Moreau Envelope ADMM for Decentralized Weakly Convex Optimization

Reza Mirzaeifard[†], Naveen K. D. Venkategowda[§], Alexander Jung[⋆], Stefan Werner[†]
[†]Department of Electronic Systems, Norwegian University of Science and Technology, Norway
[§]Department of Science and Technology, Linköping University, Sweden
[⋆]Department of Computer Science, Aalto University, Finland
E-mails: {reza.mirzaeifard, stefan.werner}@ntnu.no, naveen.venkategowda@liu.se, alex.jung@aalto.fi

***Abstract*—This paper proposes a proximal variant of the alternating direction method of multipliers (ADMM) for distributed optimization. Although the current versions of ADMM algorithm provide promising numerical results in producing solutions that are close to optimal for many convex and non-convex optimization problems, it remains unclear if they can converge to a stationary point for weakly convex and locally non-smooth functions. Through our analysis using the Moreau envelope function, we demonstrate that MADM can indeed converge to a stationary point under mild conditions. Our analysis also includes computing the bounds on the amount of change in the dual variable update step by relating the gradient of the Moreau envelope function to the proximal function. Furthermore, the results of our numerical experiments indicate that our method is faster and more robust than widely-used approaches.**

***Index Terms*—Distributed optimization, non-convex and non-smooth optimization, weakly convex functions, ADMM, Moreau envelope.**

I. Introduction

Many systems, like the internet-of-things (IoT) and cyber-physical systems, comprise distributed devices and sensors that gather data for inference and decision-making. Building distributed models in such systems without data transfer to a central hub calls for distributed optimization methods involving peer-to-peer interactions. In addition, these methods allow for coping with resource constraints, e.g., computational resources, battery power, communication bandwidth, and privacy protection [1]–[3].

There is a large body of work on distributed optimization methods from different perspectives. The most direct approach to the design of distribution optimization methods is via message-passing implementations of subgradient computation within subgradient descent methods [4]–[6]. Gradient methods are generalized to solve structured optimization problems using proximal methods [7], [8]. Additionally, variational methods for probabilistic models lend themselves naturally to optimization algorithms, such as variants of belief propagation [9]. The subgradient method is well-known for its ease of implementation, wherein a subgradient is taken at each step, followed by an average with neighbors. On the downside, subgradient descent has a sublinear convergence rate and requires tuning of the step size. Meanwhile, ADMM performs fast and accurately in many practical convex and non-convex optimization problems. For convex objective functions, subgradient methods and ADMM are guaranteed to converge to a global optimum under suitable parameter choices [6], [10], [11]. However, the analysis of these methods for non-convex problems is challenging due to the potential ill-behavior of the objective function. The convergence analysis for ADMM in non-convex problems is particularly more challenging as it requires analyzing the convergence of multiple sub-problems with different structures and assumptions.

One important family of non-convex optimization problems is weakly convex problems. Several such problems arise in machine learning, including robust phase retrieval [12], blind deconvolution [13], biconvex compressive sensing [14], and dictionary learning [15]. Smooth functions, or functions with Lipschitz continuous gradients, are weakly convex functions. Several non-convex optimization algorithms are proposed based on the smoothness assumption (e.g., [16], [17]); however, weakly convex functions are not restricted to smooth functions, and non-smooth functions can also be weakly convex [18]. Existing work on distributed optimization of weakly convex functions includes [19]. However, in the subgradient-based algorithm in [19], the local functions must satisfy the sharpness assumption, and the accuracy of the estimation depends on the step size. Although ADMM is a powerful algorithm applicable to many problems, it is not currently used to solve weakly convex problems. As seen in [20]–[23], several ADMM-based works study non-convex optimization; however, these works require a smooth objective function. It is still necessary to provide a distributed ADMM-based algorithm that could work in the weakly convex setting without having any Lipschitz differentiability condition.

We propose a Moreau envelope-based ADMM (MADM), suitable for distributed optimization where local objectives are weakly convex and not necessarily smooth. We chose the Moreau envelope-based ADMM approach because it allows us to guarantee a decrease in each primal update step and bound the amount of change in the dual update step by primary variables. This is achieved by incorporating the relationship between the Moreau envelope function and proximal function. Therefore, by selecting appropriate penalty parameters under mild conditions, including weakly convexity of each

This work was supported by the Research Council of Norway.

local function, a connected network, and the boundness of augmented Lagrangian function, we can ensure that the algorithm converges to a stationary point. The Moreau envelope-based ADMM approach stands out from other penalty-based ADMM algorithms due to its superior theoretical convergence properties. We conduct illustrative numerical experiments to verify the convergence properties of the proposed method. The experiments demonstrate the robustness of the proposed algorithm when we fix the penalty parameters and step size, and the problem structure remains the same. Unlike subgradient-based methods, the MADM approach ensures faster and more reliable convergence in this setting.

