Linear Convergence of Primal-Dual Gradient Methods and their Performance in Distributed Optimization

Sulaiman A. Alghunaim* and Ali H. Sayed†, Fellow, IEEE

Abstract—In this work, we revisit a discrete implementation of the primal-descent dual-ascent gradient method applied to saddle-point optimization problems. Through a short proof, we establish linear (exponential) convergence of the algorithm for strongly-convex cost functions with Lipschitz continuous gradients. Unlike previous studies, we provide explicit upper bounds on the step-size parameters for stable behavior and on the resulting convergence rate and highlight the importance of having two different primal and dual step-sizes. We also explain the effect of the augmented Lagrangian penalty term into a new cost function (see Assumption 1) and consider the saddle point problem:

\begin{equation}
J_\lambda(w) = J(w) + \lambda^T(Bw - b)
\end{equation}

where \( w \in \mathbb{R}^M \) is the primal variable, \( \lambda \in \mathbb{R}^E \) is the dual variable, \( B \in \mathbb{R}^{E \times M} \), and \( b \in \mathbb{R}^E \). This saddle point formulation arises in constrained optimization problems of the form

\begin{equation}
\min_{w \in \mathbb{R}^M} J(w), \quad \text{s.t.} \quad Bw = b
\end{equation}

It is well known that if a point \((w^*, \lambda^*)\) exists that solves (1), then \( w^* \) is an optimal solution to the constrained problem when strong duality holds, which is the case under our assumptions [1]. This paper focuses on the primal-dual (PD) algorithm [3], which is aimed at solving (1), establishes its linear convergence properties and studies its performance in distributed optimization problems. In this algorithm \( \nabla J(.) \) denotes the gradient of \( J(.) \) and \( (\mu_w, \mu_\lambda) \) are positive step-sizes (learning rates) chosen by the designer. Note that this is an incremental implementation since the dual update (3b) uses the most recent primal variable \( w_i \) and not \( w_{i-1} \).

Algorithms of the form (3) have been applied in various applications including but not limited to wireless systems [2], power systems [4], reinforcement learning [5], and network utility maximization [6], [7].

This work was supported in part by NSF grant CCF-1524250.

* S. A. Alghunaim is with the Electrical and Computer Engineering Department, University of California at Los Angeles (UCLA). Email: salghunaim@ucla.edu.
† A. H. Sayed is with the Ecole Polytechnique Federale de Lausanne EPFL, School of Engineering, CH-1015 Lausanne, Switzerland e-mail: ali.sayed@epfl.ch.

Algorithm (Incremental first-order PD gradient method)
Setting: Choose positive step-sizes \( \mu_w \) and \( \mu_\lambda \) and let \( w_{k-1} \) and \( \lambda_{k-1} \) be arbitrary initial conditions.

\begin{align}
w_i &= w_{i-1} - \mu_w \left( \nabla J(w_{i-1}) + B^T \lambda_{i-1} \right) \\
\lambda_i &= \lambda_{i-1} + \mu_\lambda (Bw_i - b)
\end{align}

A. Related Works

There exists a large body of literature on primal-dual saddle-point algorithms – see [6]–[14] and the references therein, including the seminal work [8], which proposed recursions of the type recursions (3a)–(3b) and established their convergence. These works focus on proving convergence to an optimal solution without providing convergence rates, provide sub-linear convergence rates (e.g., \( \frac{1}{k} \) where \( i \) is the iteration index), or show linear convergence from a starting point that is sufficiently close to a solution (local convergence). Some other works examined linear convergence under different settings.

The work [15] focuses on the continuous version of the primal-dual gradient dynamics and establishes linear convergence but using the augmented Lagrangian (AL). Similarly, the work [16] proves linear convergence for a continuous primal-dual gradient dynamics on a smoothed AL called “proximal augmented Lagrangian”, which can handle a non-smooth term. The work [17] focuses on the continuous version of the primal-dual dynamics for problem (2) with additional affine inequality constraints and establishes exponential convergence under the assumption that \( J(w) \) is twice-differentiable with upper and lower bounded Hessian (i.e., strongly convex and smooth in addition to twice differentiability). All these works focus on continuous dynamics and, moreover, they require a full row rank condition on the constraint matrix \( B \) to establish their result.

It was shown in [17] that if the continuous dynamics is discretized using Euler discretization, then the discrete version converges linearly under small enough step sizes. However, no upper bound is given on the step-size. Moreover, the derived linear convergence bound depends on the
continuous dynamics bounds. Note that Euler discretization uses identical step-sizes for the primal and dual updates (i.e., $\mu_w = \mu_\lambda$) and results in a non-incremental primal-dual dynamics, i.e., the dual update does not use the most recent primal update. Similarly, it was shown in [18] that the discretized version of the algorithm from [16] also converges linearly for small enough step-sizes. Wider stability bounds are further explicitly given by solving a linear matrix inequality (LMI) test, which guarantees linear convergence but no explicit bounds on convergence rate are given. Both works [17], [18] rely on other results (e.g., Euler discretization or LMI test) to establish linear convergence of the discrete versions.

