbf{d} \in \mathbb{R}^q$ along which the function value decreases. In the smooth setting (which is not the case here), where the gradient $\nabla f(\mathbf{x})$ exits, this concept corresponds to directions satisfying $\nabla f(\mathbf{x})^T\mathbf{d} < 0$. For precise definitions, see Section 3.

In the context of Problem (DC), the work [8] shows that if the function φ of Problem (DC) is smooth, then any d-stationary point is a differentiable point of the function Ψ. This result implies that any optimal solution of Problem (1) must be a point $\mathbf{x} \in \mathbb{R}^q$ where the gradient $\nabla\mathcal{F}(\mathbf{x})$ exists and $\nabla\mathcal{F}(\mathbf{x}) = \mathbf{0}_q$. It is important to note that the function $\mathcal{F}$ of Problem (1) is non-smooth, so its gradient is not defined for its non-differentiable points. Utilizing the notion of d-stationarity, any critical point of Problem (1) that is not a d-stationary point has at least one descent direction and is therefore not a minimum point of the problem. However, determining such descent directions is a challenging task as currently there is no clear way to identify such directions for Problem (1).

To conclude, even though framing the function $\mathcal{F}$ as a DC function exploits a certain structure, the understanding of its critical points remains very limited. In the following sections, we show that the structure of $\mathcal{F}$ is generous enough to enable us to characterize its critical points, develop a simple procedure that escapes non-differentiable saddle points, and even devise a necessary condition for a differentiable critical point to qualify as a local minimum point.

---

[1] In the convex setting, the limiting sub-differential coincides with the "regular" sub-differential.

[2] We thank the anonymous reviewer for this valuable observation.

2.3 Escaping Saddle Points

As mentioned in Section 1, in this paper we develop a procedure that escapes non-differentiable saddle point of the function $\mathcal{F}$ in Problem (1). To this end, *saddle points* are defined as points that satisfy the first-order criticality condition, but are not local minimum or maximum points.

The literature makes a distinction between *strict* saddle points and *non-strict* saddle points [41]. In the non-smooth setting, strict saddles are saddles that have a descent direction (see Section 3 for the exact definition of descent directions). In the smooth setting, strict saddles are defined as those whose Hessain matrix has at least one negative eigenvalue. Any other saddle point is called non-strict. In Section 3, we compliment the fact that all non-differentiable points of $\mathcal{F}$ are either maximum or saddle points as observed in [8], by proving that these points have an explicit descent direction, implying that all non-differentiable saddles are strict.

We mention that strict saddles can be distinguished from minimizers using first-order or second-order information, while non-strict saddles cannot. Hence, strict saddles can be escaped, or evaded.

It is well-known that for twice-differentiable functions, if the Hessian matrix at a critical point has a negative eigenvalue (strict saddle), then the corresponding eigenvector provides a direction to decrease the objective function. This property enables algorithms to escape strict saddles by following this direction or a noisy variant (e.g., [11, 16, 41]). Several works have showed that certain objectives posses the *strict saddle property* – where all critical points are either local minima or strict saddles, with no non-strict saddle points. For such functions, local search algorithms can exploit the negative curvature to escape saddle points effectively.

For example, in [41] the authors prove that shallow linear and twice-differentiable neural networks satisfy the strict saddle property, and [11] introduces an algorithm that escapes strict saddles by computing the Hessian matrix at each iteration, computing its minimal eigenvalue, and identifying an explicit descent direction satisfying some second-order conditions. Other works that prove similar results are, for instance, [16, 22, 25] and more recently [23]. We also note that while the literature suggests that strict saddles can be escaped under some conditions, local minima and non-strict saddles remain inescapable using current first-order methods (see, for example, [2, 1]).

All these works mentioned above rely on assumptions that do not apply in our setting. Specifically, they assume that the objective function is twice-differentiable and satisfies the strict saddle property – assumptions that do not hold for our non-differentiable objective function. Furthermore, many of these methods depend on second-order information, such as computing the Hessian matrix and its minimal eigenvalue, tasks that require centralized implementations. While feasible for some applications, such centralized approaches are unsuitable for sensor network problems, which typically require distributed computations.

The non-differentiable case is far more challenging. Prior works have mainly studied differentiable regions of the function or imposed restrictive assumptions, leaving the characterization of non-differentiable critical points largely unexplored. While such points have zero Lebesgue measure, they cannot be ignored, as some gradient-based algorithms may still encounter them during optimization [40].

To the best of our knowledge, the literature on non-differentiable critical points and their characterization, particularly in distributed settings, is underdeveloped for the type of problem addressed in our paper. A notable exception is [40], which examines a one-hidden-layer neural network with non-differentiable ReLU-like activations and MSE loss. By leveraging the specific structure of the problem, the authors identify critical points satisfying certain first-order non-differentiable conditions and use these conditions to escape strict saddle points (whether differentiable or not).

Similarly, our approach in this paper explores the specific non-differentiable structure of the problem under investigation. We provide a characterization of critical points (both differentiable and non-differentiable) and propose a method for escaping non-differentiable saddles. Furthermore, to the best of our knowledge, our method is the first to be applicable in a distributed setting.

3 Characterization of Extremum Points

In this section, we explore the extremum points of the Problem (3), which will provide an important and useful ground for the understanding of critical points in the following sections.

To achieve this goal, since the function $\mathcal{F}$ is non-differentiable, we first recall the notion of directional derivatives. Let $\phi\colon \mathbb{R}^q \to (-\infty, \infty]$ be a proper function and let $\mathbf{x} \in \operatorname{int}(\operatorname{dom}(\phi))$. The directional derivative in

the direction of the vector $\mathbf{d} \in \mathbb{R}^q$ is defined by

$$\phi'(\mathbf{x};\mathbf{d}) \equiv \lim_{\epsilon \to 0^+} \frac{\phi(\mathbf{x}+\epsilon\mathbf{d}) - \phi(\mathbf{x})}{\epsilon}. \tag{2}$$

The function ϕ is said to be *differentiable* at $\mathbf{x} \in \mathbb{R}^q$ if the gradient vector $\nabla\phi(\mathbf{x}) \in \mathbb{R}^q$ exists. If ϕ is continuously differentiable over an open set $U \subseteq \mathbb{R}^q$ that contains the point $\mathbf{x} \in \mathbb{R}^q$, then $\phi'(\mathbf{x};\mathbf{d}) = \nabla\phi(\mathbf{x})^T\mathbf{d}$ for any $\mathbf{d} \in \mathbb{R}^q$ [6].

For the sake of simplicity in the subsequent analysis and for ease of index notation, we express Problem (1) using the terminology of the SNL problem. In other words, we rewrite Problem (1) equivalently as

$$\min_{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_K \in \mathbb{R}^n} \mathcal{F}(\mathbf{x}) \equiv \sum_{(i,j)\in\mathcal{E}} \left(\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij}\right)^2, \tag{3}$$

and we use this formulation interchangeably with Problem (1).

Due to the non-differentiability of the norm function at $\mathbf{0}_n$, the non-differentiable points of $\mathcal{F}$ are precisely the points $\mathbf{x} \in \mathbb{R}^{nK}$ where $\mathbf{x}_i = \mathbf{x}_j \in \mathbb{R}^n$ for some pair $(i,j) \in \mathcal{E}$. For simplicity of developments, for any pair $(i,j) \in \mathcal{E}$, we define the function $\mathcal{F}_{ij}\colon \mathbb{R}^{nK} \to \mathbb{R}$ as

$$\mathcal{F}_{ij}(\mathbf{x}) \equiv \left(\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij}\right)^2, \tag{4}$$

and we notice that $\mathbf{x} \in \mathbb{R}^{nK}$ is a non-differentiable point of $\mathcal{F}_{ij}$ if and only if $\mathbf{x}_i = \mathbf{x}_j$. Following (4), it holds that

$$\mathcal{F}(\mathbf{x}) = \sum_{(i,j)\in\mathcal{E}} \mathcal{F}_{ij}(\mathbf{x}), \tag{5}$$

and notice that a point $\mathbf{x} \in \mathbb{R}^{nK}$ is a non-differentiable point of $\mathcal{F}$ if and only if it is a non-differentiable point of $\mathcal{F}_{ij}$, for some pair $(i,j) \in \mathcal{E}$.

We begin with a few simple properties of the directional derivative of $\mathcal{F}$. To this end, for any vector $\mathbf{d} \in \mathbb{R}^{nK}$ we denote by $\mathbf{d}_i$, $i = 1,2,\ldots,K$, the sub-vector obtained from $\mathbf{d}$ by taking its $n(i-1)+1$ to $n \cdot i$ coordinates.

▶ **Lemma 1.** *Let $(i,j) \in \mathcal{E}$.*

i. *Let $\mathbf{x} \in \mathbb{R}^{nK}$ be a non-differentiable point of $\mathcal{F}_{ij}$. Then, for any $\mathbf{d} \in \mathbb{R}^{nK}$, it holds that*

$$\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) = -2\delta_{ij}\|\mathbf{d}_i - \mathbf{d}_j\|.$$

In particular, $\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) < 0$ if $\mathbf{d}_i \neq \mathbf{d}_j$ and $\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) = 0$ otherwise.

ii. *Let $\mathbf{x} \in \mathbb{R}^{nK}$ be a point satisfying $\mathcal{F}_{ij}(\mathbf{x}) = 0$. Then, $\mathcal{F}_{ij}(\epsilon\mathbf{x}) > 0$ for any $\epsilon \neq 1$.*

Proof.

i. We recall that $\mathbf{x}_i = \mathbf{x}_j$ for any non-differentiable point of $\mathcal{F}_{ij}$. From (2) we have

$$\begin{aligned}\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) &= \lim_{\epsilon\to 0^+} \frac{\mathcal{F}_{ij}(\mathbf{x}+\epsilon\mathbf{d}) - \mathcal{F}_{ij}(\mathbf{x})}{\epsilon} = \lim_{\epsilon\to 0^+} \left(\epsilon\|\mathbf{d}_i - \mathbf{d}_j\|^2 - 2\delta_{ij}\|\mathbf{d}_i - \mathbf{d}_j\|\right) \\ &= -2\delta_{ij}\|\mathbf{d}_i - \mathbf{d}_j\|,\end{aligned}$$

and the result immediately follows.

ii. Notice that $\mathcal{F}_{ij}(\mathbf{x}) = 0$ if and only if $\|\mathbf{x}_i - \mathbf{x}_j\| = \delta_{ij}$. Now, the point $\epsilon\mathbf{x}$ for any $\epsilon \neq 1$ satisfies $\mathcal{F}_{ij}(\epsilon\mathbf{x}) = \left(\epsilon\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij}\right)^2 = \delta_{ij}^2(\epsilon-1)^2 > 0$, as required. ◀

Recall that $\mathbf{x} \in \mathbb{R}^q$ is a *local minimum* point of a function $\phi\colon \mathbb{R}^q \to (-\infty,\infty]$, if $\phi(\mathbf{x}) \leq \phi(\mathbf{y})$ for all $\mathbf{y} \in \mathrm{dom}(\phi)$ such that $\|\mathbf{x}-\mathbf{y}\| \leq \epsilon$ for some $\epsilon > 0$. In addition, a local minimum point $\mathbf{x} \in \mathbb{R}^q$ of ϕ is *global*, if it attains the minimal value of ϕ over its domain. Similarly, we define *local* and *global maximum* points. Moreover, an *extremum* point of ϕ is either a local minimum or a local maximum point. Of course, any global extremum point (if exists) is also a local extremum point.