Mathematical Notations: Scalars, column vectors, and matrices are respectively denoted by lowercase, bold lowercase, and bold uppercase letters. The operator $(\cdot)^{\mathrm{T}}$ denotes transpose of a matrix, and the jth column of matrix $\mathbf{A}$ is denoted by $\mathbf{A}_j$. The set $\{1, \cdots, L\}$ is denoted by $[L]$. For a function $h : \mathbb{R}^n \to \mathbb{R}$ and penalty parameter $\gamma > 0$, $\mathcal{M}_h(\mathbf{w};\gamma) = \min_{\mathbf{x}} \left\{h(\mathbf{x}) + \frac{1}{2\gamma}\|\mathbf{x}-\mathbf{w}\|_2^2\right\}$ is the Moreau envelope function [24], and $\mathrm{Prox}_h(\mathbf{w};\gamma) = \arg\min_{\mathbf{x}} \left\{h(\mathbf{x}) + \frac{1}{2\gamma}\|\mathbf{x}-\mathbf{w}\|^2\right\}$ is its associated proximal operator.

II. Problem formulation

Suppose L agents solve the following problem:

$$\min_{\mathbf{x}} \sum_{i=1}^{L} f_i(\mathbf{x}), \tag{1}$$

where $f_i(\cdot) : \mathbb{R}^N \to \mathbb{R}$ represents the local objective function that is only known to agent i. Additionally, each agent may exchange information with its neighbors through the underlying undirected communication network, which can be modeled as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = [L]$ represents the set of agents and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ represents the set of edges. In other words, the existence of $e_{i,j} \in \mathcal{E}$ indicates that i and j can exchange information. Due to the fact that both $e_{i,j}$ and $e_{j,i}$ denote the same edge, we merely use the expressions $e_{i,j}$ (if $i < j$) or $e_{j,i}$ (if $j < i$) to avoid repetition. Additionally, $E = |\mathcal{E}|$ is the total number of edges, and $|\mathcal{N}_i|$ is the number of neighbors for node i in which $\mathcal{N}_i$ is its set of neighbors. In order to apply ADMM, one can rewrite (1) in the form of an edge consensus problem as follows:

$$\begin{aligned} \min_{\{\mathbf{x}_1,\cdots,\mathbf{x}_L,\mathbf{Z}\}} \quad & \sum_{i=1}^{L} f_i(\mathbf{x}_i) + g(\mathbf{Z}) \\ \text{subject to} \quad & \mathbf{x}_i = \mathbf{z}_{i,j}, \mathbf{x}_j = \mathbf{z}_{i,j}, \quad \forall e_{i,j} \in \mathcal{E} \end{aligned} \tag{2}$$

where each $\mathbf{Z} = \{\{\mathbf{z}_{i,j}\}_{j\in\mathcal{N}_i, j>i}\}_{i=1}^{L}$ are auxiliary variables, and $g(\cdot) = 0$. The augmented Lagrangian of (2) is:

$$\begin{aligned} \mathcal{L}_{\rho_\lambda}(\mathbf{X},\mathbf{Z},\boldsymbol{\lambda}) = \sum_{i=1}^{L} f_i(\mathbf{x}_i) + \sum_{e_{i,j}\in\mathcal{E}} \Big(& (\boldsymbol{\lambda}_{i,j}^{i})^{\mathrm{T}}(\mathbf{x}_i - \mathbf{z}_{i,j}) \\ + (\boldsymbol{\lambda}_{i,j}^{j})^{\mathrm{T}}(\mathbf{x}_j - \mathbf{z}_{i,j}) + \frac{\rho_\lambda}{2}\|\mathbf{x}_j - \mathbf{z}_{i,j}\|^2 + \frac{\rho_\lambda}{2}\|\mathbf{x}_i - \mathbf{z}_{i,j}\|^2 \Big), \end{aligned} \tag{3}$$

where $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_L]$, $\boldsymbol{\lambda} = \{\{\boldsymbol{\lambda}_{i,j}^{i}, \boldsymbol{\lambda}_{i,j}^{j}\}_{j\in\mathcal{N}_i, j>i}\}_{i=1}^{L}$ are dual variables, and ρ_λ is a penalty parameter. Distributed ADMMs are iterative procedures that involve three steps at each iteration. The first step is to minimize $\mathcal{L}_{\rho_\lambda}$ with respect to $\mathbf{X}$. Afterward, $\mathcal{L}_{\rho_\lambda}$ is minimized based on $\mathbf{Z}$. In the last step, a dual gradient-ascent iteration is used to update $\boldsymbol{\lambda}$.