Note that there exists works such as [19], [20] that establish linear convergence of primal-dual algorithms, which solve a general saddle point problem with $L(w, \lambda) = J(w) + R(w) + \lambda^T Bw - g(\lambda)$ where $R(.)$ and $g(.)$ are not necessarily differentiable. However, the work [19] requires both the primal and dual functions ($J(.) + R(.)$ and $g(.)$) to be strongly-convex functions while the work [20] establishes linear convergence when both primal and dual functions ($J(.) + R(.)$ and $g(.)$) are strongly sub-regular, which is a weaker condition than strong convexity, or under the special case where all the functions $J(.)$, $R(.)$ and, $g(.)$ are piecewise linear-quadratic (PLQ) (i.e., can be represented as the union of finitely many polyhedral sets) – see [20] for details. However, these conditions are not met in our problem since in our problem $g(\lambda) = 0$ (not strongly sub-regular) and $J(.)$ is not necessarily PLQ. In this case, the matrix $B$ plays a critical role for linear convergence, which was not studied in these works. Similarly, the works [21], [22] also establish linear convergence of general saddle-point problems with conditions on both primal and dual functions; moreover, the recursion of the algorithm requires the proximal mappings of both $J(.) + R(.)$ and $g(.)$, which is different from the type of recursions considered in this work.

The work [23] establishes the linear convergence of a primal-dual gradient algorithm for saddle-point problems with $L(w, \lambda) = J(w) + \lambda^T Bw - g(\lambda)$, where $J(w)$ is convex and smooth, and $g(\lambda)$ is strongly-convex and smooth. Unlike the current work, the algorithm used in [23] is non-incremental; moreover, a full column rank condition on $B$ and particular fixed step-sizes are needed to establish linear convergence – [23, Theorem 3.1].

Recently many first-order primal-dual gradient algorithms have been used to solve distributed optimization problems – see [24]–[28] and references therein, which are based on augmented Lagrangian (AL) formulations and incremental updates. They have been shown to achieve linear convergence under strong-convexity even though the consensus constraint matrix is not full rank. However, the analysis techniques used to establish the convergence of these methods depend on the particular consensus constraint matrix. Moreover, unlike these works, due to our unified Lagrangian and AL framework, we clarify the effect of the AL penalty term on the performance of these types of distributed algorithms. The work [29] focuses on distributed optimization problems for mean-square-error costs and studies non-incremental primal-dual methods with identical step-sizes. It was found in [29] that unlike AL methods, Lagrangian methods suffer from stability issues when the individual costs are not strongly-convex. This work generalizes this result to general costs; moreover, we show that Lagrangian methods can use larger step-sizes compared to AL methods and, more importantly, we show how the AL penalty term affects the convergence rate.

**B. Contribution**

Given the above, our contribution is threefold: 1) Through a self-contained short proof and without requiring $B$ to have full row rank, we establish the linear convergence of the incremental implementation; 2) We provide simple upper bounds on the step-sizes and the convergence rate. Unlike previous works, our convergence rate bound helps clarify the effect of $B$ on the rate of convergence and the importance of having different primal and dual step-sizes; 3) We show the effect of the augmented penalty term on the stability and convergence rate of distributed optimization problems. It is found that the penalty term can greatly improve the convergence rate when the individual costs are ill-conditioned but the aggregate cost is well conditioned.

**Notation:** For a matrix $A \in \mathbb{R}^{M \times N}$, $\sigma_{\max}(A)$ denotes the maximum singular value of $A$. $\sigma(A)$ denotes the minimum non-zero singular value. For a vector $x \in \mathbb{R}^M$ and a positive constant $c > 0$, we let $\|x\|^2_c$ denote the weighted norm $c\|x\|^2$.

**II. CONVERGENCE RESULT**

This section gives the auxiliary results leading to the main convergence result. We start with the following condition on the cost function.

**Assumption 1. (Cost function):** It is assumed that a solution exists for problem (1). It is also assumed that the cost function $J(w)$ is strongly convex, namely,

$$(x - y)^T (\nabla J(x) - \nabla J(y)) \geq \nu\|x - y\|^2$$

and has $\delta$-Lipschitz continuous gradients:

$$\|\nabla J(x) - \nabla J(y)\| \leq \delta\|x - y\|$$

for some strictly positive scalars $0 < \nu < \delta$.

It is known that a pair $(w^*, \lambda^*)$ is an optimal solution to (1) if, and only if, it satisfies the optimality conditions (1):

$$\nabla J(w^*) + B^T \lambda^* = 0$$

$$Bw^* - b = 0$$

Since $L(w, \lambda)$ is strongly convex w.r.t. $w$, $w^*$ is unique and coincide with the minimizer of (2) – see [1, Example 5.4]. From (6a) and (6b) and uniqueness of $w^*$, $\lambda^*$ will be unique if $B$ has full row rank. In general $\lambda^*$ is not necessarily unique. Motivated by [30], we will characterize a particular dual solution that we later show convergence to. For that result and later analysis, we need the following result.
Lemma 1. If \( \lambda_x \) is in the range space of \( B \in \mathbb{R}^{E \times M} \), then it holds that:
\[
\|B^T \lambda_x\|^2 \geq \sigma^2(B) \|\lambda_x\|^2
\]
where \( \sigma(B) \) denotes the minimum non-zero singular value of \( B \).