Before we characterize the extremum points of $\mathcal{F}$, we recall the notion of descent and ascent directions. We say that $\mathbf{d} \in \mathbb{R}^q$ is a *descent direction* of $\phi\colon \mathbb{R}^q \to (-\infty,\infty]$ if $\phi'(\mathbf{x};\mathbf{d})$ exists and is negative. Similarly, if it is positive then $\mathbf{d}$ is an *ascent direction*. This notion is important since it is well-known (see, for instance, [7, Lemma 8.2]) that if $\mathbf{d} \in \mathbb{R}^q$ is a descent direction of ϕ at $\mathbf{x} \in \mathbb{R}^q$, then there exists some $\bar{\epsilon} > 0$ such that $\phi(\mathbf{x}+\epsilon\mathbf{d}) < \phi(\mathbf{x})$ for all $\epsilon \in (0,\bar{\epsilon}]$. Hence, $\mathbf{x}$ is not a minimum point[3]. Similar results hold for ascent directions.

[3] Conversely, if there exists some $\bar{\epsilon} > 0$ such that $\phi(\mathbf{x}+\epsilon\mathbf{d}) \leq \phi(\mathbf{x})$ for all $\epsilon \in (0,\bar{\epsilon}]$, then if the directional derivative exists, it follows that $\phi'(\mathbf{x};\mathbf{d}) \leq 0$ and hence $\mathbf{d}$ is a non-ascent direction (that is, $\mathbf{d}$ is either a descent direction or that the directional derivative is 0). Similarly, if $\phi(\mathbf{x}+\epsilon\mathbf{d}) \geq \phi(\mathbf{x})$ we obtain that $\mathbf{d}$ is a non-descent direction.

We recall that a point $\mathbf{x} \in \mathbb{R}^q$ is a *stationary point* of a function $\phi\colon \mathbb{R}^q \to (-\infty, \infty]$ if the gradient $\nabla\phi(\mathbf{x})$ exits and is the vector of all zeros. We should mention that in the literature the terms *critical point* and *stationary point* are sometimes used interchangeably. However, in the non-smooth setting the terms are distinguished, and throughout this paper the term *stationary point* is specifically used to refer to a *differentiable critical point*, as defined in Section 2.2.

Now, we are ready to provide a characterization of the extremum points of the functions $\mathcal{F}_{ij}$. To this end, for any point $\mathbf{x} \in \mathbb{R}^{nK}$, we denote by $\mathcal{B}_{ij}[\mathbf{x}] \subset \mathbb{R}^{nK}$ the closed ball centered at $\mathbf{x}$ with radius $\delta_{ij} > 0$.

▶ **Lemma 2.**

i. *Let $(i,j) \in \mathcal{E}$. A point $\mathbf{x} \in \mathbb{R}^{nK}$ is a stationary point of $\mathcal{F}_{ij}$ if and only if it is a global minimum point of $\mathcal{F}_{ij}$.*

ii. *Let $(i,j) \in \mathcal{E}$. Then, $\mathbf{x} \in \mathbb{R}^{nK}$ is a non-differentiable point of $\mathcal{F}_{ij}$ if and only if it is a local maximum point of $\mathcal{F}_{ij}$.*

iii. *Any non-differentiable point of $\mathcal{F}$ is a local maximum point of $\mathcal{F}_{ij}$ for some $(i,j) \in \mathcal{E}$.*

Proof.

i. Recall that the differentiable points of $\mathcal{F}_{ij}$ are exactly the points $\mathbf{x} \in \mathbb{R}^{nK}$ for which $\mathbf{x}_i \neq \mathbf{x}_j$. In particular, the gradient $\nabla\mathcal{F}_{ij}(\mathbf{x})$ exists and simple calculation show that

$$\nabla_{\mathbf{x}_i}\mathcal{F}_{ij}(\mathbf{x}) = \frac{2(\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij})}{\|\mathbf{x}_i - \mathbf{x}_j\|}(\mathbf{x}_i - \mathbf{x}_j) = -\nabla_{\mathbf{x}_j}\mathcal{F}_{ij}(\mathbf{x}),$$

where $\nabla_{\mathbf{x}_i}\mathcal{F}_{ij}$ is the gradient of the partial function $\mathbf{x}_i \mapsto \mathcal{F}_{ij}(\mathbf{x})$ (which can also be viewed as the sub-vector of $\nabla\mathcal{F}_{ij}$ corresponding to the coordinates of $\mathbf{x}_i \in \mathbb{R}^n$). Also, $\nabla_{\mathbf{x}_l}\mathcal{F}_{ij}(\mathbf{x}) = \mathbf{0}_n$ for any $l \in \{1, 2, \ldots, K\}$ such that $l \notin \{i, j\}$. Therefore, $\nabla\mathcal{F}_{ij}(\mathbf{x})$ is the vector of all zeros (i.e., $\mathbf{x}$ is a stationary point) if and only if $\|\mathbf{x}_i - \mathbf{x}_j\| = \delta_{ij}$ if and only if $\mathcal{F}_{ij}(\mathbf{x}) = 0$ if and only if $\mathbf{x}$ is a global minimum point of the non-negative function $\mathcal{F}_{ij}$.

ii. Let $\mathbf{x} \in \mathbb{R}^{nK}$ be a non-differentiable point of $\mathcal{F}_{ij}$ and we will prove that it is a local maximum point of $\mathcal{F}_{ij}$. To this end, we prove that $\mathcal{F}_{ij}(\mathbf{y}) \leq \mathcal{F}_{ij}(\mathbf{x})$ for any $\mathbf{y} \in \mathcal{B}_{ij}[\mathbf{x}]$. Recall that for any non-differentiable point it holds that $\mathbf{x}_i = \mathbf{x}_j$, hence we get from the triangle inequality

$$\|\mathbf{y}_i - \mathbf{y}_j\| \leq \|\mathbf{y}_i - \mathbf{x}_i\| + \|\mathbf{y}_j - \mathbf{x}_j\| \leq 2\delta_{ij},$$

where the last inequality is due to the fact that $\mathbf{y} \in \mathcal{B}_{ij}[\mathbf{x}]$. Hence, $\|\mathbf{y}_i - \mathbf{y}_j\| - 2\delta_{ij} \leq 0$ and we get

$$\mathcal{F}_{ij}(\mathbf{y}) = (\|\mathbf{y}_i - \mathbf{y}_j\| - \delta_{ij})^2 = \|\mathbf{y}_i - \mathbf{y}_j\|(\|\mathbf{y}_i - \mathbf{y}_j\| - 2\delta_{ij}) + \delta_{ij}^2 \leq \delta_{ij}^2 = \mathcal{F}_{ij}(\mathbf{x}), \tag{6}$$

where the last equality follows from the fact that $\mathbf{x}_i = \mathbf{x}_j$, and the required result follows.

For the converse direction, we will prove that if $\mathbf{x} \in \mathbb{R}^{nK}$ is a local maximum point of $\mathcal{F}_{ij}$, then it is also a non-differentiable point of $\mathcal{F}_{ij}$. More precisely, we will prove that $\mathbf{x}_i = \mathbf{x}_j$. Assume on the contrary that $\mathbf{x}$ is differentiable point. Meaning, the gradient $\nabla\mathcal{F}_{ij}(\mathbf{x})$ exists and $\mathbf{x}_i \neq \mathbf{x}_j$. From the continuity of $\mathcal{F}_{ij}$, there exists an open set $U \subseteq \mathbb{R}^{nK}$ such that $\mathbf{y}_i \neq \mathbf{y}_j$ for all $\mathbf{y} \in U$, and $\mathcal{F}_{ij}$ is continuously differentiable over U.

Now, either $\nabla\mathcal{F}_{ij}(\mathbf{x})$ is a non-zero vector, or it is the vector of all zeros. If $\nabla\mathcal{F}_{ij}(\mathbf{x}) \neq \mathbf{0}_{nK}$, then $\nabla\mathcal{F}_{ij}(\mathbf{x})$ is an ascent direction of $\mathcal{F}_{ij}$ at $\mathbf{x}$ (since $\nabla\mathcal{F}_{ij}(\mathbf{x})^T\nabla\mathcal{F}_{ij}(\mathbf{x}) = \|\nabla\mathcal{F}_{ij}(\mathbf{x})\|^2 > 0$), in contrary to the assumption that $\mathbf{x}$ is a local maximum point. If $\nabla\mathcal{F}_{ij}(\mathbf{x}) \equiv \mathbf{0}_{nK}$, then from item (i) it follows that $\mathcal{F}_{ij}(\mathbf{x}) = 0$. From Lemma 1(ii) it follows that any neighborhood containing $\mathbf{x}$ attains a function value that is strictly greater than 0, which again contradicts the assumption that $\mathbf{x}$ is a local maximum point.

iii. Recall that if $\mathcal{F}$ is non-differentiable at $\mathbf{x} \in \mathbb{R}^{nK}$, then there exists at least one pair $(i,j) \in \mathcal{E}$ such that $\mathbf{x}_i = \mathbf{x}_j$, which means that $\mathbf{x}$ is also a non-differentiable point of $\mathcal{F}_{ij}$. The result now follows from item (ii). ◀

Based on this result, we would like to provide a few more direct consequences regarding the extremum points of $\mathcal{F}$.

▶ Remark 3.

i. All local minimum points of $\mathcal{F}_{ij}$, for any $(i,j) \in \mathcal{E}$, are necessarily global. Indeed, let $\mathbf{x} \in \mathbb{R}^{nK}$ be a local minimum point. From Lemma 2(ii) it follows that $\mathbf{x}$ must be a differentiable point of $\mathcal{F}_{ij}$. In particular, the gradient $\nabla\mathcal{F}_{ij}(\mathbf{x})$ exists at any local minimum of $\mathcal{F}_{ij}$. If $\nabla\mathcal{F}_{ij}(\mathbf{x}) \neq \mathbf{0}_{nK}$, then surely $-\nabla\mathcal{F}_{ij}(\mathbf{x})$ is a descent direction, which contradicts the fact that it is a local minimum. If $\nabla\mathcal{F}_{ij}(\mathbf{x}) \equiv \mathbf{0}_{nK}$, then it is a global minimum point (see Lemma 2(i)).

ii. The function $\mathcal{F}_{ij}$ has no global maximum points since it is unbounded from above. Indeed, for any point $\widetilde{\mathbf{x}} \in \mathbb{R}^{nK}$ such that $\widetilde{\mathbf{x}}_i = \mathbf{1}_n$ (where $\mathbf{1}_n \in \mathbb{R}^n$ denotes the vector of all ones) and $\widetilde{\mathbf{x}}_l = \mathbf{0}_n$ for all $l \neq i$ it holds that $\mathcal{F}_{ij}(\alpha\widetilde{\mathbf{x}}) = \left(\alpha\sqrt{n} - \delta_{ij}\right)^2 \to \infty$ as $\alpha \to \infty$. Therefore, all local maximum points are necessarily non-global.

iii. The function $\mathcal{F}$ of Problem (1) (equivalently, Problem (3)) has no global maximum points since it is unbounded from above. Indeed, $\mathcal{F}$ is the sum of the non-negative functions $\mathcal{F}_{ij}$, for all $(i,j) \in \mathcal{E}$. Then, following the same arguments of item (ii) yields that all local maximum points of $\mathcal{F}$ are necessarily non-global.