Definition 1. *A function $f(x)$ is $\rho-$weakly convex ($\rho > 0$) if there exists a convex function $h(x)$ such that $h(x) = f(x) + \rho\|x\|^2$.*

Weakly convex local functions pose a challenge to distributed ADMM convergence because it can both be non-convex and non-smooth. In the absence of Lipschitz differentiability, which is coming from smoothness, and convexity of the objective function, existing distributed ADMM-based approaches cannot guarantee convergence. In the following section, we present an ADMM-based algorithm can deal with weakly convex functions, regardless of whether they are smooth.

III. Moreau envelope ADMM

The proximal augmented Lagrangian of (2) can be derived as:

$$\Psi_{\rho_\lambda,\rho_\beta}(\mathbf{X},\mathbf{Z},\boldsymbol{\beta},\boldsymbol{\lambda}) = \mathcal{L}_{\rho_\lambda}(\mathbf{X},\mathbf{Z},\boldsymbol{\lambda}) + \frac{\rho_\beta}{2}\|\mathbf{Z}-\boldsymbol{\beta}\|_F^2, \tag{4}$$

where $\boldsymbol{\beta} = \{\{\boldsymbol{\beta}_{i,j}\}_{j\in\mathcal{N}_i, j>i}\}_{i=1}^{L}$ are auxiliary variables, and ρ_β is a penalty parameter. In (4), The proximal term plays a crucial role in obtaining convergent results. It regulates the behavior of the algorithm in both the $\mathbf{Z}$-update step and indirectly in the $\boldsymbol{\lambda}$-update step by incorporating the Moreau envelope function, resulting in provable convergence. According to our proposed proximal ADMM algorithm, the $(k+1)$th iteration is as follows:

$$\mathbf{X}^{(k+1)} = \arg\min_{\mathbf{x}} \Psi_{\rho_\lambda,\rho_\beta}\left(\mathbf{X}, \mathbf{Z}^{(k)}, \boldsymbol{\beta}^{(k)}, \boldsymbol{\lambda}^{(k)}\right), \tag{5a}$$

$$\mathbf{Z}^{(k+1)} = \arg\min_{\mathbf{Z}} \Psi_{\rho_\lambda,\rho_\beta}\left(\mathbf{X}^{(k+1)}, \mathbf{Z}, \boldsymbol{\beta}^{(k)}, \boldsymbol{\lambda}^{(k)}\right), \tag{5b}$$

$$\boldsymbol{\beta}^{(k+1)} = \boldsymbol{\beta}^{(k)} - \eta\left(\boldsymbol{\beta}^{(k)} - \boldsymbol{Z}^{(k+1)}\right), \tag{5c}$$

$$\boldsymbol{\lambda}_{i,j}^{i,(k+1)} = \boldsymbol{\lambda}_{i,j}^{i,(k)} + \rho_\lambda\left(\mathbf{x}_i^{(k+1)} - \mathbf{z}_{i,j}^{(k+1)}\right), \forall e_{i,j} \in \mathcal{E} \tag{5d}$$

$$\boldsymbol{\lambda}_{i,j}^{j,(k+1)} = \boldsymbol{\lambda}_{i,j}^{j,(k)} + \rho_\lambda\left(\mathbf{x}_j^{(k+1)} - \mathbf{z}_{i,j}^{(k+1)}\right), \forall e_{i,j} \in \mathcal{E} \tag{5e}$$

where $\eta \in (0, 2)$.