**Proof:** Introduce the truncated singular value decomposition [31] of the positive semi-definite matrix \( B^T B = U_r \Sigma_r U_r^T \), where \( U_r \in \mathbb{R}^{M \times r} \) (\( r \) denotes the rank of \( B^T B \)) with \( U_r^T U_r = I_r \) and \( \Sigma_r > 0 \) is a diagonal matrix with entries equal to the non-zero eigenvalues of \( B^T B \) (i.e., the squared non-zero singular values of \( B \)). Since \( \lambda_x \) is in the range space of \( B \), it holds that \( \lambda_x = Bx \) for some \( x \). Then,
\[
\|B^T \lambda_x\|^2 = \|B^T Bx\|^2 = x^T U_r \Sigma_r^2 U_r^T x \\
\geq \sigma^2(B) x^T U_r \Sigma_r U_r^T x \tag{8}
\]
The result follows since \( x^T U_r \Sigma_r U_r^T x = \|\lambda_x\|^2 \).

Lemma 2. (Particular dual \( \lambda_d^* \)) There exists a unique optimal dual variable, denoted by \( \lambda_d^* \), lying in the range space of \( B \).

**Proof:** The argument is motivated by [30]. Any solution \( \lambda^* \) of the linear system of equations given in (6a) can be decomposed into two parts \( \lambda^* = \lambda_2^* + \lambda_d^* \), where \( \lambda_2^* \) is in \( \text{Range}(B) \) and \( \lambda_d^* \in \text{null}(B^T) \) – see [31]. Therefore, if \( (w^*, \lambda^*) \) satisfies (6), then \( (w^*, \lambda_d^*) \) also satisfies (6). We now show \( \lambda_d^* \) is unique by contradiction. Assume we have two distinct dual solutions \( \lambda_{d1}^* = Bx_1 \) and \( \lambda_{d2}^* = Bx_2 \) lying in the range space of \( B \). Then, substituting into (6a) and subtracting, we get \( B^T B(x_1 - x_2) = 0 \). It follows that \( \|B(x_1 - x_2)\|^2 = 0 \) and, consequently, \( B(x_1 - x_2) = 0 \). This means that \( \lambda_{d1}^* = Bx_1 = Bx_2 = \lambda_{d2}^* \), which is a contradiction.

From (36) we know that \( \lambda_i = \lambda_{i-1} + \mu_i (Bw_i - b) \). Therefore, from the fact that \( b = Bw^* \), \( \lambda_i \) will be in the range space of \( B \) if \( \lambda_{i-1} \) belongs to the range space of \( B \) or \( \lambda_{i-1} = 0 \). Thus, \( \{\lambda_i\}_{i \geq 0} \) will always remain in the range space of \( B \) if \( \lambda_{i-1} = 0 \) or \( \lambda_{i-1} \) belongs to the range space of \( B \). This observation will allow us to utilize the bound (8) to establish linear convergence to the particular saddle-point \( (w^*, \lambda_d^*) \) without requiring a rank condition on the matrix \( B \), which is often assumed in the literature.

We are now ready to establish our main result. Let \( \bar{w}_i \triangleq w_i - w^* \) and \( \bar{\lambda}_i \triangleq \lambda_i - \lambda_d^* \) denote the primal and dual errors, respectively.

**Theorem 1. (Linear convergence):** Suppose Assumption [1] holds and the step-sizes are positive and satisfy:
\[
\mu_w \leq \frac{\nu}{\delta^2}, \quad \mu_\lambda \leq \frac{\nu}{\sigma_{\text{max}}^2(B)} \tag{9}
\]
If \( \lambda_{i-1} = 0 \), then algorithm (3) converges exponentially fast to the unique saddle-point \( (w^*, \lambda_d^*) \), namely, it holds that
\[
\|\bar{w}_i\|^2 + \|\bar{\lambda}_i\|^2 \leq \gamma (\|\bar{w}_{i-1}\|^2 + \|\bar{\lambda}_{i-1}\|^2) \tag{10}
\]
where \( c_\lambda > 0, c_w > 0 \) and \( \gamma \triangleq \max \{\gamma_1, \gamma_2\} < 1 \) with \( \gamma_1 \triangleq 1 - \mu_w \nu + \mu_w^2 \delta^2 \) and \( \gamma_2 \triangleq 1 - \mu_\lambda \mu \nu \sigma^2(B) \).

The proof of this Theorem is given in the next section. This result guarantees linear convergence for any step-sizes satisfying (9). Moreover, the convergence rate is upper bounded by \( \gamma \), which shows the effect of \( B \) on the convergence rate. Moreover, convergence rate bound indicates that using different primal and dual step-sizes can be beneficial. For example, assume \( 1/\delta^2 < 1/\sigma_{\text{max}}^2(B) \) so that the primal-step size bound is smaller than the dual-step size bound. Now, if we use identical step-sizes, then we have to select \( \mu_w = \mu_\lambda < \min \{\nu/\delta^2, \nu/\sigma_{\text{max}}^2(B)\} = \nu/\delta^2 \) and thus \( \gamma_2 = 1 - \mu_\lambda \nu \sigma^2(B) \). However, when we have two different step-sizes then \( \gamma_2 = 1 - \mu_w \mu_\lambda \sigma^2(B) \leq 1 - \mu_w \nu \sigma^2(B) \) since \( \mu_\lambda \) can be chosen larger in this case.

