Robustness and Convergence Analysis of First-Order Distributed Optimization Algorithms over Subspace Constraints

Dennis J. Marquis, Dany Abou Jaoude, Mazen Farhood, and Craig A. Woolsey

Abstract—This paper extends algorithms that solve the distributed consensus problem to solve the more general problem of distributed optimization over subspace constraints. Leveraging the integral quadratic constraint framework, we analyze the performance of these generalized algorithms in terms of worst-case robustness and convergence rate. The utility of our framework is demonstrated by showing how one of the extended algorithms, originally designed for consensus, is now able to solve a multitask inference problem.

I. INTRODUCTION

There is considerable literature on distributed optimization algorithms that rely on local information exchange between agents over a network to solve the consensus problem. Decentralized gradient descent (DGD) has been shown to exhibit linear convergence to a fixed-point when minimizing strongly convex objective functions with a fixed step-size \( \mu \) [1]. Unlike its centralized counterpart, DGD converges to an \( O(\mu) \)-neighborhood of the optimal solution. Diminishing step-size schemes allow DGD to converge without a bias, but the resulting convergence rate is sub-linear. Algorithms have been developed that are capable of achieving linear convergence rates for strongly convex local objective functions without this fixed-point bias [2]–[8]. Techniques used to accelerate centralized algorithms have also been applied to the distributed setting. For example, Nesterov acceleration has been used to improve distributed algorithm convergence rates when the objective function has a high condition ratio [9]. Recently, these bias-correction algorithms have started to be analyzed in the stochastic setting [10], specifically to determine under which scenarios they outperform traditional methods like DGD.

In addition to algorithms designed exclusively for the consensus problem, there are algorithms that solve a more general class of problems: distributed optimization over subspace constraints. In [11], it is shown that many common distributed optimization problems can be cast as distributed subspace constrained optimization problems, including consensus optimization, coupled optimization, optimization under affine constraints, and band-limited graph signal estimation. Two iterative algorithms that use local information exchange to solve this type of problems are DiSPO [12] and the distributed adaptive strategy proposed in [11], [13], which will be referred to as DAS in the sequel. These two algorithms have update equations that are analogous to DGD and a diffusion form of DGD, respectively. Consequently, DiSPO and DAS also exhibit a fixed-point bias.

The integral quadratic constraint (IQC) framework [14] provides an approach to analyze the stability of uncertain dynamical systems by modeling these systems as a feedback interconnection between a linear time-invariant (LTI) system \( G \) and an uncertainty operator \( \Delta \) that lies in a pre-specified set \( \Delta \), described by an IQC. The original analysis conditions in the IQC framework were in the frequency domain. More recent works that rely on dissipativity theory have proposed IQC analysis conditions in the time domain; see, e.g., [15], [16], for relevant works in discrete time.

In [17], the IQC framework is adapted to analyze the convergence rate of optimization algorithms. Here, optimization algorithms are interpreted as discrete-time uncertain dynamical systems and the operator \( \Delta \) represents the nonlinear gradient computations. Semidefinite program (SDP) tools can solve the analysis problem. This approach is constructive and does not rely on algorithm-specific expert knowledge to produce algorithm performance guarantees. The IQC framework has also been used in algorithm design [18], [19].

In this paper, we extend the consensus algorithms that remove the fixed-point bias of DGD to solve the more general problem of distributed optimization over subspace constraints. These algorithms can now be applied to new problems, such as multitask inference. Additionally, we propose an IQC-based framework to analyze the performance of these generalized algorithms in terms of worst-case robustness and convergence rate. Our analysis framework is an extension of those in [8], [20], which are used to analyze the convergence rate of first-order distributed consensus optimization algorithms, and the frameworks in [21], [22], which are used to analyze the robustness of first-order centralized optimization algorithms. In addition to proposing extended algorithms for handling subspace constraints, our contributions consist of (1) handling a more general class of distributed optimization problems and (2) extending the robustness analysis results to the distributed setting. As an illustrative case study, our framework is used to compare the extended version of AugDGM [7] to DAS for solving a multitask inference problem.

This paper is organized as follows. In Section II, the preliminaries are introduced. In Section III, new algorithms are proposed for distributed optimization under arbitrary subspace constraints. Section IV presents the analysis results. Section V presents the case study. Section VI concludes the paper.
II. PRELIMINARIES

