Augmented Lagrangian Optimization under Fixed-Point Arithmetic ⋆

Yan Zhang, Michael M. Zavlanos

Mechanical Engineering and Material Science, Duke University, Durham, NC 27708, USA

Abstract

In this paper, we propose an inexact Augmented Lagrangian Method (ALM) for the optimization of convex and nonsmooth objective functions subject to linear equality constraints and box constraints where errors are due to fixed-point data. To prevent data overflow we also introduce a projection operation in the multiplier update. We analyze theoretically the proposed algorithm and provide convergence rate results and bounds on the accuracy of the optimal solution. Since iterative methods are often needed to solve the primal subproblem in ALM, we also propose an early stopping criterion that is simple to implement on embedded platforms, can be used for problems that are not strongly convex, and guarantees the precision of the primal update. To the best of our knowledge, this is the first fixed-point ALM that can handle non-smooth problems, data overflow, and can efficiently and systematically utilize iterative solvers in the primal update. Numerical simulation studies on a logistic regression problem are presented that illustrate the proposed method.

Key words: Convex optimization, Augmented Lagrangian Method, embedded systems, fixed-point arithmetic.

1 Introduction

Embedded computers, such as FPGAs (Field-Programmable Gate Arrays), are typically low-cost, low-power, perform fast computations, and for these reasons they have long been used for the control of systems with fast dynamics and power limitations, e.g., automotive, aerospace, medical, and robotics. While embedded computers have been primarily used for low-level control, they have recently also been suggested to obtain real-time solutions to more complex optimization problems Jerez et al. (2014); Richter et al. (2017). The main challenges in implementing advanced optimization algorithms on resource-limited embedded devices are providing complexity certifications and designing the data precision to control the solution accuracy and the dynamic range to avoid overflow. While recent methods, such as Jerez et al. (2014); Richter et al. (2017) , address these challenges and provide theoretical guarantees, they do so for quadratic problems. In this paper, we provide such guarantees for general convex problems. Specifically, we propose an Augmented Lagrangian Method (ALM) to solve problems of the form

$$\min \quad f(x) \quad \text{s.t.} \quad Ax = b, \quad x \in \mathcal{X} \tag{1}$$

on embedded platforms under fixed-point arithmetic, where $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{p\times n}$, $b \in \mathbb{R}^p$ and $f(x)$ is a scalar-valued function. The set $\mathcal{X}$ is convex. ALM falls in the class of first order methods, which have been demonstrated to be efficient in solving problem (1) on embedded computers due to their simple operations and less memory requirements, Richter et al. (2017).

Recent work on error analysis of inexact first-order methods is presented in Devolder, Glineur & Nesterov (2014); Necoara & Patrascu (2016); Patrinos, Guiggiani & Bemporad (2015). Specifically, Devolder et al. (2014) proposed a first-order inexact oracle to chracterize the iteration complexity and suboptimality of the primal gradient and fast gradient method. Necoara & Patrascu (2016); Patrinos et al. (2015) extend these results to the dual domain. However, these analyses assume strong convexity of the objective functions. Necoara, Patrascu & Glineur (2017); Nedelcu, Necoara & Tran-Dinh (2014) analyze the convergence of ALM using an inexact oracle for general convex problems. The inexactness comes from the approximate solution of the subproblems in ALM. Similar analysis has been conducted in Eckstein & Silva (2013); Lan & Monteiro (2016); Rockafellar (1976b). However, none of above works on ALM consider the error in the multiplier update under fixed-point arithmetic. Moreover, since no projection is used in the mul-

⋆ This work is supported in part by AFOSR under award #FA9550-19-1-0169 and by NSF under award CNS-1932011.

Email addresses: yz227@duke.edu (Yan Zhang), michael.zavlanos@duke.edu (Michael M. Zavlanos).

tiplier update, the above works cannot provide an upper bound on the multiplier iterates and therefore cannot avoid data overflow. Perhaps the most relevant work to the method proposed here is Jerez, Goulart, Richter, Constantinides, Kerrigan & Morari (2014). Specifically, Jerez et al. (2014) analyzes the behavior of the Alternating Direction Method of Multipliers (ADMM), a variant of the ALM, on fixed-point platforms. However, the analysis in Jerez et al. (2014) can only be applied to quadratic objective functions. Moreover, to prevent data overflow, the proposed method needs to monitor the iteration history of the algorithm to estimate a bound on the multiplier iterates.