Now, we are ready to state and prove the main result of this section. To this end, for any $\mathbf{x} \in \mathbb{R}^{nK}$ we denote by $\mathcal{E}_{\min}(\mathbf{x}) \subseteq \mathcal{E}$ the subset of all pairs $(i,j) \in \mathcal{E}$ for which $\mathbf{x}$ is a local minimum point of $\mathcal{F}_{ij}$. Meaning, if $(i,j) \in \mathcal{E}_{\min}(\mathbf{x})$ then $\mathbf{x}$ is a local minimum point of the function $\mathcal{F}_{ij}$. Similarly, we define the subset $\mathcal{E}_{\max}(\mathbf{x})$. In addition, for any $\mathbf{x} \in \mathbb{R}^{nK}$, we define the subset $\mathcal{E}_{\mathrm{nm}}(\mathbf{x}) \subseteq \mathcal{E}$ as the subset of all pairs $(i,j) \in \mathcal{E}$ for which $\mathbf{x}$ is not a local minimum nor a local maximum point of $\mathcal{F}_{ij}$. Meaning, if $(i,j) \in \mathcal{E}_{\mathrm{nm}}(\mathbf{x})$ then $\mathbf{x}$ is not an extremum point of the function $\mathcal{F}_{ij}$. Clearly, it holds that $\mathcal{E} = \mathcal{E}_{\min}(\mathbf{x}) \cup \mathcal{E}_{\max}(\mathbf{x}) \cup \mathcal{E}_{\mathrm{nm}}(\mathbf{x})$.

This union is also disjoint. To see this, if $(i,j) \in \mathcal{E}_{\min}(\mathbf{x})$, then $\mathbf{x}$ is a local minimum point of $\mathcal{F}_{ij}$. From Remark 3 (i) we know that $\mathbf{x}$ must be a global minimum point, and from Lemma 2 (i) we get that $\mathbf{x}$ must be a differentiable point of $\mathcal{F}_{ij}$ (since all stationary points are differentiable by their definition). In addition, if $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$ then $\mathbf{x}$ is a non-differentiable point of $\mathcal{F}_{ij}$ (see Lemma 2 (ii)). Therefore, the three subsets $\mathcal{E}_{\min}(\mathbf{x})$, $\mathcal{E}_{\max}(\mathbf{x})$ and $\mathcal{E}_{\mathrm{nm}}(\mathbf{x})$ are disjoint.

▶ **Theorem 4.** *Let $\mathbf{x} \in \mathbb{R}^{nK}$ be a non-differentiable point of $\mathcal{F}$. Let $\mathbf{d} \in \mathbb{R}^{nK}$ be such that $\mathbf{d}_i \neq \mathbf{d}_j$ for all $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$. Then, either $\mathbf{d}$ or $-\mathbf{d}$ is a descent direction of $\mathcal{F}$ at $\mathbf{x}$.*

Proof. Since $\mathbf{x}$ is a non-differentiable point of $\mathcal{F}$, it follows from Lemma 2 (iii) that $\mathcal{E}_{\max}(\mathbf{x}) \neq \emptyset$. From Lemma 1 (i) we know that for any $\mathbf{d} \in \mathbb{R}^{nK}$ such that $\mathbf{d}_i \neq \mathbf{d}_j$ for all $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$ it holds that $\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) < 0$ (and such $\mathbf{d}$ surely exists since the set $\mathcal{E}_{\max}(\mathbf{x})$ is finite). Meaning, such $\mathbf{d}$ is a descent direction of $\mathcal{F}_{ij}$ at $\mathbf{x}$ for all $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$.

If $\mathbf{d}$ is also a descent direction of $\mathcal{F}$ at $\mathbf{x}$ then we are done. Therefore, let us assume that $\mathbf{d}$ is a non-descent direction of $\mathcal{F}$ at $\mathbf{x}$ (that is, $\mathcal{F}'(\mathbf{x};\mathbf{d}) \geq 0$), and we will prove that $-\mathbf{d}$ is indeed a descent direction of $\mathcal{F}$ at $\mathbf{x}$.

Since $\mathcal{F}'(\mathbf{x};\mathbf{d}) = \sum_{(i,j)\in\mathcal{E}} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d})$, it holds that

$$0 \leq \mathcal{F}'(\mathbf{x};\mathbf{d}) = \sum_{(i,j)\in\mathcal{E}_{\max}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) + \sum_{(i,j)\in\mathcal{E}_{\min}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) + \sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}), \tag{7}$$

where we used the fact that the three sub-sets above are disjoint. From Lemma 2 (i) we know that $\nabla\mathcal{F}_{ij}(\mathbf{x}) = \mathbf{0}_{nK}$ for any $(i,j) \in \mathcal{E}_{\min}(\mathbf{x})$, and therefore

$$0 = \pm\nabla\mathcal{F}_{ij}(\mathbf{x})^T\mathbf{d} = \mathcal{F}'_{ij}(\mathbf{x};\pm\mathbf{d}), \quad \forall\,(i,j) \in \mathcal{E}_{\min}(\mathbf{x}). \tag{8}$$

Plugging (8) into (7) yields

$$0 \leq \mathcal{F}'(\mathbf{x};\mathbf{d}) = \sum_{(i,j)\in\mathcal{E}_{\max}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) + \sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}). \tag{9}$$

Now, since $\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) < 0$ for any $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$ (see Lemma 1 (i)), it follows from (9) that

$$0 < \sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}). \tag{10}$$

Recall that $\mathbf{x}$ is not an extremum point of $\mathcal{F}_{ij}$, for all $(i,j) \in \mathcal{E}_{\mathrm{nm}}(\mathbf{x})$. In particular, $\mathbf{x}$ is not a local maximum point of $\mathcal{F}_{ij}$, and from Lemma 2 (ii) we get that $\mathbf{x}$ must be a differentiable point of $\mathcal{F}_{ij}$. This means that, $\mathbf{x}_i \neq \mathbf{x}_j$ for all $(i,j) \in \mathcal{E}_{\mathrm{nm}}(\mathbf{x})$. From the continuity of each function $\mathcal{F}_{ij}$, there exists an open set $U \subseteq \mathbb{R}^{nK}$ that contains $\mathbf{x}$, such that $\mathbf{y}_i \neq \mathbf{y}_j$ for all $\mathbf{y} \in U$. This means that $\mathcal{F}_{ij}$ is continuously differentiable over U, and therefore $\mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) = \nabla\mathcal{F}_{ij}(\mathbf{x})^T\mathbf{d}$ for all $(i,j) \in \mathcal{E}_{\mathrm{nm}}(\mathbf{x})$. Thus,

$$\sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};-\mathbf{d}) = -\sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \nabla\mathcal{F}_{ij}(\mathbf{x})^T\mathbf{d} = -\sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x};\mathbf{d}) < 0, \tag{11}$$

where the last inequality follows from (10).

Last, recall that we picked $\mathbf{d}$ such that $-\mathbf{d}_i \neq -\mathbf{d}_j$ for any $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$. Then, from Lemma 1 (i) we get

$$0 > \sum_{(i,j)\in\mathcal{E}_{\max}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x}; -\mathbf{d}). \tag{12}$$

Summing (11) and (12), we derive from (8) that

$$0 > \sum_{(i,j)\in\mathcal{E}_{\max}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x}; -\mathbf{d}) + \sum_{(i,j)\in\mathcal{E}_{\min}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x}; -\mathbf{d}) + \sum_{(i,j)\in\mathcal{E}_{\mathrm{nm}}(\mathbf{x})} \mathcal{F}'_{ij}(\mathbf{x}; -\mathbf{d}) = \mathcal{F}'(\mathbf{x}; -\mathbf{d}),$$

which implies that $-\mathbf{d}$ is a descent direction of $\mathcal{F}$ at $\mathbf{x}$, as required. ◀

An immediate consequence of Theorem 4, is that any optimal solution of Problem (1) is necessarily a stationary point, i.e., a differentiable point with a gradient of all zeros. As mentioned in Section 1, this result is already known for DC programming problems [8], which as discussed above also applies to our case. However, the major motivation to develop our results is that Theorem 4 also gives an easy-to-find and *explicit* descent direction, that can be used to escape non-differentiable points, as we discuss next.

4 Escaping Non-Differentiable Saddle Points

As mentioned in Section 1, in this paper we aim at finding locally optimal solutions for Problem (1). Theorem 4 asserts that every non-differentiable point possesses an easy-to-find descent direction, hence such points cannot be optimal solutions for Problem (1). With this information in mind, one can evade any non-differentiable point encountered by an algorithm and reach a differentiable point with a lower function value by applying a simple backtracking procedure. This escape procedure, abbreviated as EP, is recorded in Procedure 1.

Procedure 1 Escape Procedure (EP)

1: **Initialization:** $\mathbf{x} \in \mathbb{R}^{nK}$ a non-differentiable point of $\mathcal{F}$ and $t > 0$.
2: Pick $\mathbf{d} \in \mathbb{R}^{nK}$ such that $\mathbf{d}_i \neq \mathbf{d}_j$ for all $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$.
3: *Double backtracking procedure:* **do in parallel**
→ **while** $\mathcal{F}(\mathbf{x}) \leq \mathcal{F}(\mathbf{x} + t\mathbf{d})$ or $\mathbf{x}_i + t\mathbf{d}_i = \mathbf{x}_j + t\mathbf{d}_j$ for some $(i,j) \in \mathcal{E}$ **then** set $t := t/2$.
→ **while** $\mathcal{F}(\mathbf{x}) \leq \mathcal{F}(\mathbf{x} - t\mathbf{d})$ or $\mathbf{x}_i - t\mathbf{d}_i = \mathbf{x}_j - t\mathbf{d}_j$ for some $(i,j) \in \mathcal{E}$ **then** set $t := t/2$.
4: Set the output as $\mathbf{z} = \mathbf{x} \pm t\mathbf{d}$ according to the first while loop that breaks.

The importance of the procedure EP comes from the fact that incorporating this procedure can prevent any optimization algorithm from being trapped in non-differentiable points, which are all non-optimal solutions, and by that may lead to the desired convergence to differentiable points. This phenomena is very important in the case of Problem (1), since the Hessian matrix of the function $\mathcal{F}$ is continuous around differentiable points, and therefore one can utilize its eigenvalues to deduce whether the differentiable point at hand is a (local) minimum point or not. This topic will be further discussed in Section 5.

Now, we are ready to prove that the procedure EP indeed leads to a differentiable point with a lower function value.

▶ **Proposition 5.** *Let $\mathbf{x} \in \mathbb{R}^{nK}$ be the non-differentiable input point of EP. Then, the output point $\mathbf{z} \in \mathbb{R}^{nK}$ is a differentiable point of $\mathcal{F}$ for which $\mathcal{F}(\mathbf{z}) < \mathcal{F}(\mathbf{x})$.*

Proof. In order to prove the result, we show that the while loop in Step 3 of the procedure EP terminates after a finite number of iterations.

Let $\mathbf{d} \in \mathbb{R}^{nK}$ be a direction picked according to Step 2 in EP. That is, $\mathbf{d}_i \neq \mathbf{d}_j$ for all $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$. Note that such $\mathbf{d}$ surely exists as the set $\mathcal{E}_{\max}(\mathbf{x})$ is finite by its definition. Now, since $\mathbf{x}$ is a non-differentiable point, it follows from Theorem 4 that either $\mathbf{d}$ or $-\mathbf{d}$ is a descent direction of $\mathcal{F}$ at $\mathbf{x}$. We assume without the loss of generality that $\mathbf{d}$ is a descent direction. Therefore, there exists $\bar{\epsilon} > 0$ such that $\mathcal{F}(\mathbf{x}) > \mathcal{F}(\mathbf{x} + t\mathbf{d})$ for all $t \in (0, \bar{\epsilon}]$. Notice that if $\mathbf{x}_i = \mathbf{x}_j$ then $(i,j) \in \mathcal{E}_{\max}(\mathbf{x})$, hence $\mathbf{d}_i \neq \mathbf{d}_j$ and therefore $\mathbf{x}_i + t\mathbf{d}_i \neq \mathbf{x}_j + t\mathbf{d}_j$ for all $t \in (0, \bar{\epsilon}]$. If $\mathbf{x}_i \neq \mathbf{x}_j$ then we can set $\mathbf{d}_i = \mathbf{d}_j$ and therefore $\mathbf{x}_i + t\mathbf{d}_i \neq \mathbf{x}_j + t\mathbf{d}_j$ for all $t \in (0, \bar{\epsilon}]$. This means that $\mathbf{x}_i + t\mathbf{d}_i \neq \mathbf{x}_j + t\mathbf{d}_j$ for all $(i,j) \in \mathcal{E}$. Therefore, $\mathbf{z} \equiv \mathbf{x} + t\mathbf{d}$ is a differentiable point with a lower function value than $\mathbf{x}$, as required. ◀

▶ Remark 6 (Escaping Approximate Non-Differentiable Points). In practice, when we run algorithms we might never converge to a point, since we use a finite number of iterations. Hence, we might stop at an approximate non-differentiable point. Mathematically, a point $\mathbf{y} \in \mathbb{R}^{nK}$ is considered an ε-non-differentiable point if $\|\mathbf{y}_i - \mathbf{y}_j\| \leq \varepsilon$ for some $\varepsilon > 0$ and some $(i, j) \in \mathcal{E}$.