A. Notation

\(\mathbb{R}^n\) denotes the space of real-valued vectors of dimension \(n\).
\(N\) corresponds to the set of non-negative integers. \(0\) denotes a zero matrix of appropriate dimension. \(I_i\) denotes the \(i \times i\) identity matrix. \(L_N\) is an \(N\)-entry vector of ones. \(X \succ 0\) and \(X \succeq 0\) indicate that a symmetric matrix \(X\) is positive definite and positive semidefinite, respectively. \(\mathbb{R}(M), N(M),\) and \(\text{tr}(M)\) denote the range, nullspace, and trace of matrix \(M\), respectively. \(\otimes\) denotes the Kronecker product. \(\text{col}\{v_k\}_{k=1}^N\) denotes the vertical concatenation of the vectors \(v_1, \ldots, v_N\) and \(\text{diag}\{\lambda_k\}_{k=1}^N\) denotes the diagonal augmentation of the scalars \(\lambda_1, \ldots, \lambda_N\). Given a projection matrix \(P_u\) and a network with gossip matrix \(A\), we define the spectral gap as \(\sigma := ||A-P_u||\), where \(0 \leq \sigma < 1\) and \(||\cdot||\) denotes the spectral norm. \(\|v\|\) denotes the \(\ell_2\)-norm of vector \(v \in \mathbb{R}^n\).

B. Problem Formulation

The optimization problem to be solved is

\[
\min_{\omega} \sum_{k=1}^{N} J_k(\omega_k) \quad \text{subject to} \quad \omega \in \mathbb{R}(U), \tag{1}
\]

where \(U\) is a matrix with full column rank, whose columns form a basis of the subspace constraining \(\omega := \text{col}\{\omega_k\}_{k=1}^N\).

To solve (1) in a distributed way, we consider a network consisting of \(N\) agents, connected over a simple undirected graph \(G\). The set of vertices of \(G\) is defined as \(V = \{1, \ldots, N\}\), where each vertex \(k \in V\) corresponds to an agent. The ordered pair \((i, j)\) is in the edge set \(E\) if and only if there is an edge between vertices \(i \in V\) and \(j \in V\). Each agent \(k\), for \(k \in V\), has access to part of the objective function, i.e. a local objective function \(J_k : \mathbb{R}^d \rightarrow \mathbb{R}\) that is strongly convex and continuously differentiable. The subspace constraint is a coupling constraint. Otherwise, solving (1) would simply require minimizing the local objectives separately.

The local objective functions satisfy the assumptions below.

Assumption 1: The local objective function \(J_k\) of agent \(k\) has an \(L_k\)-Lipschitz continuous gradient \(\nabla J_k\); i.e.,

\[
||\nabla J_k(\omega_a) - \nabla J_k(\omega_b)|| \leq L_k||\omega_a - \omega_b|| \quad \text{for all} \quad \omega_a, \omega_b \in \mathbb{R}^d.
\]

Assumption 2: The local objective function \(J_k\) of agent \(k\) is \(m_k\)-strongly convex; i.e., for all \(\omega_a, \omega_b \in \mathbb{R}^d\)

\[
J_k(\omega_b) \geq J_k(\omega_a) + \nabla J_k(\omega_a)^T(\omega_b - \omega_a) + \frac{m_k}{2}||\omega_b - \omega_a||^2.
\]

The vector \(\xi^t\) is the state value at iteration \(t\), of dimension \(m_{\text{alg}}\), where \(n := Nd\) and \(n_{\text{alg}}\) is algorithm-specific. The input vector \(u^t\) is a stack of the local gradients evaluated at the respective \(y_k^t\) of each agent and \(y^t := \text{col}\{y_k^t\}_{k=1}^N\). The algorithm iterate \(\omega^t\) can be measured from state vector \(\xi^t\) using \(C_\omega\).

The state update is corrupted by zero-mean additive noise \(v^t\). Additive noise can be used to model gradient noise due to numerical errors or approximations. In learning applications, where the true objective function is unknown but approximated through sample collection, gradient noise models the error between the true and estimated objectives. In the absence of gradient noise, \(v^t \equiv 0\), each algorithm is assumed to have a fixed-point \((\xi^*, y^*, u^*, \omega^*)\) satisfying (2). The noise satisfies the following assumption.

Assumption 3: The noise \(v^t\) is zero-mean (i.e. unbiased), with covariance \(\mathbb{E}v^t(v^t)^T \preceq R\) for some \(R \succ 0\), for all \(t \in \mathbb{N}\). The noise sequence has joint distribution \(P\), which is independent across iterations. If \(v \sim P\), then \(v^t\) and \(v^\tau\) are independent for all \(t \neq \tau\).

The dimension \(d\) is the same for all agents and the algorithms to be analyzed have state-space matrices that exhibit a special structure: \(A = \bar{A} \otimes I_d\) for some matrix \(\bar{A}\), \(B = \bar{B} \otimes I_d\) for some matrix \(\bar{B}\), etc. Thus, the following assumption is made without loss of generality.

Assumption 4: The dimension \(d = 1\).

Each algorithm satisfies a respective invariant:

\[
F_t \xi^t + F_\omega u^t = 0 \quad \text{for all} \quad t \in \mathbb{N}. \tag{3}
\]

An invariant follows from the initialization constraint of a given algorithm and is often necessary to produce a feasible solution using the upcoming analysis methods.