More precisely, each $\mathbf{x}_i$ can be updated individually in update-step $\mathbf{X}$, which after several simplifications, can be stated as follows:

$$\begin{aligned} \mathbf{x}_i^{(k+1)} = \mathrm{Prox}_{f_i}\Bigg(& \frac{\sum_{j\in\mathcal{N}_i, j>i} \mathbf{z}_{i,j}^{(k)} - \frac{\lambda_{i,j}^{i,(k)}}{\rho_\lambda}}{|\mathcal{N}_i|} \\ & + \frac{\sum_{j\in\mathcal{N}_i, j<i} \mathbf{z}_{j,i}^{(k)} - \frac{\lambda_{j,i}^{i,(k)}}{\rho_\lambda}}{|\mathcal{N}_i|}; \frac{1}{\rho_\lambda|\mathcal{N}_i|}\Bigg) \end{aligned} \tag{6}$$

Algorithm 1: Moreau envelope ADMM for distributed optimization (MADM)

Initialize $\mathbf{X}^{(0)}$, $\mathbf{Z}^{(0)}$, $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\lambda}^{(0)}$, ρ_β, ρ_λ and $\eta \in (0,2)$;
repeat
  for $i \in [L]$ **do**
    Update $\mathbf{x}_i$ as:
    $$\mathbf{x}_i^{k+1} = \text{Prox}_{f_i}\left(\sum_{j\in\mathcal{N}_i} \frac{\mathbf{z}_{i,j}^{(k)} - \frac{\lambda_{i,j}^{i,(k)}}{\rho_\lambda}}{|\mathcal{N}_i|}; \frac{1}{\rho_\lambda|\mathcal{N}_i|}\right);$$
  end
  Each agent sends its local vector $\mathbf{x}_i^{k+1}$ to neighboring agents;
  for $i \in [L]$ **do**
    for $j \in \mathcal{N}_i$ **do**
      Update $\mathbf{z}_{i,j}$ as: $\mathbf{z}_{i,j}^{(k+1)} = \frac{\rho_\lambda\left(\mathbf{x}_j^{(k+1)}+\mathbf{x}_i^{(k+1)}\right)+\rho_\beta\boldsymbol{\beta}_{i,j}^{(k)}+\lambda_{i,j}^{i,(k)}+\lambda_{i,j}^{j,(k)}}{2\rho_\lambda+\rho_\beta}$;
      Update $\boldsymbol{\beta}_{i,j}$ as: $\boldsymbol{\beta}_{i,j}^{(k)} - \eta\left(\boldsymbol{\beta}_{i,j}^{(k)} - \boldsymbol{z}_{i,j}^{(k+1)}\right)$;
      Update $\boldsymbol{\lambda}_{i,j}^{i}$ as:
      $\boldsymbol{\lambda}_{i,j}^{i,(k+1)} = \boldsymbol{\lambda}_{i,j}^{i,(k)} + \rho_\lambda\left(\mathbf{x}_i^{(k+1)} - \mathbf{z}_{i,j}^{(k+1)}\right)$;
      Update $\boldsymbol{\lambda}_{i,j}^{j}$ as:
      $\boldsymbol{\lambda}_{i,j}^{j,(k+1)} = \boldsymbol{\lambda}_{i,j}^{j,(k)} + \rho_\lambda\left(\mathbf{x}_j^{(k+1)} - \mathbf{z}_{i,j}^{(k+1)}\right)$;
    end
  end
until *the convergence criterion is met*;

Also, in the $\mathbf{Z}$-update step, $\mathbf{z}_{i,j}$ can be updated separately as:

$$\begin{aligned}\mathbf{z}_{i,j}^{(k+1)} = \arg\min_{\mathbf{z}_{i,j}} \Big(&(\lambda_{i,j}^{i,(k)})^{\mathrm{T}}(\mathbf{x}_i^{(k+1)} - \mathbf{z}_{i,j}) \\ &+ (\lambda_{i,j}^{j,(k)})^{\mathrm{T}}(\mathbf{x}_j^{(k+1)} - \mathbf{z}_{i,j}) + \frac{\rho_\lambda}{2}\|\mathbf{x}_j^{(k+1)} - \mathbf{z}_{i,j}\|^2 \\ &+ \frac{\rho_\lambda}{2}\|\mathbf{x}_i^{(k+1)} - \mathbf{z}_{i,j}\|^2 + \frac{\rho_\beta}{2}\|\mathbf{z}_{i,j} - \boldsymbol{\beta}_{i,j}^{(k)}\|^2\Big), \quad (7)\end{aligned}$$

which can be simplified as follows:

$$\mathbf{z}_{i,j}^{(k+1)} = \frac{\rho_\lambda\left(\mathbf{x}_j^{(k+1)} + \mathbf{x}_i^{(k+1)}\right) + \rho_\beta\boldsymbol{\beta}_{i,j}^{(k)} + \lambda_{i,j}^{i,(k)} + \lambda_{i,j}^{j,(k)}}{2\rho_\lambda + \rho_\beta}. \quad (8)$$

