On the Convergence of a Distributed Augmented Lagrangian Method for Non-Convex Optimization

Nikolaos Chatzipanagiotis, Student Member, IEEE, and Michael M. Zavlanos, Member, IEEE

Abstract—In this paper we propose a distributed algorithm for optimization problems that involve a separable, possibly non-convex objective function subject to convex local constraints and linear coupling constraints. The method is based on the Accelerated Distributed Augmented Lagrangians (ADAL) algorithm that was recently developed by the authors to address convex problems. Here, we extend this line of work in two ways. First, we establish convergence of the method to a local minimum of the problem, using assumptions that are common in the analysis of non-convex optimization methods. To the best of our knowledge this is the first work that shows convergence to local minima specifically for a distributed augmented Lagrangian (AL) method applied to non-convex optimization problems; distributed AL methods are known to perform very well when used to solve convex problems. Second, we propose a more general and decentralized rule to select the stepsizes of the method. This improves on the authors’ original ADAL method, where the stepsize selection used global information at initialization. Numerical results are included to verify the correctness and efficiency of the proposed distributed method.

Index Terms—Distributed optimization, non-convex optimization, augmented Lagrangian.

I. INTRODUCTION

Many applications in areas as diverse as wireless communications, machine learning, artificial intelligence, power systems, computational biology, logistics, finance and statistics involve very large datasets that are obtained, stored, and retrieved in a decentralized manner. Within these areas, a significant number of problems involving, e.g., cellular phone networks, sensor networks, multi-agent robotics, power grids, and the Internet, also possess a network structure wherein processors, sensors, actuators, and controllers need to cooperate in a distributed fashion over geographically disparate locations, based only on local information and communication. The increasing size and complexity, and the local nature of information that is particular to these problems has created a need for efficient distributed computation methods.

In this paper we are particularly interested in distributed optimization algorithms. Such methods have been used recently to address a wide range of modern day problems involving wired and wireless communication networks [1]–[3], multi-agent robotic networks [4, 5], machine learning [6], power distribution systems [7], image processing [8], model predictive control [9], statistics [10], and logistics [11].

We propose a distributed algorithm to solve the following class of constrained optimization problems

\[
\begin{aligned}
&\min_{x_i} \sum_{i=1}^{N} f_i(x_i) \\
\text{subject to} &\sum_{i=1}^{N} A_i x_i = b,
\end{aligned}
\]

where, for every \(i \in \mathcal{I} = \{1, 2, \ldots, N\}\), the function \(f_i : \mathbb{R}^{n_i} \to \mathbb{R}\) is twice continuously differentiable, \(\mathcal{X}_i \subseteq \mathbb{R}^{n_i}\) denotes a nonempty closed, convex subset of \(n_i\)-dimensional Euclidean space, and \(A_i\) is a matrix of dimension \(m \times n_i\).

Problem (1) models situations where a set of \(N\) decision makers, henceforth referred to as agents, need to determine local decisions \(x_i \in \mathcal{X}_i\) that minimize a collection of local cost functions \(f_i(x_i)\), while respecting a set of affine constraints \(\sum_{i=1}^{N} A_i x_i = b\) that couple the local decisions between agents. In previous work [12, 13], we presented the Accelerated Distributed Augmented Lagrangians (ADAL) method to solve such problems in a distributed fashion, when the objective functions \(f_i\) are convex but not necessarily differentiable. ADAL is a primal-dual iterative scheme based on the augmented Lagrangian (AL) framework [14, 15]. In ADAL, every agent is assumed to know its local problem parameters \(f_i, A_i, \mathcal{X}_i\), and is also responsible for determining its own decision variables \(x_i\). Each iteration of ADAL consists of three steps. First, every agent solves a local convex optimization problem based on a separable approximation of the AL, that utilizes only locally available variables. Then, the agents update and communicate their primal variables to neighboring agents. Here, the communication neighbors of agent \(i\) are all those agents \(j\) that are coupled in the same constraints as \(i\), i.e., the communication requirements between the agents are determined by the structure of the (static) coupling constraint set \(\sum_{i=1}^{N} A_i x_i = b\). Finally, in the last step the dual variables are updated in a distributed fashion based on the new values of the primal variables; the Lagrange multiplier of the \(j\)-th constraint is updated based on communicated information from those agents whose decisions are coupled in this constraint, i.e., those \(i\) for which \([A_i]_{j} \neq 0\). The computations at each step are performed in parallel. It was shown in [16] that ADAL has a worst-case \(O(1/k)\) convergence rate, where \(k\) denotes the number of iterations. Moreover, a stochastic convergence framework for ADAL was established in [13] for convex constrained optimization problems that are subject to noise corruption and uncertainties.
In this paper we extend ADAL in two ways. First, under assumptions that are common in the study of non-convex optimization methods, we prove the convergence of ADAL to a local minimum of problem (1) when the local cost functions \( f_i \) are non-convex. To the best of our knowledge, this is the first published work that formally establishes the convergence of a distributed augmented Lagrangian method for non-convex optimization problems. Second, we propose a way to select the stepsizes used in the algorithm that is more general compared to [12]. Specifically, it was shown in [12] that ADAL converges to the optimal solution of (1) if the stepsizes satisfy a certain condition that requires knowledge of the global structure of the constraint set at initialization. Here, we lift this requirement for global information and, instead, define \( m \) stepsizes associated with each one of the \( m \) coupling constraints in (1); each stepsize must adhere to a condition that requires only local information from the corresponding constraint. It is worth noting that these two contributions are independent from each other, meaning that convergence of the non-convex ADAL method can still be shown using the stepsizes from [12], and, similarly, convergence of the convex ADAL method can be shown using the stepsizes proposed in this paper.

A. Related Literature

The existing literature on distributed optimization methods mostly focuses on convex problems. The classic approach is that of dual decomposition and is based on Lagrangian duality theory [15, 17, 18]. Dual methods are simple and popular, however, they suffer from exceedingly slow convergence rates and require strict convexity of the objective function.

The main drawbacks of simple dual decomposition methods are alleviated by utilizing the augmented Lagrangian (AL) framework, which has recently received considerable attention as a most efficient approach for distributed optimization in deterministic settings; see, e.g., [6, 12, 19]–[23]. The ADAL method [12] considered in this paper belongs in this class of distributed AL algorithms, along with the Alternating Directions Method of Multipliers (ADMM) [6], and the Diagonal Quadratic Approximation (DQA) [20] methods. A distributed AL algorithm similar to ADAL that solves deterministic convex problems of the form (1) has been proposed in [22]. The main difference between [22] and [12] lies in the stepsize choice; in [22] the stepsize is determined by the total number of agents in the problem, while in [12] the stepsize is determined by the number of agents coupled in the “most populated” constraint, which naturally leads to larger stepsizes in most cases. Another pertinent method can be found in [23] that also incorporates Bregman divergence factors into the local subproblems and at each iteration only a randomly selected subset of the agents perform updates. Finally, in [24] a similar algorithm to ADAL is proposed, which has a different dual update step, and also uses the additional assumptions that the matrices \( A_i \) are mutually near-orthogonal and have full column-rank.

Apart from AL methods, alternative algorithms for distributed convex optimization include Newton methods [25]–[27], projection-based approaches [28, 29], accelerated-gradient algorithms [30]–[32], online methods [33, 34], primal-dual perturbation approaches [35], reduced-communication algorithms [36], and even continuous-time approaches [37]. On the other hand, there exist only a few works on non-convex distributed optimization methods, such as Parallel Variable Distribution schemes [38]–[40], Successive Convex Approximation algorithms [41], dual subgradient approaches [42], and Fast-Lipschitz methods [43].

In relevant literature regarding distributed AL methods for non-convex problems, it has been observed that the ADMM can converge in scenarios with non-convex objective functions; see [44]–[47] for some examples. Nevertheless, the only published work that provides some theoretical justification for such observations is found in [48]. There, the authors propose a distributed AL method that provably converges to a stationary point of the non-convex problem, for a certain class of problems and for sufficiently large values of the regularization parameter. The differences between [48] and the current paper are that, for the class of problems considered here, [48] proposes a different algorithm where the agents perform computations sequentially at each iteration, while in our method the computations are performed in parallel. Moreover, the authors of [48] prove convergence to a stationary point of the non-convex problem provided that the regularization parameter is chosen large enough, while in this paper we prove convergence to a local minimum, under the additional assumption that the initialization point is sufficiently close to a locally optimal solution. Other relevant work includes [49] where the authors provide conditions under which certain distributed AL schemes for non-convex problems are guaranteed to converge, and also [50] where an elaborate distributed AL algorithm with modified gradients and Hessian approximations is proposed, similar in spirit to sequential quadratic programming methods.

The rest of this paper is organized as follows. In section II we discuss some essential facts regarding duality and the augmented Lagrangian framework. We also provide a description of the ADAL method that utilizes the new local stepsizes, and discuss how it compares to the method using the global stepsizes presented in [12]. In section III we analyze the convergence of ADAL for problems of the form (1) under the local stepsize selection rule. Finally, Section IV contains numerical results that validate the effectiveness and efficiency of the proposed algorithm.