Notice that for such point $\mathbf{y} \in \mathbb{R}^{nK}$ there are infinitely many non-differentiable points that lies in an ε-neighborhood of $\mathbf{y}$. Therefore, we may wish to escape from this "almost" non-differentiable point $\mathbf{y}$ by obtaining a differentiable point $\mathbf{z} \in \mathbb{R}^{nK}$ with a lower function value than some of these non-differentiable points. To this end, notice that we can easily construct a non-differentiable point $\mathbf{x} \in \mathbb{R}^{nK}$ in an ε-neighborhood of $\mathbf{y}$ by projecting $\mathbf{y}$ on the set of non-differentibale points that satisfy $\mathbf{x}_i = \mathbf{x}_j$. Specifically, the point $\mathbf{x}$ is the solution of the projection problem:

$$\operatorname*{argmin}_{\mathbf{x} \in \mathbb{R}^{nK}} \left\{ \|\mathbf{y} - \mathbf{x}\|^2 \colon \mathbf{x}_i = \mathbf{x}_j \right\}.$$

It is straightforward to show that a closed-form solution is:

$$\mathbf{x}_i = \mathbf{x}_j = \frac{\mathbf{y}_i + \mathbf{y}_j}{2} \quad \text{and} \quad \mathbf{x}_k = \mathbf{y}_k, \quad \forall\, k \neq i, j.$$

Notice that $\|\mathbf{y} - \mathbf{x}\| \leq \varepsilon$, so $\mathbf{x}$ indeed lies within an ε-neighborhood of $\mathbf{y}$. We can then apply the escape procedure (EP) to the point $\mathbf{x}$, to obtain a differentiable point $\mathbf{z} \in \mathbb{R}^{nK}$ with a function value smaller than that of $\mathbf{x}$.

Computational Considerations.

Now, we would like to discuss a computational aspect of the procedure EP. The main computational effort in the procedure is the evaluation of the function $\mathcal{F}$. This effort depends on the computational setting of the function $\mathcal{F}$. To simplify the discussion, we consider Problem (1) using the terminology the SNL problem (see Problem (3) and Section 2). In this case, evaluating the $\mathcal{F}$ requires gathering information from all sensors in the network. As a result, this procedure can only be implemented in centralized network architectures, that is, networks with a central computing unit responsible for data collection from all sensors. However, in certain practical situations the architecture of the network has no centralized computing unit, like in distributed architectures. Instead, each sensor (or cluster of some sensors) performs its own calculations using information available locally, and this information is collected from neighboring sensors (or clusters) through communication (for further information about centralized and distributed network architectures in the context of the SNL problem, we refer the reader to [19]). This limitation has inspired us to also design a distributed version of the procedure EP, named EP-D. Due to the similarity of the arguments, we develop and state it in Appendix A.

4.1 Convergence to Stationary Points

Equipped with the procedure EP, or its distributed version EP-D, that escapes non-differentiable points of the function $\mathcal{F}$, one can incorporate it into any minimization algorithm that converges to critical points of $\mathcal{F}$, to possibly obtain a stationary point of $\mathcal{F}$. Recall that any optimal solution of Problem (1) is necessarily a stationary point (see Theorem 4). Therefore, by using the procedure EP we easily transform algorithms which are guaranteed to converge to critical points, into algorithms that converge to stationary points. This can be done by executing the following process:

i. Run an algorithm $\mathcal{A}$ with some starting point to obtain a critical point $\mathbf{x} \in \mathbb{R}^{nK}$ of $\mathcal{F}$.
ii. Run EP (or EP-D) to obtain a differentible point $\mathbf{z} \in \mathbb{R}^{nK}$ with a lower function value.
iii. Repeat the process with $\mathbf{z} \in \mathbb{R}^{nK}$ as the starting point of $\mathcal{A}$.

Illustration of escaping a non-differentibale saddle point.

Here we give a simple one-dimensional numerical example that illustrates the convergence to a stationary point by utilizing the process described above. We show how a convergent algorithm escapes a non-differentiable saddle point and converges to a stationary point by using the procedure EP. We note that the description provided here is applicable to any dimension, but we choose to focus on the one-dimensional setting for clarity. This decision enables us to plot the function, facilitating a clear visualization of the saddle point, which is more challenging to illustrate in higher dimensions.

Using again the terminology of SNL, we consider a one-dimensional network with three sensors, where sensor #3 is an anchor, that is, has a known location. In this example, $x_3 = 2 \in \mathbb{R}$. The unknown locations of sensors

#1 and #2 are denoted by $x_1 \in \mathbb{R}$ and $x_2 \in \mathbb{R}$, respectively. In addition, we take $d_{12} = d_{13} = 1$. Under this setting, the objective function of Problem (3) takes the form

$$\mathcal{F}(x_1, x_2) = (|x_1 - x_2| - 1)^2 + (|x_1 - 2| - 1)^2.$$

Easy calculations show that the set of critical points of $\mathcal{F}$ is classified as follows:

- $(2, 2)$ is a non-differentiable local maximum point with value 2.
- $(1, 1)$, $(3, 3)$, $(2, 1)$ and $(2, 3)$ are non-differentiable saddle points with value 1.
- $(1, 0)$, $(1, 2)$, $(3, 2)$ and $(3, 4)$ are differentiable global minimum points with value 0. By definition, these points are stationary points.

We apply iterations of the classical Sub-Gradient (SG) method to minimize $\mathcal{F}$, and we will show that this method (given a specific starting point) converges to a non-differentiable saddle point. Then, we will invoke procedure EP to establish convergence of SG to a stationary point. This process is illustrated below in Figure 1, and is discussed now.

Notice that the directional derivative of $\mathcal{F}$ at any point satisfying $x_1 = x_2$ and $x_1 \neq 2$, at the direction $\mathbf{d} \in \{\mathbf{d} \in \mathbb{R}^2 : d_1 = d_2\}$ is given by

$$g(x_1) \equiv \mathcal{F}'((x_1, x_1); \mathbf{d}) = \lim_{\epsilon \to 0^+} \frac{\mathcal{F}(x_1 + \epsilon d_1, x_1 + \epsilon d_1) - \mathcal{F}(x_1, x_1)}{\epsilon} = 2d_1 \cdot \text{sign}(x_1 - 2) \cdot (|x_1 - 2| - 1).$$

Hence, for a starting point (x_1^0, x_2^0) satisfying $x_1^0 = x_2^0$ and $x_1 \neq 2$, SG update steps take the form

$$(x_1^{k+1}, x_2^{k+1}) = (x_1^k, x_2^k) - t^k \cdot (g(x_1^k), g(x_1^k)),$$

for any iteration $k \geq 0$ and for some step-sizes $t^k > 0$, assuming that $x_1^k \neq 2$ for all $k \geq 0$.

Initializing the above update steps with $(x_1^0, x_2^0) = (-1, -1)$, $t^k \equiv 1/2$ and $\mathbf{d} = (1, 1)$ for any $k \geq 0$, it can be proved that SG converges to the critical point $(1, 1)$, which is a non-differentiable saddle. We now invoke procedure EP, and obtain a random differentiable point $(\bar{x}_1, \bar{x}_2)$ satisfying $\mathcal{F}(1, 1) > \mathcal{F}(\bar{x}_1, \bar{x}_2)$. Restarting SG with $(\bar{x}_1, \bar{x}_2)$ for additional iterations, yields convergence to one of the stationary global minimum points.

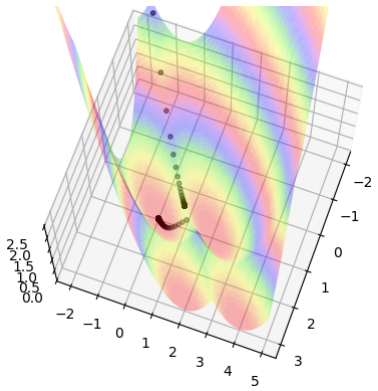

Figure 1 The trajectory of SG is depicted on the contour plots of the function $\mathcal{F}$, where the black points represent iterations. The method starts from the point $(-1, -1)$ and after 100 iterations, it converges to the non-differentiable saddle point $(1, 1)$. By invoking EP and performing an additional 100 iterations of SG, the method eventually converges to the stationary point $(1, 0)$.

Figure 2 illustrates the values of $\mathcal{F}$ plotted against the iterations $k \geq 0$ of SG. It is evident that the method converged to the value 1 after 100 iterations, which corresponds to the non-differentiable saddle point $(1, 1)$. Subsequently, after invoking EP and resuming SG for an additional 100 iterations, the function value decreased, ultimately converging to the optimal value of 0.

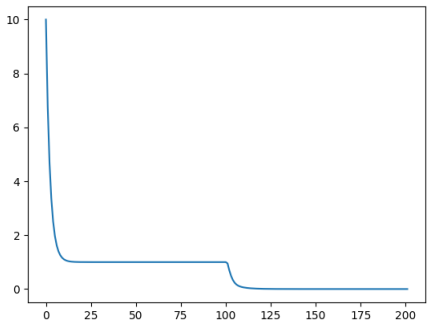

Figure 2 Values of $\mathcal{F}$ (y-axis) plotted against the iterations (x-axis) generated by SG with EP.

5 Classifying Stationary Points

In this section, our main goal is to identify the local minimum points of Problem (1). To achieve this goal, we utilized is previous sections the first-order optimality condition, which asserts that any local minimum point must be a critical point. As proved in Section 3, we established that any non-differentiable critical point is not a local minimum point. Consequently, our focus shifts to the differentiable critical points, namely stationary points. In Section 4, we introduced the procedure EP designed to navigate away from non-differentiable critical points. This procedure effectively enables us to concentrate solely on stationary points.

Now that we have narrowed our focus to stationary points, we can proceed to distinguish the local minimum points from the stationary points. It is worth noting that for the Hessian matrix of a function to be well-defined at a point, it must be twice-differentiable. Considering that the function $\mathcal{F}$ is smooth at its stationary points, we can leverage second-order information, particularly the eigenvalues of the Hessian matrix at these points, to further classify the stationary points.

We denote the Hessian matrix of $\mathcal{F}$ at a differentiable point $\mathbf{x} \in \mathbb{R}^{nK}$ as $\nabla^2\mathcal{F}(\mathbf{x}) \in \mathbb{R}^{nK\times nK}$. We recall that the necessary second-order optimality condition applied to a stationary point $\mathbf{x} \in \mathbb{R}^{nK}$, states that if $\lambda_{\min}\big(\nabla^2\mathcal{F}(\mathbf{x})\big) < 0$ (where $\lambda_{\min}$ denotes the minimal eigenvalue), then $\mathbf{x}$ is not a local minimum point [6] (in particular, it is not an optimal solution). Otherwise, if $\lambda_{\min}\big(\nabla^2\mathcal{F}(\mathbf{x})\big) \geq 0$, one cannot conclusively determine whether this point is a local minimum point using second-order information alone, and higher-order derivatives must be considered.