Moreover, for each $\boldsymbol{\beta}_{i,j}$ we have:

$$\boldsymbol{\beta}_{i,j}^{(k+1)} = \boldsymbol{\beta}_{i,j}^{(k)} - \eta\left(\boldsymbol{\beta}_{i,j}^{(k)} - \boldsymbol{z}_{i,j}^{(k+1)}\right). \quad (9)$$

By introducing $\boldsymbol{\lambda}_{i,j}^{i}$, $\boldsymbol{\lambda}_{i,j}^{j}$, $\mathbf{z}_{i,j}$, and $\boldsymbol{\beta}_{i,j}$ to represent $\boldsymbol{\lambda}_{j,i}^{i}$, $\boldsymbol{\lambda}_{j,i}^{j}$, $\mathbf{z}_{j,i}$, and $\boldsymbol{\beta}_{j,i}$, respectively in each agent i, the proposed method is simplified and summarized in Algorithm 1.

IV. Convergence proof

This section presents the convergence analysis for Algorithm 1. Several conventional assumptions are made to build our convergence proof.

Assumption 1. *The undirected graph $\mathcal{G}$ is connected.*

Assumption 2. *$\Psi_{\rho_\lambda,\rho_\beta}\left(\mathbf{X}^{(k)}, \mathbf{Z}^{(k)}, \boldsymbol{\beta}^{(k)}, \boldsymbol{\lambda}^{(k)}\right)$ is lower bounded, and $\left(\mathbf{X}^{(k)}, \mathbf{Z}^{(k)}, \boldsymbol{\beta}^{(k)}, \boldsymbol{\lambda}^{(k)}\right)$ are bounded, in each iteration k.*

Remark. *It can be shown that for coercive functions*[1] *Assumption 2 holds true.*

Assumption 3. *Local objectives $f_i(\cdot), \forall i \in [L]$, are continuous, and weakly convex by parameter ρ_f.*

The proof of convergence relies on a canonical methodology as described in [25, Theorem 2.9]. Each algorithm iteration has only one increasing step, which is the $\boldsymbol{\lambda}$-update step. As the gradient of the Moreau envelope is related to the proximal function, the amount of increase in the $\boldsymbol{\lambda}$-update step ($\rho_\lambda^{-1}\|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\|^2$), is bounded based on the primal and auxiliary variables. Thus, tuning the parameters lets us guarantee the *sufficient decrease condition* of [25, Theorem 2.9]. In addition, the subgradient of the proximal augmented Lagrangian based on each of its inputs can easily be shown to be bound in each iteration, which is sufficient to validate the *bounded subgradient condition* of the [25, Theorem 2.9]. Finally, by having the boundedness assumption and knowing that the proximal augmented Lagrangian is continuous based on each of its inputs, it can be shown that the *continuity condition* of [25, Theorem 2.9] holds.

Lemma 1. *Function $g(\cdot)$, $\forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, satisfies condition:*

$$\|\nabla\mathcal{M}_{g(\cdot)}(\mathbf{u},\gamma) - \nabla\mathcal{M}_{g(\cdot)}(\mathbf{v},\gamma)\| = 0. \quad (10)$$

Lemma 2. *If Assumption 1 is held, for any $m \geq 1$, the following inequality is held:*

$$\begin{aligned}&\|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\|_F^2 \leq \\ &\qquad \rho_\beta^2\left(\|\mathbf{Z}^{(k+1)} - \mathbf{Z}^{(k)}\|_F^2 + \|\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(k-1)}\|_F^2\right) \quad (11)\end{aligned}$$

Proof. The lemma is proved by combining Lemma 1 with [26, Lemma 4]. □

Lemma 3. *Assuming Assumptions 1, 2, and 3 and $\rho_\lambda|\mathcal{N}_i| > \rho_f, \forall i \in [L]$, the following inequality holds:*