**III. PROOF OF LINEAR CONVERGENCE**

Subtracting \( w^* \) and \( \lambda_d^* \) from both sides of (11a) and using the optimality conditions (6) we get the coupled error recursion:
\[
\bar{w}_i = \bar{w}_{i-1} - \mu_w (\nabla J(w_{i-1}) - \nabla J(w^*)) + B^T \bar{\lambda}_{i-1} \tag{11a}
\]
\[
\bar{\lambda}_i = \bar{\lambda}_{i-1} + \mu_\lambda B \bar{w}_i \tag{11b}
\]
Squaring both sides of (11a) and (11b) we get
\[
\|\bar{w}_i\|^2 = \|\bar{w}_{i-1}\|^2 - 2\mu_w \bar{\lambda}_{i-1}^T B \bar{w}_{i-1} \bar{w}_i + \mu_w^2 \|B^T \bar{\lambda}_{i-1}\|^2 \tag{12}
\]
\[
\|\bar{\lambda}_i\|^2 = \|\bar{\lambda}_{i-1}\|^2 + 2\mu_\lambda \|B \bar{w}_{i-1}\|^2 - 2\mu_\lambda \mu_w \|B^T \bar{\lambda}_{i-1}\|^2 \tag{13}
\]
Using the bound \( \|B \bar{w}_{i-1}\|^2 \leq \sigma_{\text{max}}^2(B) \|\bar{w}_{i-1}\|^2 \), multiplying equation (13) by \( c_\lambda \triangleq \mu_\lambda / \mu_w \) and adding to (12) gives:
\[
\|\bar{w}_i\|^2 + \|\bar{\lambda}_i\|^2 \leq \|\bar{w}_{i-1}\|^2 - \mu_w \|\nabla J(w_{i-1}) - \nabla J(w^*)\|^2 + \|\bar{\lambda}_{i-1}\|^2 - \mu_w \nu \sigma^2(B) \|\bar{\lambda}_{i-1}\|^2 \tag{14}
\]
where \( c_w \triangleq 1 - \mu_w \mu_\lambda \sigma^2_{\text{max}}(B) \). Let \( \tilde{\gamma} \triangleq (1 - 2\mu_w \nu + \mu_w^2 \delta^2) \), then it holds that
\[
\|\bar{w}_{i-1} - \mu_w \nabla J(w_{i-1}) - \nabla J(w^*)\|^2 \leq \tilde{\gamma} \|\bar{w}_{i-1}\|^2 \tag{15}
\]
This follows directly by expanding the square and using the bounds (11a) and (11b) given in Assumption [1]. Note that from Lemma 2 \( \lambda_d^* \) lies in the range space of \( B \). Moreover, since \( \lambda_{i-1} = 0 \), then we know that \( \lambda_i \) will always lie in the range space of \( B \). Thus, from (7) it holds that \( \|B^T \bar{\lambda}_{i-1}\|^2 \geq \sigma^2(B) \|\bar{\lambda}_{i-1}\|^2 \). Using this bound and (15) in (14), we get:
\[
\|\bar{w}_i\|^2 + \|\bar{\lambda}_i\|^2 \leq (1 - 2\mu_w \nu + \mu_w^2 \delta^2) \|\bar{w}_{i-1}\|^2 + (1 - \mu_w \nu \sigma^2(B)) \|\bar{\lambda}_{i-1}\|^2 \tag{16}
\]
Let $\gamma_1 = (1 - \mu_w, \nu + \mu_w^2 \delta^2)$ and $\gamma_2 = 1 - \mu_w, \mu_w \sigma^2_{B} (B)$ and add and subtract $\mu_w, \mu_w \sigma^2_{B} (B) \gamma_1 ||\bar{w}_{i-1}||^2$ to the right hand side of (16), we get

$$
||\bar{w}_{i}||^2_{w} + ||\bar{\lambda}_{i}||_{\lambda}^2 \leq \gamma_1 ||\bar{w}_{i-1}||^2_{w} + \gamma_2 ||\bar{\lambda}_{i-1}||_{\lambda}^2 + \mu_w, (\mu_w \sigma^2_{B} (B) \gamma_1 - \nu) ||\bar{w}_{i-1}||^2
$$

(17)

It holds that that $\gamma_1 = 1 - \mu_w, \nu + \mu_w^2 \delta^2 < 1$ if $\mu_w < \frac{\nu}{\delta^2}$. Moreover,

$$
\mu_w \sigma^2_{B} (B) \gamma_1 - \nu < \mu_w \sigma^2_{B} (B) - \nu \leq 0
$$

if $\gamma_1 < 1$ and $\mu_w \leq \frac{\nu}{\sigma^2_{B} (B)}$. Thus, under these conditions we have $\mu_w, (\mu_w \sigma^2_{B} (B) \gamma_1 - \nu) ||\bar{w}_{i-1}||^2 \leq 0$ and we can upper bound (17) by:

$$
||\bar{w}_{i}||^2_{w} + ||\bar{\lambda}_{i}||_{\lambda}^2 \leq \gamma_1 ||\bar{w}_{i-1}||^2_{\lambda} + \gamma_2 ||\bar{\lambda}_{i-1}||_{\lambda}^2
$$

(19)

Note that for positive step-sizes it holds that $c_{\lambda} = \frac{\mu_w}{\delta^2} > 0$. Under $\mu_w, \mu \leq \frac{\nu}{\sigma^2_{B} (B)}$, it also holds that $c_{\lambda} = 1 - \mu_w, \mu \sigma^2_{B} (B) > 0$ and $0 < \gamma_2 = 1 - \mu_w, \mu \sigma^2_{B} (B) < 1$. This condition is satisfied under condition (9) because under these conditions we have

$$
\mu_w, \mu \leq \frac{\nu^2}{\delta^2 \sigma^2_{B} (B)} \leq \frac{1}{\sigma^2_{B} (B)}
$$

(20)

where the last inequality hold because $\nu < \delta$. Therefore, under the condition (9) we can upper bound (19) by (10).