II. Preliminaries

We denote
\[
F(x) = \sum_{i=1}^{N} f_i(x_i)
\]
where \( x = [x_1^T, \ldots, x_N^T]^T \in \mathbb{R}^n \) with \( n = \sum_{i=1}^{N} n_i \). Furthermore, we denote \( A = [A_1 \ldots A_N] \in \mathbb{R}^{m \times n} \). The constraint \( \sum_{i=1}^{N} A_i x_i = b \) of problem (1) takes on the form \( Ax = b \). Associating Lagrange multipliers \( \lambda \in \mathbb{R}^m \) with that
Algorithm 1 Augmented Lagrangian Method (ALM)
Set $k = 1$ and define initial Lagrange multipliers $\lambda^1$.
1. For a fixed vector $\lambda^k$, calculate $x^k$ as a solution of the problem:
   \[
   \min_{x \in \mathcal{X}} \Lambda_\rho(x, \lambda^k). \tag{4}
   \]
2. If the constraints $\sum_{i=1}^N A_i x_i^k = b$ are satisfied, then stop (optimal solution found). Otherwise, set:
   \[
   \lambda^{k+1} = \lambda^k + \rho \left( \sum_{i=1}^N A_i x_i^k - b \right), \tag{5}
   \]
   Increase $k$ by one and return to Step 1.

constraint, the Lagrange function is defined as
\[
L(x, \lambda) = F(x) + \langle \lambda, Ax - b \rangle \tag{2}
\]
where $L_i(x_i, \lambda) = f_i(x_i) + \langle \lambda, A_i x_i \rangle$, and $\langle \cdot, \cdot \rangle$ denotes inner product. Then, the dual function is defined as
\[
g(\lambda) = \inf_{x \in \mathcal{X}} L(x, \lambda) = \sum_{i=1}^N g_i(\lambda) - \langle b, \lambda \rangle,
\]
where $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \cdots \times \mathcal{X}_N$, and
\[
g_i(\lambda) = \inf_{x_i \in \mathcal{X}_i} \left[ f_i(x_i) + \langle \lambda, A_i x_i \rangle \right].
\]
The dual function is decomposable and this gives rise to various decomposition methods addressing the dual problem, which is defined by
\[
\max_{\lambda \in \mathbb{R}^n} \sum_{i=1}^N g_i(\lambda) - \langle b, \lambda \rangle. \tag{3}
\]

Dual methods suffer from well-documented disadvantages, the most notable ones being their exceedingly slow convergence rates and the requirement for strictly convex objective functions. These drawbacks can be alleviated by the augmented Lagrangian framework \cite{14, 15}. The augmented Lagrangian associated with problem (1) is given by
\[
\Lambda_\rho(x, \lambda) = F(x) + \langle \lambda, Ax - b \rangle + \frac{\rho}{2} \|Ax - b\|^2,
\]
where $\rho > 0$ is a penalty parameter. We recall the standard augmented Lagrangian method (ALM), also referred to as the “Method of Multipliers” in the literature \cite{14, 15}, in Alg. 1.

The convergence of the augmented Lagrangian method is ensured when problem (3) has an optimal solution independently of the initialization. Under convexity assumptions and a constraint qualification condition, every accumulation point of the sequence $\{x^k\}$ is an optimal solution of problem (1), cf. \cite{14}. Furthermore, the augmented Lagrangian method exhibits convergence properties also in a non-convex setting, assuming that the functions $f_i, i = 1, \ldots, N$ are twice continuously differentiable and the strong second-order conditions of optimality are satisfied \cite{14}. This fact combined with the known efficiency of distributed AL methods in convex settings provide a strong motivation to develop distributed non-convex AL schemes, such as the one proposed here.

A. The ADAL algorithm
The ADAL method is based on defining the local augmented Lagrangian function $\Lambda^i : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ for every agent $i \in \mathcal{I} = \{1, \ldots, N\}$ at each iteration $k$, according to
\[
\Lambda^i(x_i, Ax^k, \lambda^k) = f_i(x_i) + \langle \lambda^k, A_i x_i \rangle \tag{6}
\]
\[
+ \frac{\rho}{2} \|A_i x_i + \sum_{j \neq i}^N A_j x_j^k - b\|^2,
\]
where $\rho > 0$ is a scalar penalty parameter defined by the user. Each iteration of ADAL is comprised of three steps: i) a minimization step of all the local augmented Lagrangians with respect to the primal variables, ii) an update step for the primal variables, and iii) an update step for the dual variables. The computations at each step are performed in a parallel fashion, so that ADAL resembles a Jacobi-type algorithm; see \cite{15} for more details on Jacobi and Gauss-Seidel type algorithms. The ADAL method is summarized in Alg. 2.

At the first step of each iteration, each agent minimizes its local AL subject to its local convex constraints. This computation step requires only local information. To see this, note that the variables $A_i x_i^k$, appearing in the penalty term of the local AL (6), correspond to the local primal variables of agent $j$ that were communicated to agent $i$ for the optimization of its local Lagrangian $\Lambda^i$. With respect to agent $i$, these are considered fixed parameters. The penalty term of each $\Lambda^i$ can be equivalently expressed as
\[
\|A_i x_i + \sum_{j \neq i}^N A_j x_j^k - b\|^2
\]
\[
= \sum_{t=1}^m \left( [A_i x_i]_t + \sum_{j \neq i}^N [A_j x_j^k]_t - b_t \right)^2,
\]
where $[A_i x_i]_t$ denotes the $t$-th entry of the vector $A_i x_i$. The above penalty term is involved only in the minimization computation (7). Hence, for those $l$ such that $[A_i]_l = 0$, the terms
needs access only to those constraints \( I \) determined. To see why (8) requires only local information, note that every agent \( i \) needs to know only the \( q_j \)'s that correspond to the constraints that this agent is involved in. Analogous arguments hold for the dual update step (9), also.

### III. Convergence of ADAL

In order to prove convergence of ADAL to a local minimum of (1), we need the following four assumptions:

(A1) The sets \( \mathcal{X}_i \subseteq \mathbb{R}^m, i = 1, \ldots, N \) are nonempty, closed and convex.

(A2) The functions \( f_i : \mathbb{R}^m \to \mathbb{R}, i \in \mathcal{I} = \{ 1, 2, \ldots, N \} \) are twice continuously differentiable on \( \mathcal{X}_i \).

(A3) The subproblems (7) are solvable.

(A4) There exists a point \( x^* \) satisfying the strong second order sufficient conditions of optimality for problem (1) with Lagrange multipliers \( \lambda^* \).

The assumptions (A1), (A2), and (A4) are common and are used in the convergence proof of the standard augmented Lagrangian method (ALM) for non-convex optimization problems, cf. [14]. Assumption (A4) implies that there exist Lagrange multipliers \( \lambda^* \in \mathbb{R}^m \) that satisfy the first order optimality conditions for problem (1) at the feasible point \( x^* \), provided that a constraint qualification condition is satisfied at \( x^* \), i.e.,

\[
\nabla F(x^*) + A^T \lambda^* \in \mathcal{N}_X(x^*),
\]

where we recall that \( x = [x_1^T, \ldots, x_N^T]^T \in \mathbb{R}^n, F(x) = \sum_{i} f_i(x_i), \) and \( A = [A_1 \ldots A_N] \in \mathbb{R}^{m \times n} \). Here, we use \( \mathcal{N}_X(x) \) to denote the normal cone to the set \( X \) at point \( x \) [14], i.e.,

\[
\mathcal{N}_X(x) = \{ h \in \mathbb{R}^n : \langle h, y - x \rangle \leq 0, \quad \forall y \in X \}.
\]

The strong second order sufficient conditions of optimality for problem (1) at a point \( x^* \) imply that

\[
\langle s, \nabla^2 F(x^*) s \rangle > 0, \quad \text{for all } s \neq 0, \quad \text{such that } As = 0,
\]

c.f. [14], Lemma 4.32.

Assumption (A3) is satisfied if for every \( i = 1, \ldots, N \), either the set \( \mathcal{X}_i \) is compact, or the function \( f_i(x_i) + \frac{\rho}{2} \| A_i x_i - c \|_2^2 \) is inf-compact for any vector \( c \). The latter condition, means that the level sets of the function are compact sets, implying that the set \( \{ x_i \in \mathcal{X}_i : f_i(x_i) + \frac{\rho}{2} \| A_i x_i - c \|_2^2 \leq \alpha \} \) is compact for any \( \alpha \in \mathbb{R} \).

Define the residual \( r(x) \in \mathbb{R}^m \) as the vector containing the amount of all constraint violations with respect to primal variable \( x \), i.e.,

\[
r(x) = \sum_{i=1}^{N} A_i x_i - b.
\]
Define also the auxiliary variables
\[ \hat{\lambda}^k = \lambda^k + \rho r(\hat{x}^k), \quad (12) \]
and
\[ \tilde{\lambda}^k = \lambda^k + \rho (I - T)r(x^k), \quad (13) \]
where \( I \) is the identity matrix of size \( m \).