This implies that upon obtaining a stationary point, such as by employing our procedure EP, one can compute the minimal eigenvalue of its Hessian matrix (which is guaranteed to exist as this point as discussed above) to further classify this point. As noted earlier, it is critical to differentiate between centralized and distributed computational settings when addressing this task.

In a centralized setup, the minimal eigenvalue can be computed directly or approximated using methods such as inverse power iteration or Rayleigh Quotient techniques [17, 32]. However, these methods rely on global access to the Hessian matrix or its inverse, making them unsuitable for distributed architectures. In distributed settings, where each sensor contributes a portion of the Hessian matrix, direct computation of the minimal eigenvalue becomes infeasible due to the lack of global information and coordination. Moreover, distributed matrix computations are complex and require powerful systems to handle tasks such as message packaging and reception effectively [17].

While the literature offers various approaches for estimating the minimal eigenvalue in distributed systems [12, 20], these often rely on iterative algorithms. For example, Rayleigh Quotient methods can be applied in distributed systems by treating each summand of the Hessian matrix separately. However, these methods require an initial value close to the true minimal eigenvalue [36] – a significant challenge in practice, as the spectrum of the Hessian matrix of $\mathcal{F}$ is very dense. Furthermore, there is no guarantee that the minimal eigenvalue can be decomposed into a sum of local eigenvalues contributed by individual sensors. Another widely used approach, which we also incorporate here, involves leveraging the Eigenvalue Interlacing theorem and Weyl's theorem to approximate the minimal eigenvalue in a distributed manner.

In this paper, we address the challenge of computing the minimal eigenvalue in distributed settings. Instead of relying on iterative algorithms to approximate the minimal eigenvalue, we analyze the sum of local Hessian matrices and their principal submatrices using the well-known second-order optimality condition. This analysis allows us to derive a simple closed-form, fully distributed condition, providing lower and upper bounds on the

minimal eigenvalue based solely on local information. Our approach involves no computational complexity or communication costs, as it is based on direct analysis rather than approximation algorithms. While our method does not depend on iterative eigenvalue approximation, such methods can still be employed if desired for further analysis.

The goal of the rest of this section is to derive bounds on the minimal eigenvalue that can be calculated in centralized and distributed computational settings. These bounds will enable us to establish a stricter necessary condition (see Section 5.2) for a stationary point to qualify as a (local) minimum point in the distributed setting. Before developing this necessary condition, we calculate the eigenvalues of the Hessian matrix of the functions $\mathcal{F}_{ij}$, $(i,j) \in \mathcal{E}$, as defined in (4).

5.1 Eigenvalues of the Hessian of $\mathcal{F}_{ij}$

We denote by $\mathbf{e}_i \in \mathbb{R}^K$ the i-th unit vector, where K is the number of nodes. That is, $\mathbf{e}_i$ is the vector of all zeros, except for the i-th coordinate which is 1. For any $(i,j) \in \mathcal{E}$, $i < j$, we denote by $\mathbf{A}_{ij} \in \mathbb{R}^{n \times nK}$ the matrix

$$\mathbf{A}_{ij} \equiv (\mathbf{e}_i - \mathbf{e}_j)^T \otimes \mathbf{I}_n = \begin{bmatrix} \mathbf{0}_{n\times n(i-1)} & \mathbf{I}_n & \mathbf{0}_{n\times n(j-i)} & -\mathbf{I}_n & \mathbf{0}_{n\times n(K-j)} \end{bmatrix}, \tag{13}$$

where $\otimes$ denotes the Kronecker matrix product, $\mathbf{I}_n$ is the $n \times n$ identity matrix, and $\mathbf{0}_{p\times q}$ is the $p \times q$ zero matrix. Under this notation we can rewrite (recall (4))

$$\mathcal{F}_{ij}(\mathbf{x}) \equiv \left(\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij}\right)^2 = \left(\|\mathbf{A}_{ij}\mathbf{x}\| - \delta_{ij}\right)^2, \quad \forall\, (i,j) \in \mathcal{E}.$$

Now, recall that for any differentiable point $\mathbf{x} \in \mathbb{R}^{nK}$ of $\mathcal{F}_{ij}$ (i.e., a point satisfying $\mathbf{x}_i \neq \mathbf{x}_j$ and hence $\|\mathbf{A}_{ij}\mathbf{x}\| > 0$), the Hessian matrix of $\mathcal{F}_{ij}$ at $\mathbf{x}$ exists and is continuous. To compute the Hessian matrix $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$, we use the following technical lemma that its proof simply follows by applying the chain rule for multi-variate functions, and is therefore skipped.

▶ **Lemma 7.** *For any matrix $\mathbf{A} \in \mathbb{R}^{p\times q}$ and vector $\mathbf{x} \in \mathbb{R}^q$ satisfying $\|\mathbf{A}\mathbf{x}\| > 0$, it holds that*

i. $\nabla(\|\mathbf{A}\mathbf{x}\|) = \frac{\mathbf{A}^T\mathbf{A}\mathbf{x}}{\|\mathbf{A}\mathbf{x}\|}$.

ii. $\nabla^2(\|\mathbf{A}\mathbf{x}\|) = \frac{\mathbf{A}^T\mathbf{A} - \nabla(\|\mathbf{A}\mathbf{x}\|)\nabla(\|\mathbf{A}\mathbf{x}\|)^T}{\|\mathbf{A}\mathbf{x}\|}$.

Following Lemma 7, we immediately obtain explicit formulas for the gradient and Hessian matrix of the functions $\mathcal{F}_{ij}$. To this end, for any $(i,j) \in \mathcal{E}$, we define the scalars $\epsilon_{ij}(\mathbf{x}) \equiv \|\mathbf{A}_{ij}\mathbf{x}\| - \delta_{ij}$ and matrices

$$\mathbf{X}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \in \mathbb{R}^{n\times n}. \tag{14}$$

▶ **Corollary 8.** *Let $(i,j) \in \mathcal{E}$ and let $\mathbf{x} \in \mathbb{R}^{nK}$ such that $\mathbf{x}_i \neq \mathbf{x}_j$. Then,*

i. $\nabla\mathcal{F}_{ij}(\mathbf{x}) = \frac{2}{\|\mathbf{A}_{ij}\mathbf{x}\|}\epsilon_{ij}(\mathbf{x})\mathbf{A}_{ij}^T\mathbf{A}_{ij}\mathbf{x}$.

ii. $\nabla^2\mathcal{F}_{ij}(\mathbf{x}) = 2\mathbf{A}_{ij}^T\Big(\frac{\epsilon_{ij}(\mathbf{x})}{\|\mathbf{A}_{ij}\mathbf{x}\|}\mathbf{I}_n + \frac{\delta_{ij}}{\|\mathbf{A}_{ij}\mathbf{x}\|^3}\mathbf{X}_{ij}\Big)\mathbf{A}_{ij}$.

To write $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$ compactly, we define for any $(i,j) \in \mathcal{E}$ and $\mathbf{x} \in \mathbb{R}^{nK}$ satisfying $\mathbf{x}_i \neq \mathbf{x}_j$, the symmetric matrices $\mathbf{B}_{ij}(\mathbf{x}), \mathbf{C}_{ij}(\mathbf{x}), \mathbf{G}_{ij}(\mathbf{x}) \in \mathbb{R}^{n\times n}$ as (recall the definition of $\mathbf{X}_{ij}$ in (14))

$$\mathbf{B}_{ij}(\mathbf{x}) \equiv \frac{\epsilon_{ij}(\mathbf{x})}{\|\mathbf{A}_{ij}\mathbf{x}\|}\mathbf{I}_n, \quad \mathbf{C}_{ij}(\mathbf{x}) \equiv \frac{\delta_{ij}}{\|\mathbf{A}_{ij}\mathbf{x}\|^3}\mathbf{X}_{ij} \quad \text{and} \quad \mathbf{G}_{ij}(\mathbf{x}) \equiv \mathbf{B}_{ij}(\mathbf{x}) + \mathbf{C}_{ij}(\mathbf{x}), \tag{15}$$

and we get that

$$\nabla^2\mathcal{F}_{ij}(\mathbf{x}) = 2\mathbf{A}_{ij}^T\mathbf{G}_{ij}(\mathbf{x})\mathbf{A}_{ij}. \tag{16}$$

Using (5) it immediately follows that

$$\nabla^2\mathcal{F}(\mathbf{x}) = 2\sum_{(i,j)\in\mathcal{E}} \mathbf{A}_{ij}^T\mathbf{G}_{ij}(\mathbf{x})\mathbf{A}_{ij}. \tag{17}$$

Simple calculations show that (17) is given explicitly as

$$\nabla^2\mathcal{F}(\mathbf{x}) = 2\begin{bmatrix} \sum_{j\in\mathcal{E}_1}\mathbf{G}_{1j}(\mathbf{x}) & -\overline{\mathbf{G}}_{12}(\mathbf{x}) & \cdots & -\overline{\mathbf{G}}_{1K}(\mathbf{x}) \\ -\overline{\mathbf{G}}_{12}(\mathbf{x}) & \sum_{j\in\mathcal{E}_2}\mathbf{G}_{2j}(\mathbf{x}) & \cdots & -\overline{\mathbf{G}}_{2K}(\mathbf{x}) \\ \vdots & \vdots & \ddots & \vdots \\ -\overline{\mathbf{G}}_{1K}(\mathbf{x}) & -\overline{\mathbf{G}}_{2K}(\mathbf{x}) & \cdots & \sum_{j\in\mathcal{E}_K}\mathbf{G}_{Kj}(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^{nK\times nK}, \tag{18}$$

where we set

$$\overline{\mathbf{G}}_{ij}(\mathbf{x}) \equiv \begin{cases} \mathbf{G}_{ij}(\mathbf{x}), & (i,j) \in \mathcal{E}, \\ \mathbf{0}_{n\times n}, & (i,j) \notin \mathcal{E}. \end{cases}$$

Now, recall that we are interested in calculating a lower bound on the minimal eigenvalue of the $\nabla^2\mathcal{F}(\mathbf{x})$ in a distributed fashion. Such lower bound can be obtained by calculating the eigenvalues of principal sub-matrices of $\nabla^2\mathcal{F}(\mathbf{x})$ (see exact definition and statement below in Theorem 13). Hence, following (18), we first calculate the eigenvalues of the matrices $\mathbf{G}_{ij}(\mathbf{x})$, $(i,j) \in \mathcal{E}$. To this end, we begin with finding the eigenvalues of the matrix $\mathbf{C}_{ij}(\mathbf{x})$ as defined in (15).