**Remark 1. (Tighter bounds)** Note that the bound (15) can be tightened so that $\gamma = 1 - 2, \mu_w, \nu + \mu_w^2, \nu^2 < 1$ if $\mu_w \leq \frac{\nu}{\delta^2}$, which is a well known bound – see [32, Theorem 3.12]. In this case, the primal step-size bound becomes $\mu_w \leq \frac{\nu}{\delta^2}$ and $\gamma_1 = 1 - \mu_w, \nu + \mu_w^2, \nu^2$. We are using (15) instead to make the argument self-contained.  

**IV. Application: Distributed Optimization**

In this section, we illustrate the benefit of the AL penalty term for distributed optimization problems. We start by describing the problem.

Consider a network of $K$ agents that are connected through a static and undirected network and interested in the following problem:

$$
\text{minimize}_{w \in \mathbb{R}^M} \sum_{k=1}^{K} J_k(w)
$$

(21)

where $J_k(w): \mathbb{R}^M \rightarrow \mathbb{R}$ is a local cost function associated with agent $k$. In order to derive the algorithm that solves (21) in a distributed manner, we will rewrite (21) in an equivalent constrained form. We introduce a combination matrix $A = [a_{sk}]$ associated with the network. The entry $a_{sk}$ is the weight used by agent $k$ to scale information arriving from agent $s$ with $a_{sk} = 0$ if $s$ is not a direct neighbor of agent $k$, i.e., there is no edge connecting them. The matrix $A$ is assumed to be primitive, i.e., there exists a large $p > 0$ such that all entries of $A^p$ are positive. We also assume $A$ to be symmetric, and doubly stochastic. There exists many rules to chose $A$ such as the Metropolis rule – see [33], which satisfy the primitiveness assumption as long as the network is connected. Under this assumption, it holds that $I_K - A$ is positive semi-definite and $(I_K - A)x = 0$ if, and only, if $x = e \in \mathbb{R}^K$ for any $e \in \mathbb{R}$. Therefore, if we let $w_k \in \mathbb{R}^M$ denote a local copy of $w$ available at agent $k$ and introduce the network quantities:

$$
\mathcal{W} \overset{\Delta}{=} \{w_1, \cdots, w_K\} \in \mathbb{R}^{KM}
$$

(22)

$$
\mathcal{B} \overset{\Delta}{=} (I_K - A) \otimes I_M, \quad \mathcal{J}(w) \overset{\Delta}{=} \sum_{k=1}^{K} J_k(w_k)
$$

(23)

Then, it holds that $\mathcal{B} \mathcal{W} = 0$ if, and only, if $w_k = w_s \forall k, s$. Thus, problem (21) is equivalent to the following penalized constrained problem:

$$
\text{minimize}_{w \in \mathbb{R}^{KM}} \mathcal{J}_p(w), \quad \text{s.t.} \ \mathcal{B} \mathcal{W} = 0
$$

(24)

where $\mathcal{J}_p(w) \overset{\Delta}{=} \mathcal{J}(w) + \frac{\rho}{2} ||\mathcal{B} \mathcal{W}||^2$ with $\rho \geq 0$ and $\mathcal{B}$ is the square root of the positive semi-definite matrix $\mathcal{B}$. A direct application of (3) to the Lagrangian of problem (24) (AL if $\rho > 0$) gives:

$$
\mathcal{W}_i = \mathcal{W}_{i-1} - \mu_w \nabla_w \mathcal{J}_p(\mathcal{W}_{i-1}) - \mu_w \mathcal{B} \mathcal{W}_{i-1}
$$

(25a)

$$
\lambda_i = \lambda_{i-1} + \mu_B \mathcal{B} \mathcal{W}_i
$$

(25b)

Recurrent (25) is not distributed yet because $\mathcal{B}$ need not have the network structure. However, this can be easily handled by a change of variable. Let $y_i = B^2 \lambda_i$ and multiply (25b) by $B^2$ gives:

$$
\mathcal{W}_i = \mathcal{W}_{i-1} - \mu_w \nabla_w \mathcal{J}_p(\mathcal{W}_{i-1}) - \mu_w \mathcal{B} \mathcal{W}_{i-1}
$$

(26a)

$$
\gamma_i = \gamma_{i-1} + \mu_B \mathcal{B} \mathcal{W}_i
$$

(26b)

Since $B$ has the network structure, then the $k$-th block of $\mathcal{B} \mathcal{W}_i$ has the distributed form $u_k = \sum_{s \in \mathcal{N}_k} a_{sk} w_s$ where $\mathcal{N}_k$ denotes the neighbors of agent $k$, including agent $k$. Therefore, recursion (26) is distributed and each can update its corresponding $k$-th blocks in $\mathcal{W}_i$ and $\gamma_i$.

**A. Relation to Other Algorithms**

Before we prove convergence of recursion (26) and show the influence of AL penalty term on its performance, we first show how the derivation of the AL PD algorithm (25) is related to some state of the art algorithms. To do so and for convenience, we rewrite (24) equivalently as

$$
\text{minimize}_{w \in \mathbb{R}^{KM}} \mathcal{J}_p(w), \quad \text{s.t.} \ \frac{1}{\mu} \mathcal{B} \mathcal{W} = 0
$$

(27)

where $\mathcal{J}_p(w) \overset{\Delta}{=} \mathcal{J}(w) + \frac{\rho}{2} ||\mathcal{B} \mathcal{W}||^2$, $\mu > 0$ and $j$ is either $j = 1$ or $j = 1/2$ depending on the algorithm derived.