The basic idea to show convergence of our method is to introduce the Lyapunov (merit) function
\[ \phi(x^k, \lambda^k) = \sum_{i=1}^{N} \rho \| A_i x_i^k - A_i x_i^* \|^2 + \frac{1}{\rho} \| \lambda^k - \lambda^* \|^2 - \frac{1}{\rho} \| x^k - x^* \|^2, \quad (14) \]
where we use the notation \( \| x \|_M = \sqrt{x^T M x} \). We will show in Theorem 1 that this Lyapunov function is strictly decreasing during the execution of the ADAL algorithm (7)-(9), given that the stepsizes \( \tau_i \) satisfy the condition \( 0 < \tau_i < 1/d_i \) for all \( j = 1, \ldots, m \). Then, in Theorem 2 we show that the strictly decreasing property of the Lyapunov function (14) implies the convergence of the primal and dual variables to their respective optimal values defined at a local minimum of problem (1).

We begin the proof by utilizing the first order optimality conditions of all the subproblems (7) in order to derive some necessary inequalities.

**Lemma 1:** Assume (A1)–(A4). Then, the following inequality holds:
\[ \sum_{i} \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^T (\hat{x}_i^k - x_i^*) + \frac{1}{\rho} (\hat{\lambda}_i^k - \lambda_i^*)^T (\hat{\lambda}_i^k - \lambda_i^k) \geq \rho \sum_{i} \left( A_i x_i - A_i x_i^* \right)^T \left( \sum_{j \neq i} (A_j x_j^* - A_j x_j^k) \right), \quad (15) \]
where \( \lambda_i^k, \hat{\lambda}_i^k, \hat{x}_i^k, \) and \( x_i^k \) are calculated at iteration \( k \).

**Proof:** The first order optimality conditions for problem (7) imply the following inclusion for the minimizer \( \hat{x}_i^k \)
\[ 0 \in \nabla f_i(\hat{x}_i^k) + A_i^T \lambda_i^k + \rho A_i^T (A_i x_i^k + \sum_{j \neq i} A_j x_j^k - b) + \lambda_i^k \quad \text{for all } i = 1, \ldots, N. \quad (16) \]
We infer that there exist normal elements \( z_i^k \in N_{\lambda_i^k}(\hat{x}_i^k) \) such that we can express (16) as follows:
\[ 0 = \nabla f_i(\hat{x}_i^k) + A_i^T \lambda_i^k + \rho A_i^T (A_i x_i^k + \sum_{j \neq i} A_j x_j^k - b) + z_i^k. \quad (17) \]
Taking inner product with \( x_i^k - \hat{x}_i^k \) on both sides of this equation and using the definition of a normal cone, we obtain
\[ \langle \nabla f_i(\hat{x}_i^k) + A_i^T \lambda_i^k \rangle + \rho A_i^T (A_i x_i^k + \sum_{j \neq i} A_j x_j^k - b), x_i^k - \hat{x}_i^k \rangle = \langle -z_i^k, x_i^k - \hat{x}_i^k \rangle \geq 0. \quad (18) \]
Using the variables \( \lambda_i^k \) defined in (12), we substitute \( \lambda_i^k \) in (18) and obtain:
\[ 0 \leq \langle \nabla f_i(\hat{x}_i^k) + A_i^T \left[ \hat{\lambda}_i^k - \rho \left( \sum_{j} A_j \hat{x}_j^k - b \right) \right. \right. \\
\left. \left. + \rho \left( A_i \hat{x}_i^k + \sum_{j \neq i} A_j x_j^k - b \right) \right], x_i^* - \hat{x}_i^k \rangle \\
= \langle \nabla f_i(\hat{x}_i^k) + A_i^T \left[ \tilde{\lambda}_i^k \right. \right. \\
\left. \left. + \rho \left( \sum_{j \neq i} A_j x_j^k - \sum_{j \neq i} A_j \hat{x}_j^k \right) \right], x_i^* - \hat{x}_i^k \rangle \quad (19) \]
The assumption (A4) entails that the following first-order optimality conditions are satisfied at the point \( (x^*, \lambda^*) \), i.e.,
\[ 0 \in \nabla f_i(x_i^*) + A_i^T \lambda_i^* + N_{\lambda_i}(x_i^*) \quad \text{for all } i = 1, \ldots, N. \quad (20) \]
After using the definition of the normal cone and taking inner product with \( \hat{x}_i^k - x_i^* \) on both sides of this equation (as before), we obtain the equivalent expression for the above inclusion
\[ \langle \nabla f_i(x_i^*) + A_i^T \lambda_i^*, \hat{x}_i^k - x_i^* \rangle \geq 0, \quad \text{for all } i = 1, \ldots, N. \quad (21) \]
Adding together (19) and (21), we obtain the following inequalities for all \( i = 1, \ldots, N \):
\[ \langle \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \rangle + A_i^T (\lambda_i^* - \hat{\lambda}_i^k) \]
\[ - \rho A_i^T \left( \sum_{j \neq i} A_j x_j^k - \sum_{j \neq i} A_j \hat{x}_j^k \right), \hat{x}_i^k - x_i^* \rangle \geq 0. \]
Adding the inequalities for all \( i = 1, \ldots, N \) and rearranging terms, we get:
\[ \sum_{i} \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^T (\hat{x}_i^k - x_i^*) + \left( \lambda_i^* - \hat{\lambda}_i^k \right)^T \left( \sum_{j \neq i} (A_i \hat{x}_j^k - A_i x_j^k) \right) \]
\[ \geq \rho \sum_{i} \left( A_i \hat{x}_i^k - A_i x_i^* \right)^T \left( \sum_{j \neq i} (A_j x_j^* - A_j \hat{x}_j^k) \right). \]
Substituting \( \sum_{i=1}^{N} A_i x_i^* = b \) and \( \sum_{i=1}^{N} A_i \hat{x}_i^k - b = \frac{1}{\rho} (\lambda^k - \hat{\lambda}^k) \) from (12), we conclude that
\[ \sum_{i} \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^T (\hat{x}_i^k - x_i^*) + \frac{1}{\rho} (\lambda^k - \hat{\lambda})^T (\lambda^k - \hat{\lambda}) \]
\[ \geq \rho \sum_{i} \left( A_i \hat{x}_i^k - A_i x_i^* \right)^T \left( \sum_{j \neq i} (A_j x_j^* - A_j \hat{x}_j^k) \right), \]
as required.

The following lemma is similar to Lemma 2 presented in [12]. The difference is that here the statement of the lemma includes also the gradient terms of the objective functions; in the convex case studied in [12] these terms are factored out due to the monotonicity property of the convex subdifferential. The proof of the lemma is given in the Appendix.
Lemma 2: Under assumptions (A1)–(A4), the following relation holds:
\[
\sum_{i} \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^*) \right)^T (\hat{x}_i^* - x_i^*) + \rho \sum_{i} \left( A_i x_i^k - A_i x_i^* \right)^T (A_i x_i^k - A_i \hat{x}_i^k) + \frac{1}{\rho} (\lambda^k - \lambda^*)^T (\lambda^k - \hat{\lambda}^k) \geq \rho \sum_{i} \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \frac{1}{\rho} \|\lambda^k - \lambda^k\|^2 + (\hat{\lambda}^k - \lambda^k)^T (r(x^k) - r(\hat{x}^k)).
\]

In the next lemma, we obtain a modified version of (22) whose right-hand side is nonnegative.

Lemma 3: Under the assumptions (A1)–(A4), the following relation holds
\[
\sum_{i} \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^*) \right)^T (\hat{x}_i^* - x_i^*) + \rho \sum_{i} \left( A_i x_i^k - A_i x_i^* \right)^T (A_i x_i^k - A_i \hat{x}_i^k) + \frac{1}{\rho} (\hat{\lambda}^k - \lambda^*)^T (\lambda^k - \hat{\lambda}^k) \geq \rho \sum_{i} \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \rho \|r(\hat{x}^k)\|^2_{T^{-\frac{1}{2}}D},
\]
where \(D = \text{diag}(q_1 \tau_1^2, \ldots, q_m \tau_m^2)\), and the variable \(\hat{\lambda}^k\) is defined in (13).

Proof: The first term in the left hand side of (22) includes the gradients of the objective functions will not be altered in what follows, so we neglect it temporarily for simplicity of notation. Add the term
\[
\frac{1}{\rho} (\rho(I - T)r(\hat{x}^k))^T (\hat{\lambda}^k - \lambda^*) = \rho((I - T)r(\hat{x}^k))^T (\hat{\lambda}^k - \lambda^*),
\]
to both sides of inequality (22). Recalling the definition of \(\hat{\lambda}^k\) from (13), we get:
\[
\rho \sum_{i} \left( A_i x_i^k - A_i x_i^* \right)^T (A_i x_i^k - A_i \hat{x}_i^k) + \frac{1}{\rho} (\hat{\lambda}^k - \lambda^*)^T (\lambda^k - \hat{\lambda}^k) \geq \rho \sum_{i} \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \rho \|r(\hat{x}^k)\|^2_{T^{-\frac{1}{2}}D} + (\hat{\lambda}^k - \lambda^k)^T (r(x^k) - r(\hat{x}^k)) - \rho((I - T)r(\hat{x}^k))^T r(\hat{x}^k).
\]