Compared to existing literature on inexact ALM, we propose a new inexact ALM that incorporates errors in both the primal and dual updates and contains a projection operation in the dual update. Assuming a uniform upper bound on the norm of the optimal multiplier is known, this projection step can guarantee no data overflow during the whole iteration history. To the best of our knowledge, this is the first work to provide such guaranetees for general convex and non-smooth problems. Furthermore, we show that our proposed algorithm has $O(1/K)$ convergence rate and provide bounds on the achievable primal and dual suboptimality and infeasibility. In general, iterative solvers are needed to solve the subproblems in ALM but the theoretical complexity of such solvers is usually conservative. Therefore, we present a stopping criterion that allows us to terminate early the primal iteration in the ALM while guaranteeing the precision of the primal update. This stopping condition is simple to check on embedded platforms and can be used for problems that are not necessarily strongly convex. We note that in this paper we do not provide a theoretical uniform upper bound on the optimal multiplier for general convex problems. Instead, our contribution is to develop a new projected ALM method that relies on such bounds to control data overflow on fixed-point platforms.

The rest of this paper is organized as follows. In Section 2, we formulate the problem and introduce necessary notations and lemmas needed to prove the main results. In Section 3, we characterize the convergence rate of the algorithm and present bounds on the primal suboptimality and infeasiblity of the solution. In Section 4, we present the stopping criterion for the solution of the primal subproblem under fixed-point arithmetic. In Section 5, we present simulations that verify the theoretical analysis in the previous sections. In Section 6, we conclude the paper.

2 Preliminaries

We make the following assumptions on problem (1).

Assumption 2.1 *The function $f(x)$ is convex and is not necessarily differentiable. The problem* (1) *is feasible.*

Algorithm 1 Augmented Lagrangian Method

```
Require: Initialize λ_0 ∈ ℝ^p, k=0
while Ax_k ≠ b do
    x_k = argmin_{x∈𝒳} L_ρ(x; λ_k)
    λ_{k+1} = λ_k + ρ(Ax_k − b)
    k ← k + 1
end while
```

The Lagrangian function of problem (1) is defined as $L(x;\lambda) = f(x) + \langle Ax - b, \lambda\rangle$ and the dual function is defined as $\Phi(\lambda) \triangleq \min_{x\in\mathcal{X}} L(x;\lambda)$, where $\lambda \in \mathbb{R}^n$ is the Lagrangian multiplier (Ruszczyński, 2006) and $\langle\cdot,\cdot\rangle$ is the inner product between two vectors. Then the dual problem associated with problem (1) can be defined as $\max_{\lambda\in\mathbb{R}^p} \Phi(\lambda)$. Suppose $x^\star$ is an optimal solution of the primal problem (1) and $\lambda^\star$ is an optimal solution of the dual problem. Then, we make the following assumption:

Assumption 2.2 *Strong duality holds for the problem* (1). *That is, $f(x^\star) = \Phi(\lambda^\star)$.*

Assumption 2.2 implies that $(x^\star, \lambda^\star)$ is a saddle point of the Lagrangian function $L(x;\lambda)$ (Ruszczyński, 2006). That is, $\forall x \in \mathcal{X}, \lambda \in \mathbb{R}^p$,

$$L(x^\star;\lambda) \le L(x^\star;\lambda^\star) \le L(x;\lambda^\star). \tag{2}$$

The Augmented Lagrangian function of the primal problem (1) is defined by $L_\rho(x;\lambda) = f(x) + \langle Ax - b, \lambda\rangle + \frac{\rho}{2}\|Ax - b\|^2$, where ρ is the penalty parameter and $\|\cdot\|$ is the Euclidean norm of a vector. Moreover, we define the augmented dual function $\Phi_\rho(\lambda) = \min_{x\in\mathcal{X}} L_\rho(x;\lambda)$. Note that $\Phi_\rho(\lambda)$ is always differentiable with respect to λ and its gradient is $\nabla\Phi_\rho(\lambda) = Ax^\star_\lambda - b$, where $x^\star_\lambda = \arg\min_{x\in\mathcal{X}} L_\rho(x;\lambda)$. Moreover, $\nabla\Phi_\rho(\lambda)$ is Lipschitz continuous with constant $L_\Phi = \frac{1}{\rho}$ (Rockafellar, 1976a). The ALM can be viewed as a gradient ascent method on the multiplier λ with step size $\frac{1}{L_\Phi} = \rho$. We present the ALM in Algorithm 1. Discussion on the convergence of Algorithm 1 can be found in Rockafellar (1976a); Ruszczyński (2006) and the references therein. ALM converges faster than dual method when the problem is not strongly convex due to its smoothing effect on the dual objective function $\Phi(\lambda)$, (Rockafellar, 1976a).