$$\begin{aligned}&\Psi_{\rho_\lambda,\rho_\beta}\left(\mathbf{X}^{(k)}, \mathbf{Z}^{(k)}, \boldsymbol{\beta}^{(k)}, \boldsymbol{\lambda}^{(k)}\right) - \\ &\qquad \Psi_{\rho_\lambda,\rho_\beta}\left(\mathbf{X}^{(k+1)}, \mathbf{Z}^{(k+1)}, \boldsymbol{\beta}^{(k+1)}, \boldsymbol{\lambda}^{(k+1)}\right) \geq \quad (12a)\\ &C(\rho_\lambda)\|\mathbf{X}^{(k+1)} - \mathbf{X}^{(k)}\|_F^2 + \left(\frac{\rho_\lambda}{2} + \frac{\rho_\beta}{2}\right)\|\mathbf{Z}^{(k+1)} - \mathbf{Z}^{(k)}\|_F^2 \quad (12b)\\ &\qquad + \frac{\rho_\beta}{2}\left(\frac{2}{\eta} - 1\right)\|\boldsymbol{\beta}^{(k+1)} - \boldsymbol{\beta}^{(k)}\|_F^2 \quad (12c)\\ &-\rho_\lambda^{-1}\rho_\beta^2\left(\|\mathbf{Z}^{(k+1)} - \mathbf{Z}^{(k)}\|_F^2 + \|\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(k-1)}\|_F^2\right), \quad (12d)\end{aligned}$$

where $C(\cdot)$ is a function with positive value for ρ_λ.

Proof. Equation (12b) is derived based on the weak convexity of local functions, and $g(\cdot)$, while (12c) is the result of

[1] A function $f(\cdot)$ is coercive if $f(\mathbf{x})\mathbf{x} \to \infty$ as $\|\mathbf{x}\| \to \infty$

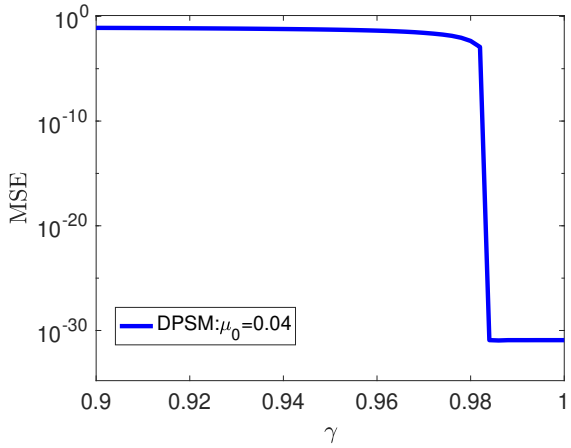

Fig. 1: MSE versus γ

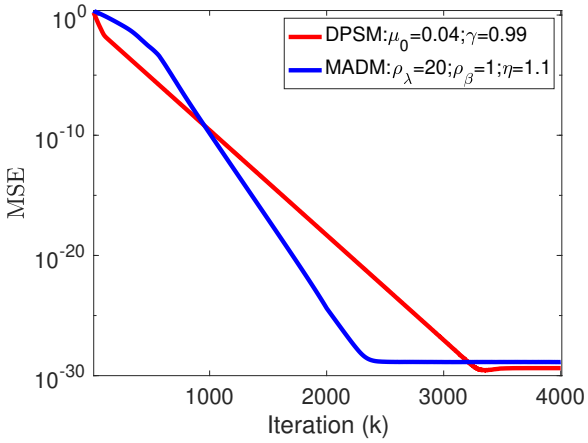

Fig. 2: MSE versus iteration

expanding the amount of change from the $\boldsymbol{\beta}$-update step. Further, (12d) derives from multiplying the bound obtained from Lemma 2 with $-\rho_\lambda^{-1}$, which gives an upper bound for the amount of change in the $\boldsymbol{\lambda}$-update step. □

Theorem 1. *By having* $\rho_\lambda|\mathcal{N}_i| > \rho_f, \forall i \in [L]$, $\frac{1}{\eta} \geq \frac{1}{2} + \rho_\lambda^{-1}\rho_\beta$, *and* $\rho_\lambda \geq \frac{(2\sqrt{2}-1)}{2}\rho_\beta$, *if Assumptions 1,2, and 3 hold, the algorithm 1 converges to a stationary point.*

Proof. The *sufficient decrease condition* of [25, Theorem 2.9] holds when ρ_λ, ρ_β, and ρ_β satisfy the condition of Theorem 1, by Lemma 3. The same results are obtained for the *bounded subgradient condition* of [25, Theorem 2.9] when it depends on the norm of the successive difference of the variables. Finally, employing Assumption 2 and knowing that the proximal augmented Lagrangian is continuous for each of its inputs, we can prove the *continuity condition* of [25, Theorem 2.9], which completes the proof. □