Consider the term \((\hat{\lambda}^k - \lambda^k)^T (r(x^k) - r(\hat{x}^k)) - \rho((I - T)r(\hat{x}^k))^T r(\hat{x}^k)\) in the right hand side of (24). We manipulate it to get:
\[
(\hat{\lambda}^k - \lambda^k)^T (r(x^k) - r(\hat{x}^k)) - \rho((I - T)r(\hat{x}^k))^T r(\hat{x}^k) = \rho(r(\hat{x}^k))^T (r(x^k) - r(\hat{x}^k)) - \rho((I - T)r(\hat{x}^k))^T (r(x^k) - r(\hat{x}^k)) = \rho(r(\hat{x}^k))^T (r(x^k) - r(\hat{x}^k)) - \rho((I - T)[r(x^k) - r(\hat{x}^k)] + r(\hat{x}^k))^T r(\hat{x}^k) = \rho(T r(\hat{x}^k))^T (r(x^k) - r(\hat{x}^k)) - \rho(r(\hat{x}^k))^T (I - T)r(\hat{x}^k) = \rho(T r(\hat{x}^k))^T (\sum_{i} A_i(x_i^k - \hat{x}_i^k)) - \rho(r(\hat{x}^k))^T (I - T)r(\hat{x}^k).
\]
Substituting back in (24), we obtain:
\[
\rho \sum_{i} \left( A_i x_i^k - A_i x_i^* \right)^T (A_i x_i^k - A_i \hat{x}_i^k) + \frac{1}{\rho} (\hat{\lambda}^k - \lambda^*)^T (\lambda^k - \hat{\lambda}^k) \geq \rho \sum_{i} \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \rho \|r(\hat{x}^k)\|^2_{T^{-\frac{1}{2}}D} \tag{26}
\]
where \([A_i]_{j} = 0\), then it follows that \([r(\hat{x}^k)]_{j}[A_i(x_i^k - \hat{x}_i^k)]_{j} = 0\).

Hence, denoting the set of nonzero rows of \(A_i\) as \(Q_i\), i.e., \(Q_i = \{j : 1, \ldots, m : [A_i]_{j} = 0\}\), we can obtain a tighter lower bound for each term \(\rho(T r(\hat{x}^k))^T (A_i x_i^k - A_i \hat{x}_i^k)\) as
\[
\rho(T r(\hat{x}^k))^T (A_i x_i^k - A_i \hat{x}_i^k) \geq -\frac{\rho}{2} \sum_{j \in Q_i} \left( [A_i(x_i^k - \hat{x}_i^k)]_{j}^2 + \tau_j^2 [r(\hat{x}^k)]_{j}^2 \right),
\]
where \([\cdot]_{j}\) denotes the \(j\)-th entry of a vector. Note, however, that some of the rows of \(A_i\) might be zero. If \([A_i]_{j} = 0\), then it follows that \([r(\hat{x}^k)]_{j}[A_i(x_i^k - \hat{x}_i^k)]_{j} = 0\). Hence, denoting the set of nonzero rows of \(A_i\) as \(Q_i\), i.e., \(Q_i = \{j : 1, \ldots, m : [A_i]_{j} = 0\}\), we can obtain a tighter lower bound for each term \(\rho(T r(\hat{x}^k))^T (A_i x_i^k - A_i \hat{x}_i^k)\) as
\[
\rho(T r(\hat{x}^k))^T (A_i x_i^k - A_i \hat{x}_i^k) \geq -\frac{\rho}{2} \sum_{j \in Q_i} \left( [A_i(x_i^k - \hat{x}_i^k)]_{j}^2 + \tau_j^2 [r(\hat{x}^k)]_{j}^2 \right). \tag{27}
\]
Now, recall that \(q_j\) denotes the number of non-zero blocks \([A_i]_{j}\) over all \(i = 1, \ldots, N\), in other words, \(q_j\) is the number of decision makers \(i\) that are involved in the constraint \(j\). Then, summing inequality (27) over all \(i\), we observe that each term \(\tau_j^2 [r(\hat{x}^k)]_{j}^2\) is included in the summation at most \(q_j\) times.
This observation leads us to the bound
\[ \rho \sum_i \left( T r(\hat{x}^k) \right)^T \left( A_i x_i^k - A_i \hat{x}_i^k \right)\]
\[ \geq -\frac{\rho}{2} \left( \sum_i \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \sum_{j=1}^m q_j \tau_j^2 |r(\hat{x}^k)|^2_j \right), \]

or, equivalently,
\[ \rho \sum_i \left( T r(\hat{x}^k) \right)^T \left( A_i x_i^k - A_i \hat{x}_i^k \right)\]
\[ \geq -\frac{\rho}{2} \sum_i \|A_i(x_i^k - \hat{x}_i^k)\|^2 - \frac{\rho}{2} r(\hat{x}^k)^T D r(\hat{x}^k), \quad (28) \]

where \( D = \text{diag}(q_1 \tau_1^2, \ldots, q_m \tau_m^2). \) Finally, we substitute (28) into (26) to get
\[ \rho \sum_i \left( A_i x_i^k - A_i \hat{x}_i^k \right)^T \left( A_i x_i^k - A_i \hat{x}_i^k \right)\]
\[ + \frac{1}{\rho} \left( \lambda^k - \lambda^* \right)^T \left( \lambda^k - \lambda^* \right)\]
\[ \geq \frac{\rho}{2} \sum_i \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \rho \|r(\hat{x}^k)\|^2_{T^{-\frac{1}{2}} D^{-1}}. \]

After reinstating the gradient terms that we have neglected thus far, we obtain the required result. \( \square \)

Next, our goal is to find a lower bound for the gradient terms appearing in (23). To do so, let \( C \) denote any diagonal matrix with strictly positive diagonal entries, and consider the function
\[ G_\rho(x) = F(x) + \frac{\rho}{2} x^T A^T C A x, \quad (29) \]

where we recall that \( x = [x_1^T, \ldots, x_n^T]^T \in \mathbb{R}^n, \)
\( F(x) = \sum_i f_i(x_i), \)
\( A = [A_1 \ldots A_N] \in \mathbb{R}^{m \times n}. \) In the next lemmas, we will make use of the fact that, for sufficiently large \( \rho, \) the function \( G_\rho(x) \) is strongly convex in a neighborhood around the optimal solution \( x^* \) of (1). For this, we will use the making of the following result.

**Lemma 4** ([14], Lemma 4.28): Assume that a symmetric matrix \( Q \) of dimension \( n \) and a matrix \( B \) of dimension \( m \times n \) are such that
\[ \langle x, Bx \rangle > 0, \quad \text{for all } x \neq 0 \] such that \( Bx = 0. \)

Then, there exists \( \rho_0 \) such that for all \( \rho > \rho_0 \) the matrix \( Q + \rho B^T B \) is positive definite.

Using Lemma 4, we can obtain an important relation involving the gradient terms that appear in (23).

**Lemma 5**: Assume (A1)–(A4). Then, for any diagonal matrix \( C \) with strictly positive diagonal entries, there exists some \( \kappa > 0 \) such that the following relation holds
\[ \rho \sum_i \left( A_i x_i^k - A_i \hat{x}_i^k \right)^T \left( A_i x_i^k - A_i \hat{x}_i^k \right)\]
\[ + \frac{1}{\rho} \left( \lambda^k - \lambda^* \right)^T \left( \lambda^k - \lambda^* \right)\]
\[ \geq \frac{\rho}{2} \sum_i \|A_i(x_i^k - \hat{x}_i^k)\|^2 + \rho \|r(\hat{x}^k)\|^2_{T^{-\frac{1}{2}} D^{-1}} + \kappa \|x^k - x^*\|^2, \quad (30) \]

provided that \( \rho \) is sufficiently large, and that for all iterations \( k \) the terms \( \lambda^k + \rho \sum_{j \neq i} (A_j x_j^k - A_j \hat{x}_j^k) \) are sufficiently close to \( \lambda^* \) for all \( i = 1, \ldots, N. \)

**Proof**: From assumption (A4), we have that there exists a point \( x^* \) satisfying the strong second order sufficient conditions of optimality for problem (1). These conditions imply that
\[ \langle s, \nabla^2 F(x^*) s \rangle > 0, \quad \text{for all } s \neq 0, \] such that \( As = 0. \)

Now, combine this with the result of Lemma 4 for
\[ Q = \nabla^2 F(x^*), \quad \text{and } B = C^{1/2} A. \]

It follows that there exists \( \rho_0 \) such that for all \( \rho > \rho_0 \) the matrix \( \nabla^2 F(x^*) + \rho A^T CA \) is positive definite with some modulus \( \kappa_0 > 0. \) Moreover, from assumption (A2) the matrix \( \nabla^2 F(x) + \rho A^T CA \) (note that this matrix is defined w.r.t. \( x \) instead of \( x^* \)) is also continuous, hence there exists sufficiently large \( \rho \) such that, for all \( x \in \mathcal{X} \) sufficiently close to \( x^* \), i.e., for \( \|x - x^*\| \leq \beta \), all the eigenvalues of \( \nabla^2 F(x) + \rho A^T CA \) lie above some \( \kappa > 0. \) To see this, observe that from Schwarz’s theorem [51] we have that the continuous differentiability assumption (A2) means that the Hessian matrix \( H(x) = \nabla^2 F(x) \) is symmetric within \( \mathcal{X}. \) According to eigenvalue perturbation theory [52], the symmetry of the Hessian entails that for a perturbation \( \delta H \) of the matrix \( H, \) the perturbation \( \delta \epsilon \) of its smallest eigenvalue \( \epsilon \) is bounded by \( \delta H, \) i.e., \( |\delta \epsilon| \leq \|\delta H\|. \) By the continuity of the Hessian, we infer that there exists some neighborhood \( \|x - x^*\| \leq \beta \) around \( x^* \) such that \( |\delta \epsilon| < \kappa_0, \) which in turn means that within this neighborhood the matrix \( \nabla^2 F(x) + \rho A^T CA \) remains positive definite with a modulus at least \( \kappa = \kappa_0 - |\delta \epsilon| > 0. \)