▶ **Lemma 9.** *Let $(i,j) \in \mathcal{E}$ and $\mathbf{x} \in \mathbb{R}^{nK}$ satisfying $\mathbf{x}_i \neq \mathbf{x}_j$. Then,*

i. $\operatorname{rank}(\mathbf{X}_{ij}) = 1$ *and* $\lambda_{\max}(\mathbf{X}_{ij}) = \|\mathbf{x}_i - \mathbf{x}_j\|^2 = \|\mathbf{A}_{ij}\mathbf{x}\|^2$.

ii. *The eigenvalues of the matrix $\mathbf{C}_{ij}(\mathbf{x})$ are $\delta_{ij}/\|\mathbf{A}_{ij}\mathbf{x}\|$ with multiplicity 1, and 0 with multiplicity $n-1$.*

Proof.

i. Notice that for any two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ we have $\mathbf{u}\mathbf{u}^T\mathbf{v} = \big(\mathbf{u}^T\mathbf{v}\big)\mathbf{u}$, which means that the matrix $\mathbf{u}\mathbf{u}^T$ maps any vector $\mathbf{v}$ to a vector in the space $\operatorname{span}(\mathbf{u})$. If $\mathbf{u} \neq \mathbf{0}_n$, then $\operatorname{rank}\big(\mathbf{u}\mathbf{u}^T\big) = \dim\big(\operatorname{image}\big(\mathbf{u}\mathbf{u}^T\big)\big) = 1$. This means that all eigenvalues of $\mathbf{u}\mathbf{u}^T$ are 0 except one of them. Moreover, taking $\mathbf{v} = \mathbf{u}$ we see that $\mathbf{u}$ is an eigenvector of $\mathbf{u}\mathbf{u}^T$ corresponding to $\lambda_{\max}\big(\mathbf{u}\mathbf{u}^T\big) = \mathbf{u}^T\mathbf{u} = \|\mathbf{u}\|^2$. Hence, item (i) now follows by taking $\mathbf{u} = \mathbf{x}_i - \mathbf{x}_j$ and recalling that $\mathbf{X}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$.

ii. First, from item (i) it follows that $\operatorname{rank}(\mathbf{C}_{ij}(\mathbf{x})) = 1$, which implies that 0 is an eigenvalue of $\mathbf{C}_{ij}(\mathbf{x})$ with multiplicity $n-1$. Last, since the only non-zero eigenvalue of $\mathbf{X}_{ij}$ is $\|\mathbf{A}_{ij}\mathbf{x}\|^2$, it follows that $\delta_{ij}/\|\mathbf{A}_{ij}\mathbf{x}\|$ is an eigenvalue of $\mathbf{C}_{ij}(\mathbf{x})$ with multiplicity 1. ◀

Now, we are ready to explicitly find the eigenvalues of the matrix $\mathbf{G}_{ij}(\mathbf{x})$, a result that is formalized in the next lemma.

▶ **Lemma 10.** *Let $(i,j) \in \mathcal{E}$ and $\mathbf{x} \in \mathbb{R}^{nK}$ satisfying $\mathbf{x}_i \neq \mathbf{x}_j$. Then, the eigenvalues of $\mathbf{G}_{ij}(\mathbf{x})$ are $\lambda_{\max}(\mathbf{G}_{ij}(\mathbf{x})) = 1$ with multiplicity 1, and $\lambda_{\min}(\mathbf{G}_{ij}(\mathbf{x})) = \epsilon_{ij}(\mathbf{x})/\|\mathbf{A}_{ij}\mathbf{x}\|$ with multiplicity $n-1$.*

Proof. First, we denote by $\mathbf{D}_{\mathbf{C}_{ij}}(\mathbf{x})$ the diagonal matrix containing the eigenvalues of $\mathbf{C}_{ij}(\mathbf{x})$. Then, there exists an orthogonal matrix $\mathbf{U}$ such that

$$\mathbf{B}_{ij}(\mathbf{x}) + \mathbf{C}_{ij}(\mathbf{x}) = \mathbf{B}_{ij}(\mathbf{x}) + \mathbf{U}^T\mathbf{D}_{\mathbf{C}_{ij}}(\mathbf{x})\mathbf{U} = \mathbf{U}^T\big(\mathbf{B}_{ij}(\mathbf{x}) + \mathbf{D}_{\mathbf{C}_{ij}}(\mathbf{x})\big)\mathbf{U},$$

where we used the fact that $\mathbf{B}_{ij}(\mathbf{x})$ is a scalar multiplication of the identity matrix. Since the matrix $\mathbf{B}_{ij}(\mathbf{x}) + \mathbf{D}_{\mathbf{C}_{ij}}(\mathbf{x})$ is diagonal, it follows that the eigenvalues of $\mathbf{G}_{ij}(\mathbf{x}) = \mathbf{B}_{ij}(\mathbf{x}) + \mathbf{C}_{ij}(\mathbf{x})$ are exactly the sum of eigenvalues of $\mathbf{B}_{ij}(\mathbf{x})$ and $\mathbf{C}_{ij}(\mathbf{x})$. Since the eigenvalues of $\mathbf{B}_{ij}(\mathbf{x})$ are $\epsilon_{ij}(\mathbf{x})/\|\mathbf{A}_{ij}\mathbf{x}\|$ with multiplicity n, and since $(\epsilon_{ij}(\mathbf{x}) + \delta_{ij})/\|\mathbf{A}_{ij}\mathbf{x}\| = 1$, the required result now follows from Lemma 9(ii).

Finally, we have

$$\frac{\epsilon_{ij}(\mathbf{x})}{\|\mathbf{A}_{ij}\mathbf{x}\|} = 1 - \frac{\delta_{ij}}{\|\mathbf{A}_{ij}\mathbf{x}\|} < 1,$$

and therefore 1 is indeed the maximal eigenvalue of $\mathbf{G}_{ij}(\mathbf{x})$. ◀

We conclude this part with a full characterization of eigenvalues of the Hessian $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$.

▶ **Lemma 11.** *Let $(i,j) \in \mathcal{E}$ and $\mathbf{x} \in \mathbb{R}^{nK}$ such that $\mathbf{x}_i \neq \mathbf{x}_j$. Then, $\lambda \neq 0$ is an eigenvalue of $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$ if and only if $\lambda/4$ is an eigenvalue of $\mathbf{G}_{ij}(\mathbf{x})$.*

Proof. Let $\mathbf{0}_{nK} \neq \mathbf{y} \in \mathbb{R}^{nK}$ be an eigenvector of $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$ corresponding to the eigenvalue $\lambda \neq 0$. Hence, $\nabla^2\mathcal{F}_{ij}(\mathbf{x})\mathbf{y} = \lambda\mathbf{y}$, and by writing in an explicit form using (16) we get

$$\begin{cases} 2\mathbf{G}_{ij}(\mathbf{x})\mathbf{y}_i - 2\mathbf{G}_{ij}(\mathbf{x})\mathbf{y}_j = \lambda\mathbf{y}_i, \\ -2\mathbf{G}_{ij}(\mathbf{x})\mathbf{y}_i + 2\mathbf{G}_{ij}(\mathbf{x})\mathbf{y}_j = \lambda\mathbf{y}_j. \end{cases} \tag{19}$$

Summing (19) we get $\frac{\lambda}{2}(\mathbf{y}_i + \mathbf{y}_j) = 0$. Since $\lambda \neq 0$ then $\mathbf{y}_i = -\mathbf{y}_j$. If $\mathbf{y}_i = \mathbf{0}_n$ then it follows that $\mathbf{y} = \mathbf{0}_{nK}$ which is a contradiction. Plugging $\mathbf{y}_i = -\mathbf{y}_j$ in the first equation of (19) we get $\mathbf{G}_{ij}(\mathbf{x})\mathbf{y}_i = \frac{\lambda}{4}\mathbf{y}_i$, which implies that $\lambda/4$ is an eigenvalue of $\mathbf{G}_{ij}(\mathbf{x})$.

Conversely, assume that $\lambda/4$ is an eigenvalue of $\mathbf{G}_{ij}(\mathbf{x})$ with eigenvector $\mathbf{0}_n \neq \widetilde{\mathbf{z}} \in \mathbb{R}^n$. Let $\mathbf{0}_{nK} \neq \mathbf{z} \in \mathbb{R}^{nK}$ such that $\mathbf{z}_i = -\mathbf{z}_j = \widetilde{\mathbf{z}}$. Now, it immediately follows from the LHS of (19) that $\nabla^2\mathcal{F}_{ij}(\mathbf{x})\mathbf{z} = \lambda\mathbf{z}$, and therefore λ is an eigenvalue of $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$, as required. ◀

The following result is an immediate consequence.

▶ **Corollary 12.** *Let $(i,j) \in \mathcal{E}$ and $\mathbf{x} \in \mathbb{R}^{nK}$ such that $\mathbf{x}_i \neq \mathbf{x}_j$. Then,*

i. *The eigenvalues of $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$ are $\lambda_{\max}\big(\nabla^2\mathcal{F}_{ij}(\mathbf{x})\big) = 4$ with multiplicity 1, 0 with multiplicity $n(K-1)$, and $4(\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij})/\|\mathbf{x}_i - \mathbf{x}_j\|$ with multiplicity $n-1$.*

ii. *If $\|\mathbf{x}_i - \mathbf{x}_j\| \geq \delta_{ij}$ then $\nabla^2\mathcal{F}_{ij}(\mathbf{x})$ is positive semi-definite, and otherwise it is indefinite.*

5.2 Necessary Condition for a Locally Optimal Solution

We recall that given a stationary point $\mathbf{x} \in \mathbb{R}^{nK}$ of $\mathcal{F}$, if the Hessian $\nabla^2\mathcal{F}(\mathbf{x})$ has a negative eigenvalue, then $\mathbf{x}$ is *not* a minimum point. Therefore, in this sub-section, we find lower and upper bounds on the minimal eigenvalue of $\nabla^2\mathcal{F}(\mathbf{x})$ that can be calculated in a distributed fashion. This is accomplished using the Eigenvalue Interlacing theorem and Weyl's theorem (see, for example, [21]), which are also stated below. First, we recall that for any square matrix $\mathbf{A} \in \mathbb{R}^{q\times q}$, then a square matrix $\mathbf{B} \in \mathbb{R}^{p\times p}$ for some $p < q$ is called a *principal sub-matrix* of $\mathbf{A}$, if there exists an orthogonal matrix $\mathbf{P} \in \mathbb{R}^{q\times p}$ such that $\mathbf{P}^T\mathbf{A}\mathbf{P} = \mathbf{B}$. In addition, in this paper, we index the eigenvalues of a symmetric matrix $\mathbf{A} \in \mathbb{R}^{q\times q}$ in a non-decreasing order, i.e.,

$$\lambda_{\max}(\mathbf{A}) \equiv \lambda_q(\mathbf{A}) \geq \ldots \geq \lambda_2(\mathbf{A}) \geq \lambda_1(\mathbf{A}) \equiv \lambda_{\min}(\mathbf{A}).$$

▶ **Theorem 13** (Eigenvalue Interlacing theorem)**.** *Let $\mathbf{A} \in \mathbb{R}^{q\times q}$ be a symmetric matrix. Let $\mathbf{B} \in \mathbb{R}^{p\times p}$ for some $p < q$ be a principal sub-matrix of $\mathbf{A}$. Then, it holds that*

$$\lambda_s(\mathbf{A}) \leq \lambda_s(\mathbf{B}) \leq \lambda_{s+q-p}(\mathbf{A}), \quad \forall\, s = 1, 2, \ldots, p.$$

▶ **Theorem 14** (Weyl's theorem)**.** *Let $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{q\times q}$ be two symmetric matrices. Then, for any $p = 1, 2, \ldots, q$ it holds that*

$$\lambda_p(\mathbf{A}+\mathbf{B}) \leq \lambda_{p+s}(\mathbf{A}) + \lambda_{q-s}(\mathbf{B}), \quad \forall\, s = 0, 1, \ldots, q-p.$$

Now, we are ready to provide the main result of this section, which is an explicit necessary fully-distributed condition for a stationary point of $\mathcal{F}$ to be a minimum point.