V. Simulation Results

In this section, we evaluate the performance of the MADM by conducting simulations of distributed robust phase retrieval with the objective function:

$$\hat{\mathbf{x}} = \min_{\mathbf{x}} \frac{1}{m} \sum_{i=1}^{L} |\langle \mathbf{a}_i, \mathbf{x}\rangle^2 - b_i^2|, \tag{13}$$

where $\mathbf{x}$ is the target signal, $\mathbf{a}_i$ is the measurement and b_i is the observation in node i. We assume that each node i has one measurement and observation, $\mathbf{a}_i \in \mathbb{R}^N \overset{\text{i.i.d}}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{1}), \forall i \in [L]$, and $\mathbf{x} \in \mathbb{R}^N \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$. For simplicity, we assume a noiseless setting with $b_i = \langle \mathbf{a}_i, \mathbf{x}\rangle, \forall i \in [L]$. All simulations are performed by averaging over 50 trials, and in each case, an Erdös-Rényi graph consisting of $L = 50$ nodes was generated as the communication network. To evaluate the performance of this method, we also simulated the distributed projected subgradient method (DPSM) proposed in [19]. The mean square error (MSE) $:= \frac{\sum_{i=1}^{L} ||\hat{\mathbf{x}}_i - \mathbf{x}||_2^2}{L}$ was utilized as the performance measure. Moreover, $\mathbf{z}_{i,j}^{(0)}, \forall e_{i,j} \in \mathcal{E}$ in MADM and $\mathbf{x}_i^{(0)}, \forall i \in \mathcal{V}$ in DPSM were initialized based on the procedure proposed in [27, Sec. 4.2].

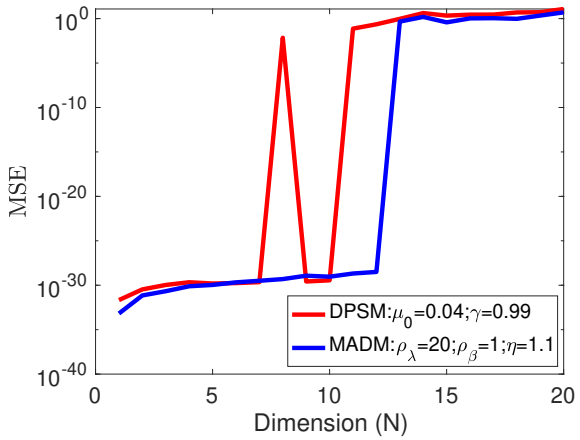

Fig. 3: MSE versus dimension

We first compare the convergence rate and efficiency of the two algorithms. The dimension of the target signal was $N = 10$, and for the DPSM, μ_0 was 0.04, while γ was chosen through a grid search to achieve a minimum and stable error with fast convergence. Fig. 1 illustrates the results of the grid search. We see that the DPSM highly depends on the choice of γ. In our algorithm we set $\rho_\lambda = 20$, $\rho_\beta = 1$, and $\eta = 1.1$. These values satisfy the conditions in Theorem 1. Fig. 2 shows that MADM can achieve a faster convergence rate than DPSM while maintaining a similar MSE.

Next, we study the robustness of the algorithms as a function of the dimension of the target signal for fixed parameters; N ranged from 1 to 20. A comparison of MADM and DPSM behavior under different dimensions is illustrated in Fig. 3. We see that MADM is more stable than DPSM when parameters are fixed. Although Fig. 1 indicates $\gamma = 0.99$ is in the safe zone for $N = 10$, it fails for $N = 8$ and $N > 10$.

VI. Conclusions

This paper presented a new proximal variant of the ADMM algorithm, named MADM, for solving distributed optimization problems. Our analysis demonstrated that the proposed method could be applied to weakly convex functions under mild conditions. In particular, we derived a bound on the change in the dual variable update step by leveraging the relationship between the gradient of the Moreau envelope function and the proximal function. This allowed us to ensure convergence to a stationary point. The simulation results showed that MADM outperforms subgradient methods in terms of speed and robustness. These findings suggest that MADM can be a promising tool for solving a wide range of distributed optimization problems in practice.

References

[1] V. C. Gogineni, S. Werner, Y.-F. Huang, and A. Kuh, "Communication-efficient online federated learning framework for nonlinear regression," in *Proc. - ICASSP IEEE Int. Conf. Acoust. Speech Signal Process.*, 2022, pp. 5228–5232.