Since the positive definite matrix \( \nabla^2 F(x) + \rho A^T CA \) is the Hessian of the function \( G_\rho(x) = F(x) + \frac{\rho}{2} x^T A^T C A x \) and \( \mathcal{X} \) is a convex closed set, we infer that, for sufficiently large \( \rho, \) there exists some \( \beta \) such that the function \( G_\rho(x) \) is strongly convex with modulus \( \kappa \) for every \( x \) belonging in the set \( \{x \in \mathcal{X} : \|x - x^*\| \leq \beta\}. \) From the definition of strongly convex functions, we get that the following holds for all \( x \) that are sufficiently close to \( x^*: \)
\[ \left( \nabla G_\rho(x) - \nabla G_\rho(x^*) \right)^T (x - x^*) \geq \kappa \|x - x^*\|^2. \]
For the term \((\nabla G_\rho(x) - \nabla G_\rho(x^*)^T(x - x^*)\), we have
\[
\left(\nabla G_\rho(x) - \nabla G_\rho(x^*)^T\right)(x - x^*) = \left(\nabla F(x) - \nabla F(x^*) + \rho A^TCA(x - x^*)\right)^T(x - x^*)
\]
\[
= \left(\nabla F(x) - \nabla F(x^*) + \rho A^TC(Ax - b)\right)^T(x - x^*)
\]
\[
= \sum_i \left(\nabla f_i(x_i) - \nabla f_i(x_i^*)\right)^T(x_i - x_i^*) + \rho r(x)^TCr(x)
\]
where we have used the fact that \(Ax^* = b\), and \(r(x) = Ax - b\). It follows that
\[
\sum_i \left(\nabla f_i(x_i) - \nabla f_i(x_i^*)\right)^T(x_i - x_i^*) + \rho r(x)^TCr(x)
\geq \kappa \|x - x^*\|^2.
\]
(31)

Now, substitute \(x = \tilde{x}^k\) in (31), and add it to (23). We get the following relation
\[
\rho \sum_i \left(A_i x_i^k - A_i x_i^\lambda\right)^T \left(A_i x_i^k - A_i x_i^\lambda\right)
+ \frac{1}{\rho} \left(\lambda^k - \lambda^*\right)^T \left(\lambda^k - \lambda^*\right)
\geq \rho \sum_i \|A_i (x_i^k - \tilde{x}^k)\|^2 + \rho r(\tilde{x}^k)^T \left(T - \frac{1}{2}D - C\right)r(\tilde{x}^k)
+ \kappa \|\tilde{x}^k - x^*\|^2,
\]
which is the required result.

Note that, in order to substitute \(x = \tilde{x}^k\) in (31), it is necessary that the \(\tilde{x}^k\) are sufficiently close to \(x^*\) at iteration \(k\), i.e., that they belong to the set \(\{\tilde{x} \in \mathcal{X} : \|\tilde{x} - x^*\| \leq \beta\}\). To see when this condition holds, note that the local AL for each \(i\) can be expressed as
\[
A_i^k(x_i, Ax^k, \lambda^k) = f_i(x_i)
+ \left(\lambda^k + \rho \sum_{j \neq i}(A_j x_j^k - A_j x_j^\lambda), A_i x_i\right)
+ \frac{\rho}{2} \|A_i x_i + \sum_{j \neq i} A_j x_j^k - b\|^2 + \frac{\rho}{2} \sum_{j \neq i} \|A_j x_j^k - A_j x_j^\lambda\|^2
+ \rho \left(\sum_{j \neq i} (A_j x_j^k - A_j x_j^\lambda), A_j x_j^k - A_j x_j^\lambda\right),
\]
where we have added the zero terms \(\sum_{j \neq i} A_j x_j^k - \sum_{j \neq i} A_j x_j^\lambda\) in the penalty term of the AL, and expanded it. The last two terms can be disregarded when minimizing with respect to \(x_i\). Recalling a well-known result on the sensitivity analysis of augmented Lagrangians, c.f. [14, 53, 54], we have that, given assumptions (A1)-(A4), if \(\rho\) is sufficiently large and the terms \(\xi_i = \lambda^k + \rho \sum_{j \neq i}(A_j x_j^k - A_j x_j^\lambda)\) are sufficiently close to \(\lambda^*\) for all \(i = 1, \ldots, N\), then \(\sup_i \|x_i - x_i^\lambda\| = O(\|\xi_i - \lambda^*\|)\) holds, i.e., the \(x_i^k\) will be sufficiently close to \(x_i^\lambda\), as required for (31) to hold.

We are now ready to prove the key result pertaining to the convergence of our method. We will show that the function \(\phi\) defined in (14) is a strictly decreasing Lyapunov function for ADAL. The results from Lemmas 3 and 5 will help us characterize the decrease of \(\phi\) at each iteration.

**Theorem 1:** Assume (A1)-(A4). Assume also that \(\rho\) is sufficiently large, and that the initial iterates \(x^1, \lambda^1\) are chosen such that \(f(x^1, \lambda^1)\) is sufficiently small. If the ADAL method uses stepsizes \(\tau_j\) satisfying
\[
0 < \tau_j < \frac{1}{\rho'}, \quad \forall j = 1, \ldots, m,
\]
then, the sequence \(\{\phi(x^k, \lambda^k)\}\), with \(\phi(x^k, \lambda^k)\) defined in (14), is strictly decreasing.

**Proof:** First, we show that the dual update step (9) in the ADAL method results in the following update rule for the variables \(\tilde{x}^k\), which are defined in (13):
\[
\tilde{x}^{k+1} = \tilde{x}^k + \rho \text{Tr}(\tilde{x}^k)
\]
Indeed,
\[
\tilde{x}^{k+1} = \lambda^k + \rho \text{Tr}(\tilde{x}^{k+1})
\]
\[
\lambda^{k+1} = \lambda^k + \rho \text{Tr}(\tilde{x}^{k+1}) + \rho \text{Tr}(x^{k+1})
\]
\[
\lambda^{k+1} = \lambda^k + \rho (I - T)\text{Tr}(x^{k+1}) + \rho \text{Tr}(\tilde{x}^k)
\]
\[
\lambda^{k+1} = \lambda^k + \rho \text{Tr}(\tilde{x}^k),
\]
as required.