D. Algorithm Performance Metrics

The below performance metrics are adopted from [22].

1) Rate of Convergence: The performance metric \(\rho\) is the worst-case linear convergence rate of an algorithm across all objective functions and all initial conditions. This rate describes the transient phase of algorithm iterates, where the noise input \(v^t\) is negligible compared to the gradient input \(u^t\). Thus, for this performance metric, it is assumed there is no noise and a \(\rho\) is computed such that \(||\xi^t - \xi^*|| \leq c \rho^t ||\xi^0 - \xi^*||\) for some constant \(c > 0\), for all \(t \in \mathbb{N}\). Formally, the rate of convergence is defined as

\[
\rho := \inf \left\{ \rho > 0 \mid \sup_{\rho^t} \frac{||\xi^t - \xi^*||}{\rho^t} < \infty \right\}. \tag{4}
\]

2) Sensitivity: The sensitivity \(\gamma\) is a measure of robustness to additive noise. It bounds the standard deviation of iterates produced by the algorithm during the steady-state phase. The quantity \(\gamma^2\) can be interpreted as a bound on the generalized \(H_2\)-norm of the system. Formally, the sensitivity is defined as

\[
\gamma := \lim_{T \to \infty} \sup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} ||\xi^t - \xi^*||^2. \tag{5}
\]
E. IQCs Describing Gradients of Convex Functions

First-order algorithms are viewed as uncertain systems where the nominal system corresponds to linear dynamics (2a)-(2b) and uncertainty operator \( \Delta \in \Delta \) corresponds to the local gradient computations performed in (2c). The IQC characterizing \( \Delta \) can be defined by symmetric matrix \( M \) and \( z \), the output of operator \( \Psi \), driven by \( u \) and \( y \). Specifically, \( \Psi \) is defined as

\[
\begin{align*}
\psi^{t+1} &= A \psi^t + B^0 y^t + B^1 u^t, \quad \psi^t = \psi^\ast, \\
\bar{z}^t &= C \psi^t + D^0 y^t + D^1 u^t,
\end{align*}
\]

(6a)

with a fixed-point defined by \( (\psi^\ast, y^\ast, u^\ast, z^\ast) \).

Suppose \( u = \Delta(y), \ z = \Psi(y, u), \) and \( z^\ast = \Psi(y^\ast, u^\ast) \). The operator \( \Delta \) is said to satisfy the \( (a) \) Pointwise, \( (b) \) \( \rho \)-Hard, \( (c) \) Hard, \( (d) \) Soft IQC defined by \( \Phi, M \) if the respective inequality holds for all vector-valued sequences \( y \) and \( t \in \mathbb{N} \):

\[
(z^t - z^\ast)^T M(z^t - z^\ast) \geq 0 \quad \text{for all } t \in \mathbb{N},
\]

(7a)

\[
\sum_{t=0}^{\infty} \rho^{-2t} (z^t - z^\ast)^T M(z^t - z^\ast) \geq 0,
\]

(7b)

\[
\sum_{t=0}^{\infty} (z^t - z^\ast)^T M(z^t - z^\ast) \geq 0,
\]

(7c)

\[
\sum_{t=0}^{\infty} (z^t - z^\ast)^T M^2(z^t - z^\ast) \geq 0,
\]

(7d)

where the summation in (7d) is finite.

The next IQCs characterize objective functions of interest, where \( \bar{m} := \text{diag}\{m_k\}_{k=1}^N \) and \( \bar{L} := \text{diag}\{L_k\}_{k=1}^N \).

Lemma 1 (Distributed IQC): Let \( J_k \in S(m_k, L_k) \) for all \( k \in V \), \( \phi := \{\nabla J_k(y^1_k)\}_{k=1}^N \) and \( \Phi := (\phi^1, \phi^2, \ldots) \). If \( u = \Phi(y) \), then \( \Phi \) satisfies the pointwise IQC defined by

\[
\Psi = \begin{bmatrix}
\bar{L} & -I_N \\
-m & I_N
\end{bmatrix} \quad \text{and} \quad M = \begin{bmatrix}
0 & I_N \\
I_N & 0
\end{bmatrix}.
\]

Proof: Extension of proof of Lemma 6 in [17].