In practice, Algorithm 1 cannot be implemented exactly on a fixed-point platform. We present the modified ALM in Algorithm 2 to include fixed-point arithmetic errors. In Algorithm 2, $x^\star_k = \arg\min_{x\in\mathcal{X}} L_\rho(x;\lambda_k)$. The effect of the fixed-point arithmetic is incorporated in the error terms ϵ^k_{in} and ϵ^k_{out}. Moreover, K_{out} is the number of iterations in the ALM and L is the step size used in the dual update and is defined in Lemma 3.2. Finally, D is a convex and compact set containing at least one optimal multiplier $\lambda^\star$ and Π_D denotes the projection onto the set D. This projection step is the major difference that makes the analysis in this paper different from other works on inexact ALM, e.g., Necoara et al. (2017); Nedelcu et al. (2014). The set D is predetermined before running the

Algorithm 2 Augmented Lagrangian Method under inexactness

Require: Initialize $\lambda_0 \in \mathbb{R}^p$, k=0

1: **while** $k \leq K_{out}$ **do**

2: $\quad \tilde{x}_k \approx \arg\min_{x\in\mathcal{X}} L_\rho(x;\lambda_k)$ so that:

$$L_\rho(\tilde{x}_k, \lambda_k) - L_\rho(x_k^\star, \lambda_k) \leq \epsilon_{in}^k$$

3: $\quad \lambda_{k+1} = \Pi_D[\lambda_k + \frac{1}{L}(A\tilde{x}_k - b + \epsilon_{out}^k)]$

4: $\quad k \leftarrow k+1$

5: **end while**

algorithm. We make the following assumption on D:

Assumption 2.3 *The set D is a box that contains 0, $2\lambda^\star$ and $\lambda^\star + \mathbf{1}$, where $\mathbf{1}$ is a vector of appropriate dimension and its entries are all 1.*

The above choice of D is discussed in more details in Section 3. Note that this set D depends on a uniform bound on $\|\lambda^\star\|$ over a set of problem data. The methods proposed in Devolder, Glineur & Nesterov (2012); Mangasarian (1985); Nedić & Ozdaglar (2009); Ruszczyński (2006) establish such bounds on $\|\lambda^\star\|$ for fixed problem parameters. On the other hand, Patrinos & Bemporad (2014); Richter, Morari & Jones (2011) provide uniform bounds on $\|\lambda^\star\|$ assuming $b \in \mathcal{B}$ in the constraints in problem (1). However, the methods in Patrinos & Bemporad (2014); Richter et al. (2011) can only be applied to quadratic problems. For general problems considered in this paper, the interval analysis method in Hansen & Walster (1993) can be used to estimate a uniform bound on $\|\lambda^\star\|$. However, the interval analyisis method requires restrictive assumptions on the set of problem data and gives impractical bounds (Jerez, Constantinides & Kerrigan, 2015). In practice, we can approximate these bounds using sampling and then apply an appropriate scaling factor. Our contribution in this paper is to develop a new projected ALM method that employs such bounds to control data overflow on fixed-point platforms.

3 Convergence Analysis

In this section, we show convergence of Algorithm 2 to a neighborhood of the optimal dual and primal objective value. First, we make some necessary assumptions on the boundedness of errors appearing in the algorithm.

Assumption 3.1 *At every iteration of Algorithm 2, the errors are uniformly bounded, i.e.,*

$$0 \leq \epsilon_{in}^k \leq B_{in}, \quad \|\epsilon_{out}^k\| \leq B_{out}, \text{ for all } k.$$

This assumption is satisfied by selecting appropriate data precision and subproblem solver parameters. Specifically, $\epsilon_{in}^k \geq 0$ if $\tilde{x}_k$ is always feasible, that is, $\tilde{x}_k \in \mathcal{X}$, which is possible if, e.g., the projected gradient method is used to solve the subproblem in line 2 in Algorithm 2. Due to the projection operation in line 3 of Algorithm 2, we have that for $\forall \lambda_1, \lambda_2 \in D_\delta$, $\|\lambda_1 - \lambda_2\| \leq B_\lambda$, where $D_\delta = D \cup \{\lambda_\delta \in \mathbb{R}^m : \lambda_\delta = \lambda + \epsilon_{out}^k, \text{ for all } \lambda \in D\}$. Since D is compact and $\|\epsilon_{out}^k\|$ is bounded according to Assumption 3.1, D_δ is also compact. D_δ is only introduced for the analysis. In practice, we only need to compute the size of D to implement Algorithm 2.