▶ **Theorem 15.** *Let $\mathbf{x} \in \mathbb{R}^{nK}$ be a stationary point of $\mathcal{F}$. If $\mathbf{x}$ is a local minimum point of $\mathcal{F}$, then $|\mathcal{E}_i| \geq \delta_{ij}/\|\mathbf{x}_i - \mathbf{x}_j\|$ for any $(i,j) \in \mathcal{E}$.*

Proof. We will prove that if there exists some $(i,j) \in \mathcal{E}$ such that $|\mathcal{E}_i| - \delta_{ij}/\|\mathbf{x}_i - \mathbf{x}_j\| < 0$, then $\mathbf{x}$ is not a minimum point of $\mathcal{F}$. More precisely, we will show that $\lambda_{\min}\big(\nabla^2\mathcal{F}(\mathbf{x})\big) < 0$.

Since $\mathbf{x}$ is a stationary point of $\mathcal{F}$, then in particular $\mathcal{F}$ is smooth at $\mathbf{x}$ and $\nabla^2\mathcal{F}(\mathbf{x})$ exists. Plugging $\mathbf{A} = \nabla^2\mathcal{F}(\mathbf{x})$, $\mathbf{B} = 2\sum_{j\in\mathcal{E}_i}\mathbf{G}_{ij}(\mathbf{x})$, $q = nK$, $p = n$ and $s = 1$ in Theorem 13, we obtain

$$\lambda_{\min}\big(\nabla^2\mathcal{F}(\mathbf{x})\big) \leq 2\lambda_{\min}\left(\sum_{j\in\mathcal{E}_i}\mathbf{G}_{ij}(\mathbf{x})\right). \tag{20}$$

Now, plugging $\mathbf{A} = \mathbf{G}_{ij}(\mathbf{x})$, $\mathbf{B} = \sum_{l\in\mathcal{E}_i, l\neq j}\mathbf{G}_{il}(\mathbf{x})$, $q = n$, $p = 1$ and $s = 0$ in Theorem 14, we get

$$\lambda_{\min}\left(\sum_{l\in\mathcal{E}_i}\mathbf{G}_{il}(\mathbf{x})\right) \leq \lambda_{\min}(\mathbf{G}_{ij}(\mathbf{x})) + \lambda_{\max}\left(\sum_{l\in\mathcal{E}_i, l\neq j}\mathbf{G}_{il}(\mathbf{x})\right) \leq \lambda_{\min}(\mathbf{G}_{ij}(\mathbf{x})) + \sum_{l\in\mathcal{E}_i, l\neq j}\lambda_{\max}(\mathbf{G}_{il}(\mathbf{x})), \tag{21}$$

where the second inequality follows by applying Theorem 14 with $p = q = n$ and $s = 0$.

Now, from Lemma 10 we know that $\lambda_{\min}(\mathbf{G}_{ij}(\mathbf{x})) = (\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij})/\|\mathbf{x}_i - \mathbf{x}_j\|$ and that $\lambda_{\max}(\mathbf{G}_{il}(\mathbf{x})) = 1$ for any $(i,l) \in \mathcal{E}$. Therefore, by combining (20) and (21), we obtain that

$$\lambda_{\min}\big(\nabla^2\mathcal{F}(\mathbf{x})\big) \leq \frac{2(\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij})}{\|\mathbf{x}_i - \mathbf{x}_j\|} + 2(|\mathcal{E}_i| - 1) = 2\left(|\mathcal{E}_i| - \frac{\delta_{ij}}{\|\mathbf{x}_i - \mathbf{x}_j\|}\right) < 0,$$

and hence $\mathbf{x}$ is not a minimum point of $\mathcal{F}$ (specifically, it is a strict saddle point). ◀

▶ Remark 16. By inspecting the proof of Theorem 15, we see that it can be stated in the following equivalent way: given a stationary point $\mathbf{x} \in \mathbb{R}^{nK}$ of $\mathcal{F}$, if $\min_{(i,j)\in\mathcal{E}} \{|\mathcal{E}_i| - \delta_{ij}/\|\mathbf{x}_i - \mathbf{x}_j\|\} < 0$, then $\mathbf{x}$ is not a local minimum point of $\mathcal{F}$. Hence, this point is a strict (differentiable) saddle point, that can be escaped by applying a backtracking procedure in the direction of the eigenvector corresponding to the minimal eigenvalue.

Moreover, it is worth noting that the condition $|\mathcal{E}_i| < \delta_{ij}/\|\mathbf{x}_i - \mathbf{x}_j\|$ for some $(i,j) \in \mathcal{E}$ (indicating that $\mathbf{x}$ is not a local minimum) is anticipated to be more prevalent in networks with high measurement noise. In such networks, the measurements δ_{ij} tend to be larger, increasing the likelihood of encountering this condition.

▶ Remark 17. Theorem 15 can be generalized to any network that is divided into clusters. In such a configuration, the network is divided into clusters, each containing a central processor called a *clusterhead* (see [19] for more details). For a given cluster represented by the index set $\mathcal{C} \subseteq [K]$, we can construct a principal sub-matrix $\left[\nabla^2\mathcal{F}(\mathbf{x})\right]_{\mathcal{C}} \in \mathbb{R}^{n|\mathcal{C}|\times n|\mathcal{C}|}$ of $\nabla^2\mathcal{F}(\mathbf{x})$ by selecting the rows and columns corresponding to the indices in $\mathcal{C}$. Similar to the previous analysis, it follows that

$$\lambda_{\min}\big(\nabla^2\mathcal{F}(\mathbf{x})\big) \leq \lambda_{\min}\big(\left[\nabla^2\mathcal{F}(\mathbf{x})\right]_{\mathcal{C}}\big).$$

In the case of cluster architectures, each cluster's central processor collects data from the sensors within the cluster, enabling explicit distributed computation of $\lambda_{\min}\big(\left[\nabla^2\mathcal{F}(\mathbf{x})\right]_{\mathcal{C}}\big)$.

6 Conclusion

In this paper, we delved into the mathematical geometry of a popular class of non-linear, non-convex and non-smooth least squares problems motivated by two challenging applications: the Wireless Sensor Network Localization and Multi-Dimensional Scaling. Our study led to several key findings. Firstly, we analyzed the extremum points of this class of problems and proved that any non-differentiable critical point corresponds to a saddle point. Building upon this result, we devised a procedure to identify an easy-to-find and explicit descent direction, enabling efficient escape from non-differentiable saddles. Importantly, this procedure is applicable to both centralized and distributed computational settings. Furthermore, we leveraged our understanding of the stationary points by examining the eigenvalues of the corresponding Hessian matrix. Building on this second-order information, we established a distributed necessary condition for local optimality. This condition allows us to assess the quality of stationary points in a distributed fashion, even when direct eigenvalue computations are infeasible due to limited information exchange or large matrix sizes.

A Appendix

Before we present the distributed escape procedure, we need the following notations, that will enable us to treat each sensor separately in a distributed manner. For any $i \in \{1, 2, \ldots, K\}$, we define $\mathcal{E}_i$ as the set containing the indices of all neighbors of sensor i. That is, $j \in \mathcal{E}_i$ if and only if $(i,j) \in \mathcal{E}$ or $(j,i) \in \mathcal{E}$. Now, for any $i \in \{1, 2, \ldots, K\}$, we define the function $\mathcal{F}_i \colon \mathbb{R}^{nK} \to \mathbb{R}$ as

$$\mathcal{F}_i(\mathbf{x}) \equiv \sum_{j\in\mathcal{E}_i} \mathcal{F}_{ij}(\mathbf{x}) = \sum_{j\in\mathcal{E}_i} (\|\mathbf{x}_i - \mathbf{x}_j\| - \delta_{ij})^2, \tag{22}$$

where we set $\mathcal{F}_{ij} \equiv \mathcal{F}_{ji}$ if $(j,i) \in \mathcal{E}$.

In addition, in the context of (22), by $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) \colon \mathbb{R}^n \to \mathbb{R}$ for some $\mathbf{z} \in \mathbb{R}^{nK}$ we denote the partial function $\mathbf{x}_i \mapsto \mathcal{F}_i(\mathbf{z})$. That is, $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z})$ treats $\mathbf{x}_i \in \mathbb{R}^n$ as the variable, while all $\mathbf{z}_j \in \mathbb{R}^n$, $j \neq i$, are fixed. Notice that for any $i \in \{1, 2, \ldots, K\}$, evaluating the function $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z})$ at sensor i, only requires the collection of vectors $\mathbf{z}_j \in \mathbb{R}^n$, $j \in \mathcal{E}_i$, from the neighbors of i. Hence, for each sensor i, evaluating $\mathcal{F}_i(\mathbf{x})$ only requires information that is locally available at sensor i.

In Procedure 2 below, we present the distributed version of EP, designed to escape non-differentiable points in distributed computational settings. It is important to note that EP-D operates using locally available information at each node, making it a distributed procedure.

The input point $\mathbf{x} \in \mathbb{R}^{nK}$ of EP-D can be any point, not necessarily a non-differentiable point. It is important to note that since determining the non-differentiability of a point requires full network information, which is not available in distributed architectures. However, EP-D guarantees that the output point is a differentiable point of the function $\mathcal{F}$ and has a lower function value than the input point (see Proposition 19).

Procedure 2 Escape Procedure – Distributed (EP-D)

Initialization: $\mathbf{x} \in \mathbb{R}^{nK}$ and set $\mathbf{z} = \mathbf{x}$.
for $i = 1, 2, \ldots, K$ **do**
  if $\mathbf{x}_i = \mathbf{z}_j$ for some $j \in \mathcal{E}_i$ **then** set $t = 1$ and pick $\mathbf{0}_n \neq \mathbf{d} \in \mathbb{R}^n$.
    Double backtracking procedure: **do in parallel**
      $\rightarrow$ **while** $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) \leq \mathcal{F}_i(\mathbf{x}_i + t\mathbf{d}; \mathbf{z})$ or $\mathbf{x}_i + t\mathbf{d} = \mathbf{z}_j$ for some $j \in \mathcal{E}_i$ **then** set $t \coloneqq t/2$.
      $\rightarrow$ **while** $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) \leq \mathcal{F}_i(\mathbf{x}_i - t\mathbf{d}; \mathbf{z})$ or $\mathbf{x}_i - t\mathbf{d} = \mathbf{z}_j$ for some $j \in \mathcal{E}_i$ **then** set $t \coloneqq t/2$.
    Update $\mathbf{z}_i \coloneqq \mathbf{x}_i \pm t\mathbf{d}$ according to the first while loop that breaks.
  end if
end for
Return $\mathbf{z}$ as the output.

To prove the above assertions about EP-D, it is required to derive a distributed variant of Theorem 4. To establish this, we prove a variant of Theorem 4 that considers only the sub-network consisting of sensor i and its neighbors.

▶ **Lemma 18.** *Let $i \in \{1, 2, \ldots, K\}$ and let some $\mathbf{z} \in \mathbb{R}^{nK}$. Assume that $\mathbf{x}_i \in \mathbb{R}^n$ is a non-differentiable point of the function $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z})$. Then, for any $\mathbf{0}_n \neq \mathbf{d} \in \mathbb{R}^n$ such that $\mathbf{d} \neq \mathbf{z}_j$ for all $j \in \mathcal{E}_i$, either $\mathbf{d}$ or $-\mathbf{d}$ is a descent direction of $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z})$ at $\mathbf{x}_i$.*

Proof. Denote by $\widetilde{\mathbf{x}} \in \mathbb{R}^{nK}$ the point $\widetilde{\mathbf{x}} \equiv (\mathbf{z}_1, \ldots, \mathbf{z}_{i-1}, \mathbf{x}_i, \mathbf{z}_{i+1}, \ldots, \mathbf{z}_K)$. Since $\mathbf{x}_i \in \mathbb{R}^n$ is a non-differentiable point of $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z})$, then there exist some $j \in \mathcal{E}_i$ such that $\widetilde{\mathbf{x}}_i = \widetilde{\mathbf{x}}_j$, and hence the function $\mathcal{F}_i$ (see (22)), is also non-differentiable at $\widetilde{\mathbf{x}}$. Now, we denote by $\widetilde{\mathbf{d}} \in \mathbb{R}^{nK}$ the vector satisfying $\widetilde{\mathbf{d}}_i = \mathbf{d} \neq \mathbf{0}_n$ and $\widetilde{\mathbf{d}}_j = \mathbf{0}_n$ for any $j \neq i$, and in particular $\widetilde{\mathbf{d}}_i \neq \widetilde{\mathbf{d}}_j$ for any $(i, j) \in \mathcal{E}$.