Lemma 2 (Distributed Weighted Off-By-One IQC): Let \( J_k \in S(m_k, L_k) \) for all \( k \in V \), \( \Phi := \{\nabla J_k(y^1_k)\}_{k=1}^N \) and \( \Phi := (\phi^1, \phi^2, \ldots) \). If \( u = \Phi(y) \), then for any \( (\bar{\rho}, \rho) \) where \( 0 \leq \bar{\rho} \leq \rho \leq 1 \), \( \Phi \) satisfies the \( \rho \)-hard IQC defined by

\[
\Psi = \begin{bmatrix}
0 & -\bar{L} \\
\bar{\rho}^2 I_N & \bar{L} & -I_N \\
-m & \bar{L} & -I_N
\end{bmatrix} \quad \text{and} \quad M = \begin{bmatrix}
0 & I_N \\
I_N & 0 & I_N
\end{bmatrix}.
\]

Proof: Extension of proof of Lemma 10 in [17].

III. GENERALIZATION OF FIRST-ORDER CONSENSUS ALGORITHMS TO ARBITRARY SUBSPACE CONSTRAINTS

The subspace constraint \( \omega \in \mathcal{R}(U) \) in (1) has a corresponding symmetric projection matrix \( P_U = U(U^T U)^{-1} U^T \). The following gradient projection method can solve (1) iteratively:

\[
\omega^{t+1} = P_U \left( \omega^t - \mu \text{col}\{\nabla J_k(\omega^t)_k\}_{k=1}^N \right).
\]

(8)

However, this is not a distributed method due to its reliance on a centralized projection operation. To be a distributed method, the projection operation must be replaced by a diffusion/mixing step using some symmetric gossip matrix \( A \).

The matrix \( A \) must satisfy the sparsity pattern of the network (i.e., \( A_{ij} = 0 \) if \( \{i, j\} \notin E \)) and the convergence condition

\[
\lim_{t \to \infty} A^t = P_U.
\]

(9)

Lemma 3 ([23]): Condition (9) holds if and only if

\[
AP_U = P_U, \quad P_U A = P_U, \quad \|A - P_U\| < 1.
\]

(10)

The consensus algorithms address the problem of finding a common \( \omega \in \mathbb{R}^d \) amongst agents that minimizes \( \sum_{k=1}^N J_k(\omega) \). This problem is a special case of (1) with \( U = 1_N \otimes I_d \).

Distributed consensus algorithms utilize a doubly stochastic gossip matrix \( W \) that satisfies the sparsity pattern of the network. These algorithms’ fixed-points must satisfy the conditions \( \omega^* = W \omega^* \) and \( \omega^* \in \mathcal{R}(1_N) \), which are equivalent to the consensus condition \( \mathcal{N}(I - W) = \mathcal{R}(1_N) \).

Enforcing double stochasticity is simply an application of Lemma 3, where \( P_U = \frac{1}{d} 1_N 1_N^T \). The consensus condition is a special case of the more general condition \( \mathcal{N}(I - A) = \mathcal{R}(U) \), which implies that \( \omega^* = A \omega^* \) and the fixed-point \( \omega^* \) of the algorithm satisfies the subspace constraint \( \omega \in \mathcal{R}(U) \).

The resulting conclusion from the above discussion is that distributed first-order consensus algorithms designed to accelerate DGD or remove its fixed-point bias can be generalized to a larger class of problems by careful adjustment of the gossiping scheme, i.e., by replacing \( W \) with an \( A \) satisfying Lemma 3. Table I lists the generalized versions of select algorithms, presented using the notation in Section II-C. For a given network topology, the choice of \( U \) impacts the performance of these algorithms, which can be shown by applying the upcoming analysis results.

There will not always exist an \( A \) that satisfies condition (9) for some given network sparsity pattern. In the consensus case, the graph must simply be connected to guarantee existence. Equation (66) in [13] defines a convex optimization problem based on the \( l_1 \)-norm sparsity heuristic that takes an arbitrary sparsity pattern and generates an \( A \) that satisfies (10) with minimal edges added to the original network topology.

Remark 1: The generalized distributed algorithms from Table I can be used to solve regularized problems of the form \( \min_{\omega} \sum_{k=1}^N J_k(\omega_k) + \frac{\gamma}{2} \omega^T P_U \omega \) for some regularization term \( P_U \succeq 0 \) with regularization strength \( \gamma > 0 \). \( P \) can be decomposed non-uniquely in terms of an orthonormal matrix \( V \) and a diagonal matrix \( \Lambda \) as in \( P = V \Lambda V^T \). The regularized problem can be rewritten as a minimization over \( \omega \in \mathbb{R}^n \) and \( y \in \mathbb{R}^n \) subject to a linear (i.e., subspace) constraint:

\[
\min_{\omega, y} \sum_{k=1}^N J_k(\omega_k) + \frac{\gamma}{2} y^T \Lambda y \quad \text{subject to} \quad y = V^T \omega.
\]

The diagonal structure of \( \Lambda \) makes the new objective function separable and therefore suitable for a distributed algorithm. Although the choice of the decomposition of \( P \) can affect the algorithm performance for a given network topology, it was found that the spectral gap \( \sigma \) serves as a good heuristic for selecting the best decomposition; the decomposition that can result in a gossip matrix \( A \) from \( P_U \) with the smallest \( \sigma \) will generally yield the fastest convergence.