1) EXTRA [34]: Note that the saddle point interpretation of EXTRA appeared in the work [35] — see also [28]. Applying the primal-dual gradient algorithm (3) to the Lagrangian of (27) with $j = 1/2$, $\mu_w = \mu$, $\mu_w = \mu$, and $\rho = \frac{1}{\mu}$ gives:

$$
\mathcal{W}_i = \mathcal{W}_{i-1} - \mu \nabla_w \mathcal{J}(\mathcal{W}_{i-1}) - \mathcal{B} \mathcal{W}_{i-1} - B^2 \gamma_{i-1}
$$

(28a)

$$
\lambda_i = \lambda_{i-1} + B^2 \gamma_i
$$

(28b)
By eliminating the dual-variable (see [35] for details), the above algorithm can be shown to be equivalent to the EXTRA algorithm in [34], which requires communicating the primal variable once per iteration. Note that (25) is equivalent to (25) if $B^\bot$ is replaced by $\frac{1}{\mu} B^\bot$ and $\mu = \mu_\lambda = \mu$, and $\rho = \frac{1}{\mu}$.

2) DIGing [26]: The DIGing algorithm [26], which is based on a gradient tracking scheme, admits a primal-dual interpretation as well. Consider the cost (27) with $\rho = \frac{1}{\mu}$ and $j = 1$. Applying the primal-dual gradient algorithm (3) for this case gives:

$$w_i = w_{i-1} - \mu \nabla_w J(w_{i-1}) - 2Bw_{i-1} - B\lambda_{i-1} \quad (29a)$$
$$\lambda_i = \lambda_{i-1} + B^\bot w_i \quad (29b)$$

By eliminating the dual variable, this algorithm can be shown to be equivalent to DIGing – see [26, Section 2.2]. We see that the difference from the EXTRA derivation is shown to be equivalent to DIGing – see [26, Section 2.2].

3) Exact diffusion [27]: Applying the primal-dual gradient algorithm (3) on the Lagrangian of (27) with $j = 1/2$, $\mu = \mu_\lambda = \mu$, and $\rho = \frac{1}{2\mu}$ gives:

$$w_i = w_{i-1} - \mu \nabla_w \mathcal{L}(w_{i-1}, \lambda_{i-1}) \quad (30a)$$
$$\lambda_i = \lambda_{i-1} + B^\bot w_i \quad (30b)$$

where

$$\mathcal{L}(w, \lambda) = J(w) + \frac{1}{4\mu} \|B^\bot w\|^2 + \frac{1}{\mu} \lambda^T B^\bot w \quad (31)$$

Different from a traditional gradient primal-descent (30a) that was used to derive EXTRA and (26), exact diffusion employs three incremental gradient steps (on the three terms in $\mathcal{L}(w, \lambda)$) with step-size $\mu$ to arrive at (see [27] for details):

$$w_i = \bar{A} \left( w_{i-1} - \mu \nabla_w J(w_{i-1}) \right) - B^\bot \lambda_{i-1} \quad (32a)$$
$$\lambda_i = \lambda_{i-1} + B^\bot w_i \quad (32b)$$

where $\bar{A} \triangleq I - \frac{1}{2} B = \frac{1}{2} (I + \mathcal{A})$ and $\mathcal{A} = \mathcal{A} \otimes I_M$. By eliminating the dual-variable, the above algorithm can be written in a primal form, which requires communicating the primal variable once per iteration – see [27] for details. Exact diffusion was shown to have wider stability range and better convergence performance than EXTRA – see [36].

It is worth mentioning that if we consider the penalized problem (27) without the consensus constraint and apply two incremental gradient descent steps for the two terms in the penalized cost $J_p(w) \triangleq J(w) + \frac{\eta}{\mu} \|B^\bot w\|^2$ with step-size $\mu$ and $\rho = \frac{1}{\mu}$, we arrive at:

$$z_i = \mathcal{A} \left( w_{i-1} - \mu \nabla_w J(w_{i-1}) \right) \quad (33)$$

which is the diffusion algorithm [27], [33]. The bias that arises from solving the penalized problem, rather than the original problem, can be corrected by employing exact diffusion [27].

B. AL Penalty Term Influence

We see that many algorithms can be derived by selecting different consensus constraint matrices and penalty parameters. Although, EXTRA and DIGing are traditional AL primal-dual methods, the analysis in this work does not cover these cases. This is because the step-size choice $\mu_\lambda = \mu_w = \mu$ and consensus constraint matrix $B = \frac{1}{2} I^2$, for these algorithms do not satisfy the bounds given in (9). On the other hand, unlike these methods, our analysis holds for general constraint matrix $B$ (not necessarily the consensus constraints) and, most importantly, it holds without the augmented penalty term (i.e., $\rho = 0$). This general form allows us to see the influence of the penalty parameter on the performance of such algorithms, which we explain after the next result.