3.1 Inexact Oracle

Consider the concave function $\Phi_\rho(\lambda)$ with L_Φ-Lipschitz continuous gradient. For any λ_1 and $\lambda_2 \in \mathbb{R}^p$, we have $0 \geq \Phi_\rho(\lambda_2) - [\Phi_\rho(\lambda_1) + \langle \nabla\Phi_\rho(\lambda_1), \lambda_2 - \lambda_1\rangle] \geq -\frac{L_\Phi}{2}\|\lambda_1 - \lambda_2\|^2$. Recall the expression of $\nabla\Phi_\rho(\lambda)$. Since step 2 of Algorithm 2 can only be solved approximately, $\nabla\Phi_\rho(\lambda)$ can only be evaluated inexactly. Therefore, the above inequalities can not be satisfied exactly. We extend the results in Devolder et al. (2014); Necoara et al. (2017) and propose the following inexact oracle to include the effect of ϵ_{out}^k.

Lemma 3.2 *(Inexact Oracle) Let assumptions 2.1, 2.2 and 3.1 hold. Moreover, consider the approximations $\Phi_{\delta,L}(\lambda_k) = L_\rho(\tilde{x}_k;\lambda_k) + B_{out}B_\lambda$ to $\Phi_\rho(\lambda_k)$ and $s_{\delta,L}(\lambda_k) = A\tilde{x}_k - b + \epsilon_{out}^k$ to $\nabla\Phi_\rho(\lambda_k)$. Then these approximations consititute a (δ, L) inexact oracle to the concave function $\Phi_\rho(\lambda)$ in the sense that, for $\forall\lambda \in D_\delta$,*

$$\begin{aligned} 0 &\geq \Phi_\rho(\lambda) - (\Phi_{\delta,L}(\lambda_k) + \langle s_{\delta,L}(\lambda_k), \lambda - \lambda_k\rangle) \\ &\geq -\frac{L}{2}\|\lambda - \lambda_k\|^2 - \delta, \end{aligned} \tag{3}$$

where $L = 2L_\Phi = \frac{2}{\rho}$ and $\delta = 2B_{in} + 2B_{out}B_\lambda$.

PROOF. The proof is similar to Necoara et al. (2017) and therefore is omitted.

3.2 Dual Suboptimality

Showing the convergence of the dual variable is similar to showing the convergence of the projected gradient method with the inexact oracle used in Patrinos et al. (2015). Therefore, we omit the proof and directly summarize the dual suboptimality results of our algorithm. Specifically, we have the following inequality:

$$\begin{aligned} &\Phi_\rho(\lambda^\star) - \Phi_\rho(\lambda_{k+1}) \\ &\quad \leq \frac{L}{2}(\|\lambda_k - \lambda^\star\|^2 - \|\lambda_{k+1} - \lambda^\star\|^2) + \delta. \end{aligned} \tag{4}$$

Furthermore, similar to the Theorem 5 in Patrinos et al. (2015), we have the following convergence result for the dual variable:

Theorem 3.3 *Let assumptions 2.1, 2.2 and 3.1 hold. Define $\bar{\lambda}_K = \frac{1}{K}\sum_{k=1}^K \lambda_k$. Then, we have $\Phi_\rho(\lambda^\star) - \Phi_\rho(\bar{\lambda}_k) \leq \frac{L}{2k}\|\lambda_0 - \lambda^\star\|^2 + \delta$.*

3.3 Primal Infeasibility and Suboptimality

Define the Lyapunov/Merit function $\phi^k(\lambda) = \frac{L}{2}\|\lambda_k - \lambda\|^2 + \frac{1}{2}\|\lambda_{k-1} - \lambda^\star\|^2$. Also define the residual function $r(x) = Ax - b$. We have the following intermediate result:

Lemma 3.4 *Let assumptions 2.1,2.2 and 3.1 hold. For all $k \geq 1$, and for all $\lambda \in D$, we have that*

$$f(\tilde{x}_k) - f(x^\star) + \langle \lambda, r(\tilde{x}_k)\rangle \leq \phi^k(\lambda) - \phi^{k+1}(\lambda) + E, \quad (5)$$

where $E = (1+\frac{4}{L})B_\lambda B_{out} + (1+\frac{4}{L})B_{in} + (\frac{1}{2}+\frac{1}{2L})B_{out}^2$.