We define the progress at each iteration \(k\) of the ADAL method as
\[
\theta_k(T) = \phi(x^k, \lambda^k) - \phi(x^{k+1}, \lambda^{k+1}).
\]
We substitute \(\lambda^k\) in the formula for calculating the function \(\phi\) and use relation (33). The progress \(\theta_k(T)\) can be evaluated as follows:
\[
\theta_k(T) = \sum_{i=1}^N \rho \|A_i(x_i^k - x_i^*)\|^2_{T^{-1}} - \rho \|A_i(x_i^{k+1} - x_i^*)\|^2_{T^{-1}} + \rho \|\tilde{x}^{k+1} - x^*\|^2_{T^{-1}}.
\]
First, consider the term \(\|A_i(x_i^{k+1} - x_i^*)\|^2_{T^{-1}}\). We have that
\[
\rho \|A_i(x_i^{k+1} - x_i^*)\|^2_{T^{-1}} = \rho \left(\tilde{x}^{k+1} - x_i^\lambda\right)^T \left(T^{-1}(A_i x_i^k + b) - \tilde{x}^k\right)
+ \rho \left(T(A_i x_i^k + b) - \tilde{x}^k\right)^T \left(T^{-1}(A_i x_i^k + b) - \tilde{x}^k\right)
+ 2 \rho \left(A_i x_i^k + b - \tilde{x}^k\right)^T \left(T^{-1}(A_i x_i^k + b) - \tilde{x}^k\right),
\]
where we substituted \(A_i x_i^{k+1} = A_i x_i^k + T(A_i x_i^k - A_i x_i^\lambda)\) from (8) and expanded the terms. The last equation in the above can be written as
\[
\rho \|A_i(x_i^{k+1} - x_i^*)\|^2_{T^{-1}} = \rho \|A_i(x_i^k - x_i^*)\|^2_{T^{-1}} + \rho \|\tilde{x}^k - x_i^\lambda\|^2_{T^{-1}}
+ 2 \rho \left(A_i x_i^k + b - \tilde{x}^k\right)^T \left(A_i x_i^k - A_i x_i^\lambda\right).
\]
(35)
Similarly, for the term \( \|\hat{x}^{k+1} - x^*\|_{T^{-1}}^2 \), we have
\[
\frac{1}{\rho} \|\hat{x}^{k+1} - x^*\|_{T^{-1}}^2 = \\
= \frac{1}{\rho} (\hat{x}^{k+1} - x^*)^T T^{-1} (\hat{x}^{k+1} - x^*) \\
= \frac{1}{\rho} (\hat{x}^k - x^* + \rho \text{Tr}(\hat{x}^k))^T T^{-1} (\hat{x}^k - x^* + \rho \text{Tr}(\hat{x}^k)) \\
= \frac{1}{\rho} \|\hat{x}^k - x^*\|_{T^{-1}}^2 + \rho \|\text{Tr}(\hat{x}^k)\|_{T}^2 \\
\]
where in the second equality we have used relation (33). Hence, substituting (35) and (36) into (34), and recalling that \( \hat{x}^k - x^k = \rho r(\hat{x}^k) \) we get that the progress \( \theta_k(T) \) at each iteration is given by
\[
\theta_k(T) = 2\rho \sum_i (A_i x^k_i - A_i x^*_i)^T (A_i x^k_i - A_i x^*_i) \\
+ \frac{2}{\rho} (\hat{x}^k - x^*)^T (\hat{x}^k - x^*) \\
- \rho \sum_i \|A_i \hat{x}^k_i - A_i x^*_{ki}\|_{T}^2 - \rho \|r(\hat{x}^k)\|_{T}^2. \\
(37)
\]
The last two (quadratic) terms in (37) are always negative, due to \( T \) being positive definite by construction. Hence, in order to show that \( \phi \) is strictly decreasing, we need to show that the first two terms in (37) are always “more positive” than the last two terms. This is exactly what Lemma 5 and (30) enable us to do. In particular, using (30), we obtain a lower bound for the first two terms in (37), which gives us
\[
\theta_k(T) \geq \rho \sum_i \|A_i \hat{x}^k_i - A_i x^*_{ki}\|_{T}^2 + \rho \|r(\hat{x}^k)\|_{T}^2 \\
+ 2\kappa \|\hat{x}^k - x^*\|_{T}^2 - \rho \sum_i \|A_i \hat{x}^k_i - A_i x^*_{ki}\|_{T}^2 - \rho \|r(\hat{x}^k)\|_{T}^2 \\
= \rho \sum_i \|A_i \hat{x}^k_i - A_i x^*_{ki}\|_{T}^2 + \rho \|r(\hat{x}^k)\|_{T}^2 \\
+ 2\kappa \|\hat{x}^k - x^*\|_{T}^2. \\
(38)
\]
The above relation suggests that we can choose \( T \) appropriately in order to guarantee that \( \phi \) is strictly decreasing. Specifically, it is sufficient to ensure that the matrices \( I - T \) and \( T - D - 2C \) are positive definite. From the condition \( I - T > 0 \), we infer that the diagonal elements of \( T \) must be strictly less than one. To ensure that \( T - D - 2C > 0 \), recall that \( D = \text{diag}(q_1 \tau^2_1, \ldots, q_m \tau^2_m) \) by construction. Also, according to Lemma 5, the matrix \( C \) can be any diagonal matrix with strictly positive diagonal entries. Let \( C = \frac{1}{2}TE \), where \( E = \text{diag}(\epsilon_1, \ldots, \epsilon_m) \), and each \( \epsilon_j, j = 1, \ldots, m \) is an arbitrarily small, positive number. Then, if we can choose \( T \) such that
\[
\tau_j - q_j \tau^2_j - \epsilon_j \tau_j > 0, \quad \forall j = 1, \ldots, m, \\
\]
the diagonal matrix \( T - D - 2C \) is guaranteed to be positive definite. The above relation has solution
\[
\tau_j < \frac{1 - \epsilon_j}{q_j}, \quad \forall j = 1, \ldots, m. \\
(39)
\]
Hence, if we select \( \tau_j \) according to (39), then \( \theta_k > 0 \) during the execution of ADAL, which in turn means that the sequence \( \{\phi(x^k, \lambda^k)\} \) is strictly decreasing, as required. Since the \( \epsilon_j \) can be as small as we want, we obtain the corresponding condition of the theorem.

Note that to arrive at (38), we have used the result of Lemma 5, which requires that the terms \( \lambda^k + \rho \sum_j (A_j x^k_j - A_j x^*_j - \lambda^k) \) are sufficiently close to zero for all \( i = 1, \ldots, N \) at iteration \( k \); recall that the purpose of this condition is to guarantee that the \( \hat{x}_i^k \) will fall into the strong convexity region of \( G_\rho \), which allows us to use (31). Suppose also that the terms \( A_i x^k_i - A_i x^*_i \) are sufficiently close to zero for all \( i = 1, \ldots, N \) at iteration \( k \). Adding \( A_i x^k_i - A_i x^*_i \) to \( \lambda^k + \rho \sum_j (A_j x^k_j - A_j x^*_j - \lambda^k) \), the assumption that the \( \lambda^k + \rho \sum_j (A_j x^k_j - A_j x^*_j - \lambda^k) \) are sufficiently close to zero at iteration \( k \) for all \( i = 1, \ldots, N \) in Lemma 5, becomes equivalent to the condition that the terms \( \lambda^k + \rho r(\hat{x}^k) - \lambda^* \) and \( A_i x^k_i - A_i x^*_i \) for all \( i = 1, \ldots, N \) are sufficiently close to zero at iteration \( k \).

Note that \( \lambda^k + \rho r(\hat{x}^k) - \lambda^* \) is sufficiently close to zero if and only if \( \lambda^k + (I - T)r(\hat{x}^k) - \lambda^* \) is sufficiently close to zero at iteration \( k \). This is because \( (I - T) \) is a finite multiplicative factor on \( r(\hat{x}^k) \) and \( r(\hat{x}^k) \) is close to zero, since the \( A_i x^k_i \) are close to \( A_i x^*_i \) for all \( i = 1, \ldots, N \). Now, recall the definition of \( \phi(x^k, \lambda^k) = \sum_{i=1}^N \rho \|A_i x^k_i - A_i x^*_i\|_{T}^2 + \frac{1}{\rho} \|\lambda^k + (I - T)r(\hat{x}^k) - \lambda^*\|_{T}^{-1} \), and observe that the terms in the right hand side of \( \phi(x^k, \lambda^k) \) are exactly the terms that we need to be sufficiently close to zero in order to apply the result of Lemma 5. Since \( 0 < T < I \), it follows that the terms \( \lambda^k + (I - T)r(\hat{x}^k) - \lambda^* \) and \( A_i x^k_i - A_i x^*_i \) for all \( i = 1, \ldots, N \) are sufficiently close to zero if \( \phi(x^k, \lambda^k) \) is sufficiently small. To see this, observe that all terms in the expression for \( \phi(x^k, \lambda^k) \) are individually upper bounded by the value of \( \phi(x^k, \lambda^k) \), e.g., \( \|A_i x^k_i - A_i x^*_i\|_{T}^2 < \|A_i x^k_i - A_i x^*_i\|_{T}^{-1} \leq \rho \phi(x^k, \lambda^k) \). Hence, if we choose initial values \( x^1, \lambda^1 \) such that \( \phi(x^1, \lambda^1) \) is sufficiently small, then \( \theta_1 > 0 \), which implies that \( \phi(x^2, \lambda^2) < \phi(x^1, \lambda^1) \). Since \( \phi(x^1, \lambda^1) \) is sufficiently small and \( \phi(x^2, \lambda^2) \) is even smaller, we can infer that the iterates \( \lambda^k + \rho \sum_j (A_j x^k_j - A_j x^*_j) \) will be sufficiently close to zero for all iterations \( k \). Therefore, the result of Lemma 5 can be used, as required.

**Remark 2:** In the statement of Theorem 1, we assume that the initial iterates \( x^1, \lambda^1 \) are chosen such that \( \phi(x^1, \lambda^1) \) is sufficiently small. For comparison, in the convergence proof of the standard augmented Lagrangian method (ALM) described in Alg. 1, the assumption that the dual iterates \( \lambda^k \) are sufficiently close to \( \lambda^* \) for all iterations is used. Following a similar argument as in Theorem 1, this condition holds true if the initial values \( \lambda^1 \) are sufficiently close to \( \lambda^* \); see [14, 55, 56] for more details. Here, we cannot simply require that the dual variables alone are close to their optimal values. Instead, we need to consider the terms \( \lambda^k + \rho \sum_j (A_j x^k_j - A_j x^*_j) \) for all \( i = 1, \ldots, N \), due to the structure of the local ALS, cf. (32), and the distributed nature of the algorithm. This difference gives rise to the condition that \( \phi(x^1, \lambda^1) \) is sufficiently small, which replaces the requirement that \( \lambda^1 \) is sufficiently close to
\( \lambda^* \) as is the case in the ALM.

We are now ready to prove the main result of this paper.

**Theorem 2:** Assume (A1)-(A4). Assume also that \( \rho \) is sufficiently large, and that the initial iterates \( x^1, \lambda^1 \) are chosen such that \( \phi(x^1, \lambda^1) \) is sufficiently small. Then, the ADAL method generates sequences of primal variables \( \{\hat{x}^k\} \) and dual variables \( \{\lambda^k\} \) that converge to a local minimum \( x^* \) of problem (1) and the corresponding optimal Lagrange multipliers \( \lambda^* \), respectively.