IV. ANALYSIS RESULTS

The dynamics of the algorithms from (2) and of \( \Psi \) from (6) can be used to define the following extended system \( \tilde{G} \):

\[
\xi^{t+1} = \tilde{A} \xi^t + \tilde{B} u^t + \begin{bmatrix} F \\ 0 \end{bmatrix} v^t, \quad \xi^t = \tilde{C} \xi^t + \tilde{D} u^t.
\]

(11)
where \( \hat{\xi}_t = \left[ \xi_t^T \right] \), \( A = \left[ \begin{array}{cc} A_B & C_y \\ B_y & C_y \end{array} \right] \), \( B = \left[ \begin{array}{c} B_f \\ B_y \end{array} \right] \), \( C = \left[ \begin{array}{c} D_y \end{array} \right] \), and \( \hat{D} = \hat{D}_y \). In the absence of gradient noise, the extended system has fixed-point \((\xi^*, \hat{\xi}^*, \hat{u}^*, z^*)\).

**Theorem 1 (Distributed Algorithm Rate of Convergence):** Consider solving problem (1) for a set of local objective functions \( J_k \in S(m_k, L_k) \) for all \( k \in \mathcal{V} \). The uncertainty operator satisfies the \( \rho \)-hard IQC defined by \((\Psi, M)\) for a given \( \rho > 0 \). Assume the noise \( v_t = 0 \). Also, assume the algorithm has a unique fixed-point and satisfies the invariant condition (3). Let \( H \) be a matrix whose columns form a basis for the nullspace of \([F_k \ 0 \ F_u]\).

If there exist \( P \succeq 0 \) and \( \lambda > 0 \) such that

\[
H^T \left( \begin{array}{c} \hat{A}_T P \hat{A} - \rho^2 P \\ \hat{B}_T P \hat{A} \\ \hat{B}_T P B \end{array} \right) + \lambda \left[ C \hat{D} \left( \begin{array}{c} \hat{C} \hat{D} \end{array} \right) M \left[ \begin{array}{c} \hat{C} \hat{D} \end{array} \right] H \right] \leq 0 ,
\]

then \( \| \xi_t - \xi^* \| \leq c \rho^t \| \xi_0^* - \xi^* \| \) for some \( c > 0 \) and all \( t \in \mathbb{N} \).

**Proof:** Define the error states \( \hat{\xi}_t := \xi_t^T - \xi^*, \hat{u}_t := u_t^T - u^*, \) and \( \hat{z}_t := z_t^T - z^* \). The columns of \( H \) span the nullspace of \([F_k \ 0 \ F_u]\), so any vector \([\hat{\xi}_t \ 0 \ \hat{u}_t \ \hat{z}_t]^T\) is of the form \( H \hat{h}_t \) for some \( h_t \). Pre- and post-multiply (12) by \((h_t)^T\) and \( h_t \), respectively, to obtain

\[
(\hat{\xi}_t + 1)^T P(\hat{\xi}_t + 1) - \rho^2 (\hat{\xi}_t)^T P(\hat{\xi}_t) \leq 0 ,
\]

Multiply by \( \rho^{-2t} \) and sum the resulting inequalities from 0 to \( T - 1 \) for any \( T \in \mathbb{N} \). The first two terms produce a telescoping series such that

\[
\rho^{-2T+2}(\hat{\xi}_t)^T P(\hat{\xi}_t) - \rho^2 (\hat{\xi}_t)^T P(\hat{\xi}_t) + \lambda \sum_{i=0}^{T-1} \rho^{-2i}(\hat{z}_t)^T M \hat{z}_t \leq 0 .
\]

The summation is non-negative because the uncertainty satisfies a \( \rho \)-hard IQC. As a result, \((\hat{\xi}_t)^T P(\hat{\xi}_t) \leq \rho^2 (\hat{\xi}_0)^T P(\hat{\xi}_0)\), which implies \( \lambda_{\min}(P)\|\xi_t^0\|^2 \leq \rho^2 \lambda_{\max}(P)\|\xi_0^*\|^2 \), where \( \lambda_{\min}(P) \) and \( \lambda_{\max}(P) \) are the minimum and maximum eigenvalues of \( P \), respectively. Rearrange and use \( \psi^0 = \psi^* \) to produce the result: \( \| \hat{\xi}_t - \xi^* \| \leq \sqrt{\lambda_{\max}(P)/\lambda_{\min}(P)} \rho^t \| \xi_0^* - \xi^* \| .
\]

Multiple IQCs can be used simultaneously. For \( r \) IQCs, the output \( z_t \) in (6b) becomes \( \text{col} \{ z_t^k \}_{k=1}^r \). The \( \lambda \) term in (12) is replaced by the block-diagonal matrix formed from \( \lambda M_i \) for \( i = 1, \ldots, r \), namely, \( \text{blkdiag} \{ \lambda_1 M_1, \ldots, \lambda_r M_r \} \).