PROOF. We recall that $\tilde{x}_k$ is a suboptimal solution as defined in line 2 of Algorithm 2 that satisfies $L_\rho(\tilde{x}_k;\lambda_k) - L_\rho(x_k^\star;\lambda_k) \leq \epsilon_{in}^k$. Moreover, due to the optimality of $x_k^\star$, we also have that $L_\rho(x_k^\star;\lambda_k) \leq L_\rho(x^\star;\lambda_k) = f(x^\star)$, where $x^\star$ is the optimal solution to problem (1). The equality is because $r(x^\star) = Ax^\star - b = 0$. Combining these two inequalities, we have that $L_\rho(\tilde{x}_k;\lambda_k) - f(x^\star) \leq \epsilon_{in}^k$. Expanding $L_\rho(\tilde{x}_k;\lambda_k)$ and rearranging terms, we get $f(\tilde{x}_k) - f(x^\star) \leq -\langle \lambda_k, r(\tilde{x}_k)\rangle - \frac{\rho}{2}\|r(\tilde{x}_k)\|^2 + \epsilon_{in}^k$. Adding $\langle \lambda, r(\tilde{x}_k)\rangle$ to both sides of the above inequality, we get

$$\begin{aligned} f(\tilde{x}_k) - f(x^\star) &+ \langle \lambda, r(\tilde{x}_k)\rangle \\ &\leq \langle \lambda - \lambda_k, r(\tilde{x}_k)\rangle - \frac{\rho}{2}\|r(\tilde{x}_k)\|^2 + \epsilon_{in}^k. \end{aligned} \quad (6)$$

In what follows, we show that the right hand side of (6) is upper bounded by $\phi^k(\lambda) - \phi^{k+1}(\lambda) + E$. First, we focus on the term $\langle \lambda - \lambda_k, r(\tilde{x}_k)\rangle$. For all $\lambda \in D$, we have that $\|\lambda_{k+1} - \lambda\|^2 = \|\Pi_D[\lambda_k + \frac{1}{L}(r(\tilde{x}_k) + \epsilon_{out}^k)] - \lambda\|^2 \leq \|\lambda_k + \frac{1}{L}(r(\tilde{x}_k) + \epsilon_{out}^k) - \lambda\|^2 = \|\lambda_k - \lambda\|^2 + 2\langle \frac{1}{L}(r(\tilde{x}_k) + \epsilon_{out}^k), \lambda_k - \lambda\rangle + \frac{1}{L^2}\|r(\tilde{x}_k) + \epsilon_{out}^k\|^2$, where the inequality follows from the contraction of the projection onto a convex set. Rearranging terms in the above inequality and multiplying both sides by $\frac{L}{2}$, we have $\langle r(\tilde{x}_k), \lambda - \lambda_k\rangle \leq \frac{L}{2}(\|\lambda_k - \lambda\|^2 - \|\lambda_{k+1} - \lambda\|^2) - \langle \epsilon_{out}^k, \lambda - \lambda_k\rangle + \frac{1}{2L}\|r(\tilde{x}_k)\|^2 + \frac{1}{L}\langle r(\tilde{x}_k), \epsilon_{out}^k\rangle + \frac{1}{2L}\|\epsilon_{out}^k\|^2$. Applying Assumption 3.1 and the Cauchy-Schwartz inequality, we obtain

$$\begin{aligned} \langle r(\tilde{x}_k), \lambda - \lambda_k\rangle &\leq \frac{L}{2}(\|\lambda_k - \lambda\|^2 - \|\lambda_{k+1} - \lambda\|^2) \\ &+ B_{out}B_\lambda + \frac{1}{2L}\|r(\tilde{x}_k)\|^2 + \frac{1}{L}\langle r(\tilde{x}_k), \epsilon_{out}^k\rangle + \frac{1}{2L}B_{out}^2. \end{aligned} \quad (7)$$