Theorem 2. Let $(w^*, \lambda^*_0)$ be the optimal point for the Lagrangian of (24) where $\lambda^*_0$ lies in the range space of $B^\bot$. Assume that the cost $J(w)$ is $\nu$-strongly convex and $\delta$-Lipschitz. Under the step-size conditions in Theorem 1 with $B = \frac{1}{2} I^2$ and Lipschitz constant $\delta_\nu = \delta + \rho \sigma_{\text{max}}^2 (B^\bot)$, the recursions (26a)–(26b) converge linearly to $(w^*, B^\bot \lambda^*_0)$ if $\nu_{-1} = 0$.

Proof: Since $J(w)$ is $\nu$-strongly convex and $\delta$-Lipschitz, it holds that $J_p(w) = \sum_{k=1}^K J(k w) + \frac{\eta}{\mu} \|B^\bot w\|^2$ is $\nu$-strongly convex and $\delta_\nu = \delta + \rho \sigma_{\text{max}}^2 (B^\bot)$-Lipschitz. We know that under the conditions in Theorem 1, recursions (25a)–(25b) converges to the point $(w^*, \lambda^*_0)$ linearly if $\lambda_{-1} = 0$. Now, note that if $\lambda_{-1} = 0$ and $\nu_{-1} = 0$, then from (25a) and (25b), it holds that $\nu_i = B^\bot \lambda_i$ for all $i \geq -1$. Since $\nu_i$ lies in the range space of $B^\bot$, it follows from Lemma 1 that $\nu_i = 0 \iff \lambda_i = 0$. Therefore, if recursions (25a)–(25b) converge to $(w^*, \lambda^*_0)$, then recursions (26a)–(26b) converge to $(w^*, B^\bot \lambda^*_0)$.

It can be seen that the addition of the augmented term with $\rho > 0$ leads to a larger Lipschitz constant $\delta_\nu = \delta + \rho \sigma_{\text{max}}^2 (B^\bot) > \delta$ and consequently to a smaller stability range for $\mu_w$ compared to the case when $\rho = 0$. This means that the AL method has narrower stability range. However, the augmented term can be beneficial as we now explain. Note that the proof of Theorem 2 requires the augmented cost $J_p(w)$ to be $\nu_k$-strongly convex, so that $\nu = \sum_{k=1}^K \nu_k$. This implies that for $\rho = 0$, the cost $J_0(w) = J(w)$ is not strongly-convex (can even be non-convex) unless each individual cost is strongly convex, which can lead to convergence issues. This is not always the case when $\rho > 0$ as we now explain.

Assume that the aggregate cost $\sum_{k=1}^K J_k(w) : \mathbb{R}^M \to \mathbb{R}$ is $\bar{\nu}$-strongly-convex, which does not necessarily imply that each individual cost is strongly-convex – see simulation.
section. In this case, it can be shown that for \( \rho > 0 \), the augmented penalized cost \( J_\rho (w) \) is \( \kappa \)-restricted strongly convex\(^{1}\) with respect to \( v^* \) – see \([34, \text{Proposition 3.6}]\). Hence, our convergence proof is applicable in this case and the AL method will still exhibit linear convergence as long as the aggregate cost is strongly-convex; moreover, the convergence rate bound \( \gamma_1 \) will depend on \( \kappa \) instead of \( \nu \).

Note that if the individual costs are strongly-convex but ill-conditioned (\( \nu_k \) are close to zero and \( \delta \gg \nu_k \)), then \( \nu = \sum_{k=1}^{K} \nu_k \) can be much smaller than \( \kappa \) if \( \rho \) is large enough and the aggregate cost is well-conditioned. In this case, the AL method will have a better convergence rate due to smaller \( \gamma_1 \). However, when the individual costs are strongly-convex and well conditioned, then the AL penalty term is not that beneficial. The next section illustrates these conclusions by means of simulations.

V. SIMULATION

We consider the distributed optimization problem \([21]\) with quadratic costs \( J_k(w) = w^T R_k w + r_k^T w \) where \( w \in \mathbb{R}^n \), \( R_k \in \mathbb{R}^{n \times n} \), and \( r_k \in \mathbb{R}^n \). A network of \( K = 20 \) agents shown in top rightmost side of Fig. 1 was randomly generated across a \( 1.2 \times 1.2 \) grid and two agents are neighbors if they are less than \( d = 0.25 \) distance away from each other. This network is used for all implementations. The matrix \( A \) is generated using the Metropolis rule \([33]\). In all results, PD non-distributed refers to \([25]\) (\( \rho = 0 \)). PD distributed refers to \([26]\) (\( \rho = 0 \)), and AL PD distributed refers to \([26]\) with \( \rho > 0 \). The step-sizes are manually chosen to get the best possible convergence rate for each algorithm.

The top leftmost plot of Fig. 1 shows the evolution of the relative error \( \|w_i - w^*\|^2/\|w^*\|^2 \) over iterations. The matrix \( R_k > 0 \) is a random diagonal matrix with integer diagonal entries, each chosen between \([2, 8] \), which is well conditioned because \( 8/2 \) is not very large. Similarly, \( r_k \) is randomly generated vector with each entry uniformly selected between \([0, 2] \). The dual step-size is set to \( \mu_\lambda = 30 \) for all algorithms in the top leftmost plot of Fig. 1. All algorithms converge linearly with close performance. It is seen that PD non-distributed and PD distributed have identical performance. We also see that unlike PD distributed, AL PD distributed has a smaller primal stability range for a fixed dual step-size – see explanation below Theorem 2. To test the dual stability range, we fixed \( \mu_w = 0.07 \) and simulated PD distributed and AL PD distributed for different dual step-size. The results are shown in the middle leftmost plot of Fig. 1. We see that in this settings, the AL PD distributed also has a smaller dual stability range for a fixed primal step-size. Under the same set-up, we compare the algorithm with the Extra algorithm from \([34]\) and exact diffusion from \([27]\), both are AL based methods. The results are shown in the middle rightmost plot of Fig. 1. We see that all algorithms perform similarly. In this scenario, we do not see any advantages of AL methods compared to the Lagrangian method (\( \rho = 0 \)).