**Theorem 2 (Distributed Algorithm Sensitivity):** Consider solving problem (1) for a set of local objective functions \( J_k \in S(m_k, L_k) \) for all \( k \in \mathcal{V} \). The uncertainty operator satisfies the \( \rho \)-hard IQC defined by \((\Psi, M)\) for a given \( \rho > 0 \). Assume the algorithm is subject to zero mean additive gradient noise satisfying Assumption 3. Also, assume the algorithm has a unique fixed-point and satisfies the invariant condition (3). Let \( H \) be a matrix whose columns form a basis for the nullspace of \([F_k \ 0 \ F_u]\).

If there exist \( P \succeq 0 \) and \( \lambda > 0 \) such that

\[
H^T \left( \begin{array}{c} \hat{A}_T P \hat{A} - \rho^2 P \\ \hat{B}_T P \hat{A} \\ \hat{B}_T P B \end{array} \right) + \lambda \left[ C \hat{D} \left( \begin{array}{c} \hat{C} \hat{D} \end{array} \right) M \left[ \begin{array}{c} \hat{C} \hat{D} \end{array} \right] H \right] \leq 0 ,
\]

then \( \gamma \leq \sqrt{\text{tr}(RB^T P M B)} \).

**Proof:** Define the error states \( \tilde{\xi}_t := \xi_t^T - \xi^*, \tilde{u}_t := u_t^T - u^*, \) \( \tilde{\omega}_t := \omega_t^T - \omega^* \) and \( \tilde{z}_t := z_t^T - z^* \). The columns of \( H \) span the nullspace of \([F_k \ 0 \ F_u]\), so any vector \([\tilde{\xi}_t \ 0 \ \tilde{u}_t \ \tilde{\omega}_t \ \tilde{z}_t]^T\) is of the form \( H \tilde{h}_t \) for some \( h_t \). Pre- and post-multiply (13) by \((h_t)^T\) and \( h_t \), respectively, to obtain

\[
(\hat{\xi}_t + 1)^T P(\hat{\xi}_t + 1) - (\hat{\xi}_t)^T P(\hat{\xi}_t) - 2(\hat{\xi}_t + 1)^T P(\hat{\xi}_t) + (\tilde{\omega}_t)^T P(\hat{\xi}_t) + \lambda(\tilde{z}_t)^T M(\tilde{z}_t + \|\tilde{\omega}_t\|)^2 \leq 0 .
\]
For the third term, substitute $\hat{\xi}^t$ using (11) to obtain
\[(\hat{\xi}^t)^T P(\hat{\xi}^t) - (\hat{\xi}^t)^T P \hat{\xi}^t - 2(\hat{A}\hat{\xi}^t + \hat{B}\hat{u}^t)^T P[\hat{\xi}^t]\]
\[= (v^t)^T [\hat{B}^T P \hat{B}] v^t + \lambda (\hat{z}^t)^T M \hat{z}^t + \|\hat{w}^t\|^2 \leq 0.\]
Take the expectation. The third term is zero because $\hat{v}^t$ is zero-mean and $\hat{\xi}^t$ is independent of $v^t$. Rearrange to obtain
\[E(\hat{\xi}^t)^T P(\hat{\xi}^t) - E(\hat{\xi}^t)^T P \hat{\xi}^t + E\lambda(\hat{z}^t)^T M \hat{z}^t + E\|\hat{w}^t\|^2 \leq 0.\]

The SDPs are implemented in MATLAB with the CVX modeling language and the MOSEK solver [25].

V. CASE STUDY

Consider a 4-agent fully-connected network with one edge removed. Agents have nonlinear, strongly convex local objective functions of the form $J_k(\omega_k) = a_k(\omega_k - b_k)^2 - \cos(\omega_k)$, where \(a_1, a_2, a_3, a_4 = \{3, 7, 2, 4\}\) and \(b_1, b_2, b_3, b_4 = \{-2, -1, 5, 12\}\). Distributed optimization is to be performed with a subspace constraint defined by $U = [\frac{1}{2}, \frac{3}{2}, \frac{5}{2}]^T$. We introduce gradient noise, where the noise bound $R$ is 0.251. Figure 1 shows simulation results for the algorithms defined in Table I, using a gossip matrix $A_1$ that satisfies (10) and $(2, 3) \notin E$, with step-size $\mu = 0.012$. Despite the fact that $\omega^{opt} = [-0.719, 3.996, 3.277, 7.991]^T$ (computed numerically using CVX [24]) is not a consensus solution, all generalized algorithms converge towards $\omega^{opt}$ without bias. DAS and DiSPO converge with fixed-point biases.