To upper bound the term $\langle r(\tilde{x}_k), \epsilon_{out}^k\rangle$ in (7), first we add and subtract ϵ_{out}^k from $r(\tilde{x}_k)$, and rearrange terms to get $\langle r(\tilde{x}_k), \epsilon_{out}^k\rangle = \langle r(\tilde{x}_k) + \epsilon_{out}^k, \epsilon_{out}^k\rangle - \|\epsilon_{out}^k\|^2$. Defining $\lambda_{\delta k} = \lambda_k + \epsilon_{out}^k$, recalling the definition of $s_{\delta,L}(\lambda_k)$ in Lemma 3.2, and ignoring the term $-\|\epsilon_{out}^k\|^2$, we obtain $\langle r(\tilde{x}_k), \epsilon_{out}^k\rangle \leq \langle s_{\delta,L}(\lambda_k), \lambda_{\delta k} - \lambda_k\rangle$. Next we apply the second inequality in Lemma 3.2 to upper bound $\langle s_{\delta,L}(\lambda_k), \lambda_{\delta k} - \lambda_k\rangle$. In order to apply Lemma 3.2, both $\lambda_{\delta k}$ and λ_k need to belong to D_δ. Due to the projection in line 3 in Algorithm 2, λ_k always belongs to D. Recalling the definition of D_δ, it is straightforward to verify that λ_k, $\lambda_{\delta k} \in D_\delta$. Thus applying Lemma 3.2 we get

$$\begin{aligned} \langle r(\tilde{x}_k), \epsilon_{out}^k\rangle &\leq \langle s_{\delta,L}(\lambda_k), \lambda_{\delta k} - \lambda_k\rangle \\ &\leq \Phi_\rho(\lambda_{\delta k}) - \Phi_{\delta,L}(\lambda_k) + \frac{L}{2}B_{out}^2 + \delta. \end{aligned} \quad (8)$$

Since $\lambda^\star$ is the global maximizer of the function $\Phi_\rho(\lambda)$, we get $\Phi_\rho(\lambda^\star) \geq \Phi_\rho(\lambda_{\delta k})$. We can also show that $\Phi_{\delta,L}(\lambda_k) \geq \Phi_\rho(\lambda_k)$ always holds because $\Phi_{\delta,L}(\lambda_k) = L_\rho(\tilde{x}_k;\lambda_k) + B_{out}B_\lambda \geq L_\rho(\tilde{x}_k;\lambda_k) \geq L_\rho(x_k^\star;\lambda_k)$. Combining these two inequalities we obtain that $\Phi_\rho(\lambda^\star) - \Phi_\rho(\lambda_k) \geq \Phi_\rho(\lambda_{\delta k}) - \Phi_{\delta,L}(\lambda_k)$. Substituting this inequality into (8), we have that $\langle r(\tilde{x}_k), \epsilon_{out}^k\rangle \leq \Phi_\rho(\lambda^\star) - \Phi_\rho(\lambda_k) + \frac{L}{2}B_{out}^2 + \delta$. Combining this inequality, (4) and (7), we get

$$\begin{aligned} \langle r(\tilde{x}_k), \lambda - \lambda_k\rangle &\leq \frac{L}{2}(\|\lambda_k - \lambda\|^2 - \|\lambda_{k+1} - \lambda\|^2) \\ &+ \frac{1}{2}(\|\lambda_{k-1} - \lambda^\star\|^2 - \|\lambda_k - \lambda^\star\|^2) + \frac{1}{2L}\|r(\tilde{x}_k)\|^2 \\ &+ B_{out}B_\lambda + (\frac{1}{2} + \frac{1}{2L})B_{out}^2 + \frac{2}{L}\delta. \end{aligned} \quad (9)$$

Combining (9) with (6), we have that $f(\tilde{x}_k) - f(x^\star) + \langle \lambda, r(\tilde{x}_k)\rangle \leq \frac{L}{2}(\|\lambda_k - \lambda\|^2 - \|\lambda_{k+1} - \lambda\|^2) + \frac{1}{2}(\|\lambda_{k-1} - \lambda^\star\|^2 - \|\lambda_k - \lambda^\star\|^2) + (\frac{1}{2L} - \frac{\rho}{2})\|r(\tilde{x}_k)\|^2 + B_{out}B_\lambda + (\frac{1}{2} + \frac{1}{2L})B_{out}^2 + \frac{2}{L}\delta + \epsilon_{in}^k$. From Lemma 3.2, we have that $L = 2L_\Phi = \frac{2}{\rho}$, and $\frac{1}{2L} - \frac{\rho}{2} < 0$. Therefore, the term $(\frac{1}{2L} - \frac{\rho}{2})\|r(\tilde{x}_k)\|^2 < 0$ can be neglected. Recalling the definition of $\phi^k(\lambda)$, δ and E, we obtain (5), which completes the proof. □

Next, we apply Lemma 3.4 to prove the primal suboptimality and infeasibility of Algorithm 2.