[2] N. K. Venkategowda and S. Werner, "Privacy-preserving distributed maximum consensus," *IEEE Signal Process. Lett.*, vol. 27, pp. 1839–1843, Oct. 2020.

[3] Y. Wang, "Privacy-preserving average consensus via state decomposition," *IEEE Trans. Automat. Contr.*, vol. 64, no. 11, pp. 4711–4716, Mar. 2019.

[4] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Automat. Contr.*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[5] A. Nedic, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM J. Optim.*, vol. 27, no. 4, pp. 2597–2633, Dec. 2017.

[6] A. Makhdoumi and A. Ozdaglar, "Convergence rate of distributed ADMM over networks," *IEEE Trans. Automat. Contr.*, vol. 62, no. 10, pp. 5082–5095, Oct. 2017.

[7] N. Parikh and S. Boyd, *Proximal algorithms*. Now Publishers, Inc., 2014, vol. 1, no. 3.

[8] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[9] M. J. Wainwright and M. I. Jordan, *Graphical models, exponential families, and variational inference*. Now Publishers, Inc., 2008, vol. 1, no. 1–2.

[10] I. Lobel and A. Ozdaglar, "Distributed subgradient methods for convex optimization over random networks," *IEEE Trans. Automat. Contr.*, vol. 56, no. 6, pp. 1291–1306, Nov. 2010.

[11] R. Xin, S. Kar, and U. A. Khan, "Decentralized stochastic optimization and machine learning: A unified variance-reduction framework for robust performance and fast convergence," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 102–113, May 2020.

[12] C. Qian, X. Fu, N. D. Sidiropoulos, L. Huang, and J. Xie, "Inexact alternating optimization for phase retrieval in the presence of outliers," *IEEE Trans.Signal Process.*, vol. 65, no. 22, pp. 6069–6082, Nov. 2017.

[13] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding blind deconvolution algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2354–2367, Dec. 2011.

[14] S. Ling and T. Strohmer, "Self-calibration and biconvex compressive sensing," *Inverse Problems*, vol. 31, no. 11, p. 115002, Sep. 2015.

[15] D. Davis and D. Drusvyatskiy, "Stochastic model-based minimization of weakly convex functions," *SIAM J. Optim.*, vol. 29, no. 1, pp. 207–239, Jan. 2019.

[16] X. Yi, S. Zhang, T. Yang, T. Chai, and K. H. Johansson, "Sublinear and linear convergence of modified ADMM for distributed nonconvex optimization," *IEEE Trans. Automat. Contr.*, June 2022.

[17] M. Hong, D. Hajinezhad, and M.-M. Zhao, "Prox-pda: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks," in *Proc. Mach. Learn. Res.*, 2017, pp. 1529–1538.

[18] J. C. Duchi and F. Ruan, "Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval," *Information and Inference: A Journal of the IMA*, vol. 8, no. 3, pp. 471–529, Sep. 2019.

[19] S. Chen, A. Garcia, and S. Shahrampour, "On distributed nonconvex optimization: Projected subgradient method for weakly convex problems in networks," *IEEE Trans. Automat. Contr.*, vol. 67, no. 2, pp. 662–675, Feb. 2021.

[20] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *J Sci. Comput.*, vol. 78, no. 1, pp. 29–63, Jan. 2019.

[21] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM J. Optim.*, vol. 26, no. 1, pp. 337–364, Jan. 2016.

[22] M. Yashtini, "Convergence and rate analysis of a proximal linearized ADMM for nonconvex nonsmooth optimization," *J. Glob. Optim.*, pp. 1–27, May 2022.

[23] A. Themelis and P. Patrinos, "Douglas–rachford splitting and ADMM for nonconvex optimization: Tight convergence results," *SIAM J. Optim.*, vol. 30, no. 1, pp. 149–181, Jan. 2020.

[24] J.-J. Moreau, "Proximité et dualité dans un espace hilbertien," *Bulletin de la Société mathématique de France*, vol. 93, pp. 273–299, 1965.

[25] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods," *Mathematical Programming*, vol. 137, no. 1-2, pp. 91–129, Feb. 2013.

[26] J. Zeng, W. Yin, and D.-X. Zhou, "Moreau envelope augmented lagrangian method for nonconvex optimization with linear constraints," *J. Sci. Comput.*, vol. 91, no. 2, pp. 1–36, Apr. 2022.

[27] E. J. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, Apr. 2015.