**Proof:** Relation (38) implies that
\[
\phi(x^{k+1}, \lambda^{k+1}) \leq \phi(x^k, \lambda^k) - \rho \sum_i \|A_i \tilde{x}_i^k - A_i x_i^k\|_{1-T}^2 - \rho \|r(x^k)\|_{T-D-2C}^2 - 2\kappa \|\hat{x}^k - x^*\|^2
\]
Summing the above inequality for \( k = 1, 2, \ldots, \) we obtain:
\[
\sum_{k=1}^{\infty} \left[ \rho \sum_i \|A_i \tilde{x}_i^k - A_i x_i^k\|_{1-T}^2 + \rho \|r(x^k)\|_{T-D-2C}^2 + 2\kappa \|\hat{x}^k - x^*\|^2 \right] < \phi(x^1, \lambda^1)
\]
Since \( \phi(x^1, \lambda^1) \) is bounded, this implies that the sequences \( \{r(x^k)\}, \{\tilde{x}_i^k - x_i^k\} \), and \( \{A_i \tilde{x}_i^k - A_i x_i^k\} \) for all \( i = 1, \ldots, N \), converge to zero as \( k \to \infty \). It follows that \( \{r(x^k)\} \) converges to zero as well. By the monotonicity and boundedness properties of \( \phi(x^k, \lambda^k) \), we conclude that the sequence \( \{\lambda^k\} \) is also convergent. We denote \( \lim_{k \to \infty} \lambda^k = \mu \).

From assumption (A2), the gradients of the functions \( f_i \) are continuous on \( \lambda_i \). Therefore, the sequences \( \{\nabla f_i(x_i^k)\} \) converge to \( \nabla f_i(x_i^*) \) for all \( i = 1, \ldots, N \). Passing to the limit in equation (17), we infer that each sequence \( \{z_i^k\} \) converges to a point \( z_i, i = 1, \ldots, N \). The mapping \( x_i \mapsto N_{\lambda_i}(x_i) \) has closed graph and, hence, \( z_i \in N_{\lambda_i}(x_i^*) \).

After the limit pass in (17), we conclude that
\[
0 = \nabla f_i(x_i^*) + A_i^T \mu + z_i, \quad \forall i = 1, \ldots, N.
\]
Hence, \( \mu \) satisfies the first order optimality conditions for problem (1). Since \( x^* \) is a feasible point that satisfies the strong second order sufficient conditions of optimality for problem (1), we conclude that ADAL generates primal sequences \( \{\hat{x}^k\} \) that converge to a local minimum \( x^* \) of (1), and dual sequences \( \{\lambda^k\} \) that converge to their optimal values \( \lambda^* \) for the point \( x^* \), as required.

**Remark 3:** The sufficient closeness assumption used in this paper, cf. Lemma 5 and Theorem 1, is required to establish strong convexity of the local ALs and, subsequently, local convergence of the proposed distributed AL method. Analogous proximity assumptions are used to show convergence of the centralized AL method for nonconvex problems in [14]. Nevertheless, for problems where the constraint matrices \( A_i \) are full column rank, the sufficient closeness assumption is no longer necessary. Instead, for problems with this structure the strong convexity of the local ALs can be established by selecting a large enough value for \( \rho \), which can be computed based on bounds on the gradients of the non-convex functions \( f_i \) at all points in the constraint space, in a spirit similar to the analysis presented in [48].

**IV. NUMERICAL EXPERIMENTS**

In order to illustrate the proposed method, in this section we present numerical results of ADAL applied to non-convex optimization problems. The main objectives here are two. First, we verify the correctness of the theoretical analysis developed in Section III by showing that the proposed distributed method converges to a local minimum. We also show that the Lyapunov function defined in (14) is indeed strictly decreasing for all iterations, as expected. Second, we examine how sensitive ADAL is to the choice of the user-defined penalty coefficient \( \rho \), and also to different initialization points.

Since the problems are non-convex, ADAL will converge to some local minimum. To evaluate the quality of this local minimum, we use the solution that is obtained by directly solving the non-convex problems with a commercial nonlinear optimization solver; we refer to that solution as “centralized”, as we do not enforce any decomposition when using this solver. Note that the goal here is not to compare the centralized solution to the solution that is returned by ADAL, but rather to establish that ADAL does not converge to trivial solutions. In comparison, in the convex case we would compare the solution of ADAL to the global optimal solution and show that they are the same. The simulations were carried out in MATLAB, using the \texttt{fmincon} command to solve the centralized problem, as well as the non-convex local subproblems (7) at each iteration of ADAL. \footnote{We note that, for the problems considered here, the \texttt{fmincon} solver of Matlab returned the same solutions as other solvers such as MINOS, LANCELOT, SNOPT, and IPOPT in AMPL for the vast majority of cases. Since the purpose of this paper is not to compare the performance of nonlinear optimization solvers, we have focused just on the \texttt{fmincon}.}

The results correspond to the ”active-set” solver option of \texttt{fmincon}, which performed better than all other options, in terms of optimality and computation time.

First, we examine a simple non-convex optimization problem with \( N = 2 \) agents that control their decision variables \( x_1 \) and \( x_2 \), respectively. The problem is:
\[
\min_{x_1, x_2} x_1 \cdot x_2, \quad \text{s.t.} \quad x_1 - x_2 = 0. \tag{41}
\]
This problem is particularly interesting because the straightforward application of the popular ADMM algorithm fails to
converge, as discussed in [50]. The problem has an obvious optimal solution at \( x_1^* = x_2^* = \lambda^* = 0 \). It is shown in [50] that initializing ADMM at \( x_1^* = x_2^* = 0 \) and \( \lambda^1 \neq 0 \) for this problem gives iterates of the form \( x^{k+1} = 0 \) and \( \lambda^{k+1} = -2\lambda^k \), and we can see how the latter update produces a diverging dual sequence. On the other hand, the proposed ADAL method is convergent for the same initialization, as can be seen in Fig. 1.

Next, we consider a non-convex problem with \( N = 6 \) agents, where each agent controls a scalar decision variable \( x_i, i = 1, \ldots, 6 \) that is subject to box constraints. Each agent has a different non-convex objective function and all decisions are coupled in a single linear constraint:

\[
\min_x \quad \cos(x_1) + \sin(x_2) + e^{x_3} + 0.1x_4^3 + \frac{1}{1 + e^{-x_5}} + 0.05(x_6^5 - x_6 - x_6^4 + x_6^3)
\]

s.t. \( x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 4 \), \( -5 \leq x_i \leq 5, \quad \forall i = 1, \ldots, 6 \). (42)

The simulation results for this problem are depicted in Fig. 2, where we compare the solutions of ADAL and the centralized solver for 50 different initialization instances. For each instance, the initialization point is the same for both ADAL and the centralized solver. The red and blue squares indicate the objective function value at the point of convergence for the centralized method and ADAL, respectively. A blue dashed line indicates that ADAL converged to a better (or the same) local minimum, while a red dashed line indicates the opposite.

Next, we consider a problem with multiple constraints \( m = 5 \), more agents \( N = 8 \), and larger box constraint sets

\[
\min_x \quad \cos(x_1) + \sin(x_2) + e^{x_3} + 0.1x_4^3 + \frac{1}{1 + e^{-x_5}} + 0.01(x_6^5 - x_6 - x_6^4 + x_6^3) + \sqrt{x_7 + 15}\sin(x_7/10) + e^{x_8}/(x_8^2 + e^{x_8})
\]

s.t. \( Ax = b \), (43)

where the constraint parameters \( A \in \mathbb{R}^{5 \times 8} \) and \( b \in \mathbb{R}^5 \) are randomly generated with entries sampled from the standard normal distribution (such that the problem is feasible). When generating \( A \), we always ensure that it has full row rank (to prevent trivial constraint sets), and that at least two decision variables are coupled in each constraint.

Fig. 3 depicts the convergence results of ADAL applied to problem (43), where the generated matrix \( A \) is

\[
\begin{pmatrix}
0 & 0 & 1.2634 & 0.9864 & 0 & 0.4970 & -0.2259 & -0.2783 \\
-1.8780 & 0 & 0 & 0 & 0 & -2.5070 & -0.8325 & 0 \\
-0.8666 & 0 & 0 & 0.2461 & -0.1226 & 0 & 0 & 0
\end{pmatrix},
\]

and \( b = [-0.0579, -1.6883, 0.8465, 0.1843, 0.6025]^T \). In this case the stepsizes are set to \( T = \text{diag}(1/5, 1/2, 1/3, 1/2, 1/3) \). To examine how sensitive ADAL is to the choice of the user-defined penalty coefficient \( \rho \), we present convergence results...
for four different choices $\rho = 1, 3, 10, 20$. We terminate ADAL after reaching a maximum constraint violation of $3e{-4}$. Two significant observations can be made based on these results. On one hand, choosing larger values of $\rho$, e.g. 10 or 20, leads to faster convergence, albeit at the cost of converging to a worse local minimum in terms of objective function value. On the other hand, choosing small $\rho$, e.g. 1 or 3, allows ADAL to find a better solution, however, convergence of the constraint violation slows down significantly after reaching accuracy levels of about $1e{-3}$. Furthermore, to clarify the necessary communication pattern between agents during the execution of ADAL, cf. the pertinent discussion in section II, Fig. 4 illustrates the communication network that needs to be established for this particular problem. For example, agent 2 is coupled only in the 2nd constraint with agent 6, hence, it only needs to communicate with 6.