Theorem 3.5 *(Primal Suboptimality and Infeasibility) Let assumptions 2.1, 2.2 and 3.1 hold. Define $\bar{x}_K = \frac{1}{K}\sum_{k=1}^K \tilde{x}_k$. Then, we have that (a) primal optimality:*

$$-(\frac{1}{K}\phi^1(2\lambda^\star) + E) \leq f(\bar{x}_K) - f(x^\star) \leq \frac{1}{K}\phi^1(0) + E, \quad (10)$$

(b) primal feasibility:

$$\|r(\bar{x}_K)\| \leq \frac{1}{K}\phi^1(\lambda^\star + \frac{r(\bar{x}_K)}{\|r(\bar{x}_K)\|}) + E. \quad (11)$$

PROOF. Summing inequality (5) in Lemma 3.4 for $k = 1, 2, \ldots, K$, we have that $\sum_{k=1}^K f(\tilde{x}_k) - Kf(x^\star) + \langle \lambda, \sum_{k=1}^K r(\tilde{x}_k)\rangle \leq \phi^1(\lambda) - \phi^{K+1}(\lambda) + KE \leq \phi^1(\lambda) + KE$, where the second inequality follows from $\phi^k(\lambda) \geq 0$. Dividing both sides of the above inequality by K and using the fact that $\frac{1}{K}\sum_{k=1}^K r(\tilde{x}_k) = r(\bar{x}_K)$, we get $\frac{1}{K}\sum_{k=1}^K f(\tilde{x}_k) - f(x^\star) + \langle \lambda, r(\bar{x}_K)\rangle \leq \frac{1}{K}\phi^1(\lambda) + E$. From the convexity of the function $f(x)$, we have that $f(\bar{x}_K) \leq \frac{1}{K}\sum_{k=1}^K f(\tilde{x}_k)$. Combining the above two inequalities, we obtain

$$f(\bar{x}_K) - f(x^\star) + \langle \lambda, r(\bar{x}_K)\rangle \leq \frac{1}{K}\phi^1(\lambda) + E. \quad (12)$$

To prove the right inequality in (10), let $\lambda = 0$ in (12). Then, we have that $f(\bar{x}_K) - f(x^\star) \le \frac{1}{K}\phi^1(0) + E$. To prove the left inequality in (10), according to the inequality (2) and Assumption 2.2, we have that

$$f(x^\star) - f(\bar{x}_K) \le \langle \lambda^\star, r(\bar{x}_K)\rangle. \quad (13)$$

Adding $\langle \lambda^\star, r(\bar{x}_K)\rangle$ to both sides of (13) and rearranging terms, we have that

$$\langle \lambda^\star, r(\bar{x}_K)\rangle \le f(\bar{x}_K) - f(x^\star) + \langle 2\lambda^\star, r(\bar{x}_K)\rangle. \quad (14)$$

Combining inequalities (13) and (14), we obtain $f(x^\star) - f(\bar{x}_K) \le f(\bar{x}_K) - f(x^\star) + \langle 2\lambda^\star, r(\bar{x}_K)\rangle$. Combining this inequality and the inequality in (12) with $\lambda = 2\lambda^\star$, we get $f(x^\star) - f(\bar{x}_K) \le \frac{1}{K}\phi^1(2\lambda^\star) + E$.

To prove (11), let $\lambda = \lambda^\star + \frac{r(\bar{x}_K)}{\|r(\bar{x}_K)\|}$ in (12), which also belongs in D by Assumption 2.3. Rearranging terms, we obtain $f(\bar{x}_K) - f(x^\star) + \langle \lambda^\star, r(\bar{x}_K)\rangle + \|r(\bar{x}_K)\| \le \frac{1}{K}\phi^1(\lambda^\star + \frac{r(\bar{x}_K)}{\|r(\bar{x}_K)\|}) + E$. Letting $x = \bar{x}_K$ in the second inequality in (2), we have that $f(\bar{x}_K) + \langle \lambda^\star, r(\bar{x}_K)\rangle - f(x^\star) \ge 0$. Combining this inequality with the above inequality, we get $\|r(\bar{x}_K)\| \le \frac{1}{K}\phi^1(\lambda^\star + \frac{r(\bar{x}_K)}{\|r(\bar{x}_K)\|}) + E$, which completes the proof. □

4 Fixed-point Implementation

To apply the Algorithm 2 to solve the problem (1) on fixed-point platforms, we need another assumption.