To show the advantages of the AL method, we consider the same setting as before but under two scenarios: non-convex individual costs or ill-conditioned strongly-convex costs. For the non-convex case, we let all entries to be zero except for the following diagonal entries: the \((k,k)\)-th diagonal entry for each agent (\( R_k(k,k) \)) are chosen randomly between \([2, 8] \), the entries \( R_k(k+1,k-1) = -R_k(k-1,k-1)/2 \) for all \( k \geq 2 \). In this case, the aggregate cost \( \sum_{k=1}^{K} (w^T R_k w + r_k^T w) \) is strongly convex since \( R = \sum_{k=1}^{K} R_k > 0 \), but the individual costs \( \{J_k(w)\}_{k \geq 2} \) are non-convex and therefore \( \nabla^2 J(w) = \sum_{k=1}^{K} (w_k^T R_k w_k + r_k^T w_k) \) is non-convex as well. This is because the Hessian \( \nabla^2 J(w) = \text{blkdiag} \{R_k\}_{k=1}^{K} \) is indefinite. The results of this set-up is shown in the bottom leftmost plot of Fig. 1. We see that the AL PD distributed method still converges linearly. However, the PD distributed method diverges even under much smaller step-sizes than the previous set-ups. The bottom rightmost plot of Fig. 1 shows the ill conditioned case. We let all entries to be zero except for the diagonal entries: the \((k,k)\)-th diagonal entry for each agent (\( R_k(k,k) \)) are chosen randomly between \([2, 8] \) and the other diagonal entries are chosen uniformly between \([0,1] \), which can be very small. In this case, we see that the Lagrangian methods performs poorly compared to the other PD AL method (\( \rho = 50 \)) and the other AL methods, which is in agreement with our conclusion from the previous section.

REFERENCES

[1] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge University Press, 2004.
[2] R. Ayadi, I. Kammoun, and M. Siala, “Optimization of the pulse shape of OFDM systems using the Arrow-Hurwicz algorithm,” in 4th International Symposium on Wireless Communication Systems, Dec. 2007, pp. 91–95, Trondheim, Norway.

[3] J. Chen and V. K. Lau, “Convergence analysis of saddle point problems in time varying wireless systems,” in Control and Applications to Network Optimization, Automatica, vol. 46, no. 12, pp. 1974–1981, 2010.

[4] J. Wang and N. Elia, “A control perspective for centralized and distributed coordination for economic dispatch under varying loads and generator commitment,” Automatica, vol. 74, pp. 183–193, 2016.

[5] S. V. Macac, J. Chen, S. Zazo, and A. H. Sayed, “Distributed policy evaluation under multiple behavior strategies,” IEEE Transactions on Automatic Control, vol. 60, no. 5, pp. 1260–1274, 2015.

[6] D. Fei, and F. Paganini, “Stability of primal–dual gradient dynamics and applications to network optimization,” Automatica, vol. 46, no. 12, pp. 1974–1981, 2010.

[7] J. Yuan, B. Ying, X. Zhao, and A. H. Sayed, “Exact diffusion for distributed optimization and learning—Part I: Algorithm development,” IEEE Transactions on Signal Processing, vol. 67, no. 3, pp. 708–723, Feb. 2019.

[8] D. Jakovetic, “A unification and generalization of exact distributed first-order methods,” IEEE Transactions on Signal and Information Processing over Networks, vol. 5, no. 1, pp. 31–46, 2019.

[9] Z. J. Towfic and A. H. Sayed, “Stability and performance limits of adaptive primal–dual networks,” IEEE Trans. Signal Process., vol. 63, no. 11, pp. 2888–2903, 2015.

[10] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, “On the linear convergence of the ADMM in decentralized consensus optimization,” IEEE Transactions on Signal Processing, vol. 62, pp. 1750–1761, 2014.

[11] A. J. Laub, Matrix Analysis For Scientists And Engineers. SIAM, PA, USA, 2004.

[12] S. Bubeck, “Convex optimization: Algorithms and complexity,” Foundations and Trends in Machine Learning, vol. 8, no. 3–4, pp. 231–357, 2015.

[13] A. H. Sayed, “Adaptation, learning, and optimization over networks,” Foundations and Trends in Machine Learning, vol. 7, no. 4–5, pp. 311–801, 2014.

[14] W. Shi, Q. Ling, G. Wu, and W. Yin, “Extra: An exact first-order algorithm for decentralized consensus optimization,” SIAM Journal on Optimization, vol. 25, no. 2, pp. 944–966, 2015.

[15] A. Mokhtari and A. Ribeiro, “DSA: Decentralized double stochastic averaging gradient algorithm,” Journal of Machine Learning Research (JMLR), vol. 17, no. 1, pp. 2165–2199, 2016.

[16] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, “Exact diffusion for distributed optimization and learning—Part II: Convergence analysis,” IEEE Transactions on Signal Processing, vol. 67, no. 3, pp. 724–739, Feb. 2019.