![Fig. 1. Norm of error with respect to $\omega^{opt}$](image)

For the same parameters, the above analysis tools are used to show how an algorithm from Table I will perform when solving the multitask inference problem found in [11], [13]. Here, the optimal parameter must lie in a low-dimensional subspace and the local objective functions can be considered to be the expectation of some loss function $Q(\omega_k; x_k)$. The random variable $x_k$ corresponds to data received by agent $k$, whose distribution is unknown. Since only a finite number of samples $x_k$ are received by each agent, their local gradient computations are subject to gradient noise. The local costs are assumed to be strongly convex with bounded Hessian and so Assumptions 1 and 2 are satisfied. In [11], DAS is proposed to solve this type of problem.

Instead of using the local objective functions above, worst-case analysis is performed over all objective functions in $S(m_k, L_k)$ for all $k \in V$. Another gossip matrix $A_2$ is considered, where $(1, 4) \notin E$. $A_1$ and $A_2$ have spectral gaps of $\sigma = 0.19$ and $\sigma = 0.63$, with respect to $P_{it}$. Figure 2 shows the trade-off between sensitivity $\gamma$ and convergence rate $\rho$ as step-size $\mu$ varies for the DAS algorithm and the generalized AugDGM algorithm [7]. These numerical results are obtained by implementing Theorems 1 and 2, using both IQCs defined in Lemmas 1 and 2. The SDPs are implemented in MATLAB with the CVX modeling language and the MOSEK solver [25].
In general, an algorithm will exhibit faster convergence (smaller $\rho$) as $\mu$ increases, at the cost of worse robustness against additive gradient noise (higher $\gamma$). At an algorithm-specific limiting $\mu$, which is dictated by $L := \max \{L_k\}_{k=1}^m$, $m := \min \{m_k\}_{k=1}^N$, and $\sigma$, any further increase of $\mu$ will be detrimental to both convergence rate and robustness. This phenomenon is shown in the AugDGM curve for $\sigma = 0.63$, where worst-case convergence rate cannot improve beyond about 0.8 (corresponding to $\mu = 0.05$).

For the $\sigma = 0.19$ case, the AugDGM algorithm performs strictly better than DAS over the prescribed step-size range, with slightly lower $\gamma$ for each given $\rho$. Results for the $\sigma = 0.63$ case are similar up to the limiting step-size of AugDGM. If prioritizing convergence rate, it would appear that DAS has greater potential since it can achieve convergence rates between 0.5 and 0.6. However, both DAS and DiSPO are subject to fixed-point biases, which is not captured in this analysis, since $\gamma$ is relative to the algorithm’s fixed-point. In this large step-size regime, the bias introduced by DAS is large enough to prohibit the use of the algorithm. For example, for the defined objective functions and gossip matrix $A_1$, the bias $\|\omega_{\text{opt}} - \omega^*\|$ is 0.7 at $\mu = 0.05$, increasing to 1.5 at $\mu = 0.12$. Both are an order of magnitude larger than $\gamma$. For $A_2$, the bias is even higher, at 3.1 and 3.8 for $\mu = 0.05$ and $\mu = 0.12$, respectively. In summary, the generalized AugDGM algorithm converges to the correct solution and is more robust compared to the DAS algorithm, with a stronger benefit when the network is well-connected (low $\sigma$).

VI. CONCLUSION

In this paper, we show that consensus algorithms that remove the fixed-point bias of DGD can be extended to solve the more general problem of distributed optimization over subspace constraints. We provide a framework that can analyze the performance of these generalized algorithms in terms of worst-case robustness and convergence rate. Our framework can certify (or improve) the convergence rates provided by algorithm designers, as well as provide new robustness guarantees for algorithms that have not been previously considered in the stochastic setting. Finally, we demonstrate the utility of our framework by showing how a generalized consensus algorithm can be applied to a multitask inference problem.

REFERENCES

[1] K. Yuan, Q. Ling, and W. Yin, “On the Convergence of Decentralized Gradient Descent,” SIOPT, vol. 26, no. 3, pp. 1835–1854, 2016.

[2] W. Shi, Q. Ling, G. Wu, and W. Yin, “EXTRA: An Exact First-Order Algorithm for Decentralized Consensus Optimization,” SIOPT, vol. 25, no. 2, pp. 944–966, 2015.

[3] Z. Li, W. Shi, and M. Yan, “A Decentralized Proximal-Gradient Method With Network Independent Step-Sizes and Separated Convergence Rates,” IEEE Trans. Signal Process., vol. 67, pp. 4494–4506, 2019.

[4] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, “Exact Diffusion for Distributed Optimization and Learning Part I: Algorithm Development,” IEEE Trans. on Signal Processing, vol. 67, no. 3, pp. 708–723, 2019.