In order to test the sensitivity of ADAL to initialization for problem (43), we test it for 50 different initialization instances. The results are depicted in Fig. 5(a), where we also include the solutions obtained from the centralized scheme for the same initializations as ADAL. We observe that, for this problem, the choice of initialization point plays a more significant role in determining which local minimum ADAL will converge to, as compared to the corresponding results for the previous problem (42) where ADAL converged to the same local minimum for the vast majority of initializations. Moreover, in Fig. 5(b) we plot the evolution of the Lyapunov function $\phi(x^k, \lambda^k)$, cf. (14), for an instance of problem (43) where ADAL is initialized (randomly) at $x^0 = [4.993, -5.904, -4.087, 2.292, -1.648, -2.883, 6.388, 7.331]$ and $\lambda = 0$ with $\rho = 1$. We observe that $\phi$ is strictly decreasing at each iteration, as expected.

Next, we test ADAL on problems of the form (43) for 50 different instances of the problem parameters $\mathbf{A}$ and $\mathbf{b}$. The objective of this experiment is to examine the behavior of ADAL with a predefined value of $\rho$ for a wide range of problems, instead of finding the best $\rho$ for a given problem as in Fig. 3. This is important for practical applications, where we need to choose a value for $\rho$ without knowing the exact problem parameters. In order to ensure that $\rho$ is sufficiently large for all problem realizations, in this experiment we set $\rho = 5$. We terminate ADAL after reaching a maximum constraint violation of $3e{-4}$. The results are shown in Fig. 6. We observe that overall the performance of ADAL is satisfactory, judging by the fact that it converges to the same local minimum as the centralized solver for most of the cases.

In the theoretical analysis of section III, we used the assumptions that the initialization point is sufficiently close to a locally optimal solution and that $\rho$ is large enough. Here, we perform numerical experiments to explore more thoroughly how these conditions affect the convergence of the proposed method. Towards this goal, we consider the following optimal consensus problem, where 25 agents have different versions of the Rosenbrock function and all need to agree on a common optimal decision that minimizes the sum of the individual...
TABLE I

<table>
<thead>
<tr>
<th>Value of $\rho$</th>
<th>50</th>
<th>100</th>
<th>250</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Converged cases</td>
<td>820 (41%)</td>
<td>960 (48%)</td>
<td>220 (11%)</td>
<td>N/A</td>
</tr>
<tr>
<td>Mean obj. value</td>
<td>346.83</td>
<td>817.04</td>
<td>2003.9</td>
<td>N/A</td>
</tr>
<tr>
<td>Min obj. value</td>
<td>257.78</td>
<td>505.02</td>
<td>1483.8</td>
<td>N/A</td>
</tr>
<tr>
<td>Max obj. value</td>
<td>479.66</td>
<td>1319.4</td>
<td>2250</td>
<td>N/A</td>
</tr>
</tbody>
</table>

V. CONCLUSIONS

In this paper we have investigated a distributed solution technique for a certain class of non-convex constrained optimization problems. In particular, we have considered the problem of minimizing the sum of, possibly non-convex, local objective functions whose arguments are local variables that are constrained to lie in closed, convex sets. The local variables are also globally coupled via a set of affine constraints. We have proposed an iterative distributed algorithm and established its convergence to a local minimum of the problem under assumptions that are commonly used for the convergence of non-convex optimization methods. The proposed method is based on the augmented Lagrangian framework and is an extension of previous work that considered only convex problems. To the best of our knowledge this is the first paper that formally establishes the convergence to local minima for a distributed augmented Lagrangian method in non-convex settings. Moreover, compared to our previous work, in this paper we have proposed a more general and fully decentralized rule to select the stepsizes involved in the method. We have verified the theoretical convergence analysis via numerical simulations.

VI. APPENDIX

Proof of Lemma 2: Consider the result of Lemma 1 and add the term $\rho \sum_i \left( A_i \hat{x}_i^k - A_i x_i^* \right)^\top \left( A_i x_i^* - A_i \hat{x}_i^k \right)$ to both sides of inequality (15), which gives us

$$\sum_i \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^\top (\hat{x}_i^k - x_i^*) + \rho \sum_i \left( A_i \hat{x}_i^k - A_i x_i^* \right)^\top \left( A_i x_i^* - A_i \hat{x}_i^k \right) + \frac{1}{\rho} (\lambda^k - \lambda^*)^\top (\lambda^k - \lambda^*) \geq \rho \sum_i \left( A_i \hat{x}_i^k - A_i x_i^* \right)^\top \left( \sum_{j \neq i} (A_j x_j^k - A_j \hat{x}_j^k) \right) + \rho \sum_i \left( A_i \hat{x}_i^k - A_i x_i^* \right)^\top \left( A_i x_i^k - A_i \hat{x}_i^k \right).$$

The aforementioned results suggest that certain heuristics can be used to appropriately tune ADAL. For example, we can perform an online hyper-parameter search by running in parallel multiple instances of ADAL, each one for a different value of $\rho$ and a different initialization, and then selecting the best solution. If running multiple problem instances in parallel is not possible due to limited resources, we can alternatively perform a dynamic-update search where we run one instance of ADAL each time, starting with small values of $\rho$, and increasing $\rho$ if the solution does not yield a reasonable reduction in the constraint violations within a pre-specified number of iterations. Note that the theoretical analysis does not allow for varying $\rho$ during the execution of a single ADAL instance, i.e., if we change $\rho$ between iterations there is no guarantee that ADAL will converge. This is a typical characteristic of all augmented Lagrangian methods, distributed or not.

The aforementioned results suggest that certain heuristics can be used to appropriately tune ADAL. For example, we can perform an online hyper-parameter search by running in parallel multiple instances of ADAL, each one for a different value of $\rho$ and a different initialization, and then selecting the best solution. If running multiple problem instances in parallel is not possible due to limited resources, we can alternatively perform a dynamic-update search where we run one instance of ADAL each time, starting with small values of $\rho$, and increasing $\rho$ if the solution does not yield a reasonable reduction in the constraint violations within a pre-specified number of iterations. Note that the theoretical analysis does not allow for varying $\rho$ during the execution of a single ADAL instance, i.e., if we change $\rho$ between iterations there is no guarantee that ADAL will converge. This is a typical characteristic of all augmented Lagrangian methods, distributed or not.

![Graph showing simulation results for ADAL and the centralized solver applied to problem (43). The results correspond to 50 different choices of the problem parameters $\mathbf{A}$ and $\mathbf{b}$. At each instance, the initialization point is the same for both ADAL and the centralized solver. The red and blue squares indicate the objective function value at the point of convergence for the centralized method and ADAL, respectively. A blue dashed line indicates that ADAL converged to a better (or the same) local minimum, while a red dashed line indicates the opposite.](image-url)
Grouping the terms at the right-hand side of the inequality by their common factor, we transform the estimate as follows:

\[
\sum_i \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^T (\hat{x}_i^k - x_i^*) + \rho \sum_i \left( A_i x_i^k - A_i x_i^* \right)^T (A_i x_i^k - A_i x_i^*) + \frac{1}{\rho} (\lambda^k - \lambda^*)^T (\lambda^k - \lambda^*) \geq \rho \sum_i \left( A_i \hat{x}_i^k - A_i x_i^* \right)^T \left( A_i \hat{x}_i^k - A_i x_i^* \right)
\]

Recall that \( \sum_j A_j (x_j^k - \hat{x}_j^k) = r(x^k) - r(\hat{x}^k) \), which means that this term is a constant factor with respect to the summation over \( i \) in the right-hand side of the previous relation. Moreover, \( \sum_i A_i \hat{x}_i^k - \sum_i A_i x_i^* = \sum_i A_i \hat{x}_i^k - B = r(\hat{x}^k) \). Substituting these terms at the right-hand side of the previous relation, gives us

\[
\sum_i \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^T (\hat{x}_i^k - x_i^*) + \rho \sum_i \left( A_i \hat{x}_i^k - A_i x_i^* \right)^T (A_i \hat{x}_i^k - A_i x_i^*) + \frac{1}{\rho} (\lambda^k - \lambda^*)^T (\lambda^k - \lambda^*) \geq \rho (r(x^k)^T r(x^k) - r(\hat{x}^k)^T r(\hat{x}^k))
\]

(45)

Next, we substitute the expressions

\[
(A_i \hat{x}_i^k - A_i x_i^*) = (A_i x_i^k - A_i x_i^*) + (A_i \hat{x}_i^k - A_i x_i^k)
\]

and \( \lambda^k - \lambda^* = (\lambda^k - \lambda^*) + (\lambda^k - \lambda^*) \),

in the left-hand side of (45). We obtain

\[
\sum_i \left( \nabla f_i(x_i^*) - \nabla f_i(\hat{x}_i^k) \right)^T (\hat{x}_i^k - x_i^*) + \rho \sum_i \left( A_i x_i^k - A_i x_i^* \right)^T (A_i x_i^k - A_i x_i^*) + \frac{1}{\rho} (\lambda^k - \lambda^*)^T (\lambda^k - \lambda^*) \geq \sum_i \rho \|A_i (x_i^k - \hat{x}_i^k)\|^2 + \frac{1}{\rho} \|\lambda^k - \lambda^*\|^2 + (\lambda^k - \lambda^*)^T (r(x^k) - r(\hat{x}^k)).
\]

which completes the proof.

REFERENCES