Notice that Theorem 4, which holds true for *any* network, considers the function $\mathcal{F}$ of Problem (3), which in turn is the sum of all functions $\mathcal{F}_{ij}$, for all $(i, j) \in \mathcal{E}$. Therefore, by taking the (sub)network that is composed of the sensor i and all it neighboring sensors, it immediately follows from Theorem 4 that either $\widetilde{\mathbf{d}}$ or $-\widetilde{\mathbf{d}}$ is a descent direction of the function $\mathcal{F}_i$ (as defined in (22)), at the point $\widetilde{\mathbf{x}}$. We assume without the loss of generality that $\widetilde{\mathbf{d}}$ is a descent direction. Hence, there exists $t > 0$ such that

$$\mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) = \mathcal{F}_i(\widetilde{\mathbf{x}}) > \mathcal{F}_i\left(\widetilde{\mathbf{x}} + t\widetilde{\mathbf{d}}\right) = \mathcal{F}_i(\mathbf{x}_i + t\mathbf{d}; \mathbf{z}),$$

and we obtain that $\mathbf{d}$ is a descent direction of $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z})$ at $\mathbf{x}_i$, as required. ◀

Next, we prove that EP-D indeed yields a differentiable point of the function $\mathcal{F}$ with a lower function value.

▶ **Proposition 19.** *Let $\mathbf{x} \in \mathbb{R}^{nK}$ and $\mathbf{z} \in \mathbb{R}^{nK}$ be the input and output points, respectively, of EP-D. Then, $\mathbf{z}$ is a differentiable point of $\mathcal{F}$ for which $\mathcal{F}(\mathbf{z}) < \mathcal{F}(\mathbf{x})$.*

Proof. Initially we set $\mathbf{z} = \mathbf{x}$ (see Step 1 in EP-D). We focus on the case in which $\mathbf{x}$ is a non-differentiable point of $\mathcal{F}$. In particular, there exist i and some $j \in \mathcal{E}_i$ such that $\mathbf{x}_i = \mathbf{z}_j$. For any $\mathbf{d} \neq \mathbf{0}_n$, it follows from Lemma 18 that there exists $\bar{\epsilon}_i > 0$ such that either $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) > \mathcal{F}_i(\mathbf{x}_i + t\mathbf{d}; \mathbf{z})$ or $\mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) > \mathcal{F}_i(\mathbf{x}_i - t\mathbf{d}; \mathbf{z})$ for all $t \in (0, \bar{\epsilon}_i]$. For the sake of simplicity, we assume without the loss of generality that $\mathbf{d}$ is a descent direction.

We now derive that (recall that initially $\mathbf{z} = \mathbf{x}$)

$$\mathcal{F}(\mathbf{x}) = \mathcal{F}_i(\mathbf{x}_i; \mathbf{z}) + \sum_{\substack{(k,j) \in \mathcal{E} \\ k,j \neq i}} \mathcal{F}_{kj}(\mathbf{z}) > \mathcal{F}_i(\mathbf{x}_i + t\mathbf{d}; \mathbf{z}) + \sum_{\substack{(k,j) \in \mathcal{E} \\ k,j \neq i}} \mathcal{F}_{kj}(\mathbf{z}) = \mathcal{F}(\mathbf{z}),$$

where the last eqaulity follows from the fact that we set $\mathbf{z}_i \coloneqq \mathbf{x}_i + t\mathbf{d}$ (see Step 5 in EP-D). Since the above process holds true for any $\mathbf{x}$, then indeed the output point of EP-D has a lower function value, as required.

Last, since the set $\mathcal{E}$ is finite, one can pick $t \in (0, \bar{\epsilon}_i]$ such that $\mathbf{x}_i + t\mathbf{d} \neq \mathbf{z}_j$ for all $j \in \mathcal{E}_i$ (see Step 4 in EP-D). Therefore, the output of EP-D is a differentiable point of $\mathcal{F}$. ◀

Acknowledgments

We would like to thank Tehila Dayan and Elad Buchris from the Faculty of Data and Decision Sciences at Technion – Israel Institute of Technology for their help in formulating and testing the algorithms in this paper.

References

1 El Mehdi Achour, François Malgouyres, and Sébastien Gerchinovitz. Global minimizers, strict and non-strict saddle points, and implicit regularization for deep linear neural networks. `https://hal.science/hal-03299887`, 2021.

2 El Mehdi Achour, François Malgouyres, and Sébastien Gerchinovitz. The loss landscape of deep linear neural networks: a second-order analysis. *J. Mach. Learn. Res.*, 25(242):1–76, 2024.

3 H. M. Ammari. *The Art of Wireless Sensor Networks. Volume 1: Fundamentals*. Springer, 2014.

4 Hedy Attouch and Jérôme Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Math. Program.*, 116(1-2):5–16, 2009.

5 Hedy Attouch, Jérôme Bolte, and Benar Fux 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):91–129, 2013.

6 Amir Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*, volume 19 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics, 2014.

7 Amir Beck. *First-Order Methods in Optimization*, volume 25 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics, 2017.

8 Amir Beck and Nadav Hallak. On the convergence to stationary points of deterministic and randomized feasible descent directions methods. *SIAM J. Optim.*, 30(1):56–79, 2020.

9 Amir Beck, Petre Stoica, and Jian Li. Exact and approximate solutions of source localization problems. *IEEE Trans. Signal Process.*, 56(5):1770–1778, 2008.

10 Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Math. Program.*, 146(1-2):459–494, 2014.

11 Frank E. Curtis and Daniel P. Robinson. Exploiting negative curvature in deterministic and stochastic optimization. *Math. Program.*, 176:69–94, 2019.

12 Davor Davidović. An overview of dense eigenvalue solvers for distributed memory systems. In *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, pages 265–271. IEEE, 2021.

13 Jan De Leeuw. Applications of Convex Analysis to Multidimensional Scaling. In *Recent Developments in Statistics*, pages 133–145. North-Holland, 1977.

14 Jan De Leeuw and Patrick Mair. Multidimensional Scaling using majorization: SMACOF in R. *J. Stat. Softw.*, 31(3):1–30, 2009.

15 Albert Fannjiang and Thomas Strohmer. The numerics of phase retrieval. *Acta Numer.*, 29:125–228, 2020.

16 Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pages 797–842. PMLR, 2015.

17 Gene H. Golub and Charles F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, 2013.

18 Eyal Gur, Alon Amar, and Shoham Sabach. Direct, fast and convergent solvers for the non-convex and non-smooth TDoA localization problem. *Digital Signal Process.*, 139: article no. 104074, 2023.

19 Eyal Gur, Shoham Sabach, and Shimrit Shtern. Alternating minimization based first-order method for the wireless sensor network localization problem. *IEEE Trans. Signal Process.*, 68:6418–6431, 2020.

20 Azwirman Gusrialdi and Zhihua Qu. Distributed estimation of all the eigenvalues and eigenvectors of matrices associated with strongly connected digraphs. *IEEE Control Sys. Lett.*, 1(2):328–333, 2017.

21 Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2012.

22 Chi Jin, Rong Ge, Praneeth Netrapalli, and Michael I. Kakade, Sham M. and. Jordan. How to escape saddle points efficiently. In *International conference on machine learning*, pages 1724–1732. PMLR, 2017.

23 Ronald Katende and Henry Kasumba. Efficient Saddle Point Evasion and Local Minima Escape in High-Dimensional Non-Convex Optimization. `https://arxiv.org/abs/2409.12604`, 2024.

24 Hoai An Le Thi and Tao Pham Dinh. Dc programming and DCA: thirty years of developments. *Math. Program.*, 169(1):5–68, 2018.

25 Jason D. Lee, Ioannis Panageas, Georgios Piliouras, Max Simchowitz, Michael I. Jordan, and Benjamin Recht. First-order methods almost always avoid strict saddle points. *Math. Program.*, 176:311–337, 2019.

26 D. Russell Luke, Shoham Sabach, Marc Teboulle, and Kobi Zatlawey. A simple globally convergent algorithm for the nonsmooth nonconvex single source localization problem. *J. Glob. Optim.*, 69(4):889–909, 2017.

27 Boris S. Mordukhovich. *Variational Analysis and Generalized Differentiation I: Basic Theory*, volume 330 of *Grundlehren der Mathematischen Wissenschaften*. Springer, 2006.

28 Jennifer L. Mueller and Samuli Siltanen. *Linear and nonlinear inverse problems with practical applications*. Society for Industrial and Applied Mathematics, 2012.

29 Jong-Shi Pang, Meisam Razaviyayn, and Alberth Alvarado. Computing B-stationary points of nonsmooth DC programs. *Math. Oper. Res.*, 42(1):95–118, 2017.

30 Tao Pham Dinh, Van Ngai Huynh, Hoai An Le Thi, and Vinh Thanh Ho. Alternating DC algorithm for partial DC

programming problems. *J. Glob. Optim.*, 82(4):897–928, 2022.

31 Nicola Piovesan and Tomaso Erseghe. Cooperative localization in WSNs: A hybrid convex/nonconvex solution. *IEEE Trans. Signal Inf. Process. Networks*, 4(1):162–172, 2018.

32 Yousef Saad. *Numerical methods for large eigenvalue problems: revised edition.* Society for Industrial and Applied Mathematics, 2011.

33 Qingjiang Shi, Chen He, Hongyang Chen, and Lingge Jiang. Distributed wireless sensor network localization via sequential greedy optimization algorithm. *IEEE Trans. Signal Process.*, 58(6):3328–3340, 2010.

34 Pham Dinh Tao et al. Algorithms for solving a class of nonconvex optimization problems. Methods of subgradients. In *Fermat days 85: Mathematics for optimization*, volume 129 of *North-Holland Mathematics Studies*, pages 249–271. North-Holland, 1986.

35 Pham Dinh Tao and El Bernoussi Souad. Duality in DC (difference of convex functions) optimization. Subgradient methods. In *Trends in Mathematical Optimization: 4th French-German Conference on Optimization*, volume 84 of *International Series of Numerical Mathematics/Internationale*, pages 277–293. Springer, 1988.

36 Richard A. Tapia, John E. Dennis, Jr, and Jan P. Schafermeyer. Inverse, shifted inverse, and Rayleigh quotient iteration as Newton's method. *SIAM Rev.*, 60(1):3–55, 2018.

37 L. Taylor. The phase retrieval problem. *IEEE Trans. Antennas Propag.*, 29(2):386–391, 1981.

38 Jianzhong Wang. *Geometric structure of high-dimensional data and dimensionality reduction.* Springer, 2012.

39 Changzhi Wu, Chaojie Li, and Qiang Long. A DC programming approach for sensor network localization with uncertainties in anchor positions. *J. Ind. Manag. Optim.*, 10(3):817–826, 2014.

40 Zhengqing Wu, Berfin Simsek, and Francois Ged. The Loss Landscape of Shallow ReLU-like Neural Networks: Stationary Points, Saddle Escaping, and Network Embedding. `https://arxiv.org/abs/2402.05626`, 2024.

41 Zhihui Zhu, Daniel Soudry, Yonina C. Eldar, and Michael B. Wakin. The global optimization geometry of shallow linear neural networks. *J. Math. Imaging Vis.*, 62(3):279–292, 2019.