[5] A. Nedić, A. Olshevsky, and W. Shi, “Achieving Geometric Convergence for Distributed Optimization Over Time-Varying Graphs,” SIOPT, vol. 27, no. 4, pp. 2597–2633, 2017.

[6] D. Jakovetic, “A Unification and Generalization of Exact Distributed First-Order Methods,” IEEE Trans. on Signal and Information Processing over Networks, vol. 5, no. 1, pp. 31–46, 2019.

[7] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Augmented Distributed Gradient Methods for Multi-Agent Optimization Under Uncoordinated Constant Stepsizes,” in CDC, vol. 54, pp. 2055–2060, IEEE, 2015.

[8] A. Sundararajan, B. Van Scyoc, and L. Lessard, “Analysis and Design of First-Order Distributed Optimization Algorithms Optimizing 2nd-Order Graphs,” IEEE Trans. on Control of Network Systems, vol. 7, no. 4, pp. 1597–1608, 2020.

[9] G. Qu and N. Li, “Accelerated Distributed Nesterov Gradient Descent,” IEEE Trans. on Autom. Control, vol. 65, no. 6, pp. 2566–2581, 2020.

[10] K. Yuan, S. A. Alghunaim, B. Ying, and A. H. Sayed, “On the Influence of Bias-Correction on Distributed Stochastic Optimization,” IEEE Trans. on Signal Processing, vol. 68, pp. 4352–4367, 2020.

[11] R. Nissaf, S. Vlaski, and A. H. Sayed, “Adaptation and Learning Over Networks Under Subspace Constraints Part I: Stability Analysis,” IEEE Trans. on Signal Processing, vol. 68, pp. 1346–1360, 2020.

[12] P. D. Lorenzo, S. Barbarossa, and S. Sardellitti, “Distributed Signal Recovery Based on In-Network Subspace Projections,” in ICASSP, pp. 5242–5246, IEEE, 2019.

[13] R. Nissaf, S. Vlaski, and A. H. Sayed, “Adaptation and Learning Over Networks Under Subspace Constraints Part II: Performance Analysis,” IEEE Trans. on Signal Processing, vol. 68, no. 1, pp. 1–1, 2020.

[14] A. Megretskii and A. Rantzer, “System Analysis via Integral Quadratic Constraints,” IEEE Trans. on Autom. Control, vol. 42, no. 6, pp. 819–830, 1997.

[15] D. Abou Jouade, D. Muniraj, and M. Farhood, “Robustness Analysis of Uncertain Time-Varying Interconnected Systems Using Integral Quadratic Constraints,” in American Control Conference, IEEE, 2019.

[16] J. M. Fry, D. Abou Jouade, and M. Farhood, “Robustness analysis of uncertain time-varying systems using integral quadratic constraints with time-varying multipliers,” International Journal of Robust and Nonlinear Control, vol. 31, no. 3, pp. 733–758, 2021.

[17] L. Lessard, B. Recht, and A. Packard, “Analysis and Design of Optimization Algorithms via Integral Quadratic Constraints,” SIOPT, vol. 26, no. 1, pp. 57–95, 2016.

[18] B. Van Scyoc, R. A. Freeman, and K. M. Lynch, “The Fastest Known Globally Convergent First-Order Method for Minimizing Strongly Convex Functions,” IEEE Control Syst. Lett., vol. 2, pp. 49–54, 2018.

[19] S. Cyrus, B. Hu, B. Van Scyoc, and L. Lessard, “A Robust Accelerated Optimization Algorithm for Strongly Convex Functions,” in American Control Conference, pp. 1376–1381, IEEE, 2018.

[20] A. Sundararajan, B. Hu, and L. Lessard, “Robust Convergence Analysis of Distributed Proximal Optimization Algorithms,” in Allerton Conference on Communication, Control, & Computing, pp. 1206–1212, 2017.

[21] S. Michalowsky, C. Scherer, and C. Ebenbauer, “Robust and Structure Exploiting Optimisation Algorithms: an IQC Approach,” International Journal of Control, vol. 94, no. 11, pp. 2956–2979, 2021.

[22] B. Van Scyoc and L. Lessard, “The Speed-Robustness Trade-Off for First-Order Methods with Additive Gradient Noise,” 2021, arXiv:2109.05059v1.

[23] P. Di Lorenzo, S. Barbarossa, and S. Sardellitti, “Distributed Signal Processing and Optimization Based on In-Network Subspace Projections,” IEEE Trans. Signal Process., vol. 68, pp. 2061–2076, 2020.

[24] M. C. Grant and S. P. Boyd, “CVX: Matlab Software for Disciplined Convex Programming,” 2012.

[25] M. ApS, “The MOSEK Optimization Toolbox for MATLAB,” 2019.