Assumption 4.1 *The set $\mathcal{X}$ is compact and simple.*

The compactness is needed to bound the primal iterates $\tilde{x}_k$ in Algorithm 2. $\mathcal{X}$ is simple in the sense that computing the projection onto $\mathcal{X}$ is of low complexity on fixed-point platforms, for example, $\mathcal{X}$ is a box. We note that our convergence analysis in Section 3 does not rely on the Assumption 4.1. This assumption is needed only for fixed-point implementation. Due to the projection operation in the multiplier update in Algorithm 2, $\|\lambda_k\|$ is always bounded. Since now both x_k and λ_k lie in compact sets, the bounds on the norms of these variables can be used to design the word length of the fixed-point data to avoid overflow, similar to Section 5.3 in Patrinos et al. (2015). Meanwhile, according to the results in (10) and (11), we can choose the number K_{out} of the outer iterations, the accuracy of the multiplier update B_{out} and the accuracy of the subproblem solution B_{in} in Algorithm 2 to achieve an ϵ-solution to the problem (1).

Specifically, to achieve the accuracy B_{in}, we can determine the iteration complexity of the subproblem solver based on the studies in Devolder et al. (2014); Patrinos et al. (2015); Schmidt et al. (2011). However, these results are usually conservative in practice. Therefore, in this section, we propose a stopping criterion for early termination of the subproblem solver with guarantees on the solution accuracy. This stopping criterion can be used for box-constrained convex optimization problems that are not necessarily strongly convex, and is simple to check on embedded devices. A similar stopping criterion has been proposed in Nedelcu et al. (2014), which however requires the objective function of the subproblem to be strongly convex. In what follows, we relax the strong convexity assumption so that this stopping criterion can be applied.

Lemma 4.2 *Consider a convex and nonsmooth function $f(x)$, with support $\mathcal{X} \subset \mathbb{R}^n$ and set of minimizers $X^\star$. Assume $f(x)$ satisfies the quadratic growth condition,*

$$f(x) \ge f^\star + \frac{\sigma}{2}\mathtt{dist}^2(x, X^\star), \text{ for } \forall x \in \mathcal{X}, \quad (15)$$

where $\mathtt{dist}(x, X^\star)$ is the distance from x to the set $X^\star$. and $\sigma > 0$ is a scalar. Then, we have that for any $x \in \mathcal{X}$ and any $s \in \mathcal{N}_{\mathcal{X}}(x)$,

$$f(x) - f^\star \le \frac{2}{\sigma}\|\partial f(x) + s\|^2, \quad (16)$$

where $\mathcal{N}_{\mathcal{X}}(x)$ is the normal cone at x with respect to $\mathcal{X}$.

PROOF. The proof is the similar to the proof in Nedelcu et al. (2014) and therefore is omitted. □

Necoara, Nesterov & Glineur (2018) have shown that the function $h(Ax)$ satisfies the inequality (15) when the set $\mathcal{X}$ is polyhedral and $h(y)$ is strongly convex. However, since the subproblem objective $L_\rho(x, \lambda_k)$ is in the form $\sum_i h_i(A_i x)$ rather than $h(Ax)$, in what follows we present a generalization of Theorem 8 in Necoara et al. (2018) so that the function $L_\rho(x, \lambda_k)$ satisfies the inequality (15) and the condition (16) can be applied.

Lemma 4.3 *Consider a convex function $H(x) = \sum_i h_i(A_i x - b_i)$ with the polyhedral support $\mathcal{X} = \{x : Cx \le d\}$, where $h_i(y)$ is σ_i-strongly convex with respect to y for any i. Then we have*

$$H(x) \ge H^\star + \frac{\sigma}{2}\mathtt{dist}^2(x, X^\star), \quad (17)$$

where $\sigma = \frac{\min_i\{\sigma_i\}}{\theta^2(\tilde{A}, C)}$, the matrix $\tilde{A}$ is in the form $[\cdots|A_i^T|\cdots]^T$ and $\theta(\tilde{A}, C)$ is a constant only related to the matrices $\{A_i\}$ and C.

PROOF. Given $x^\star \in X^\star$, for any $x \in \mathcal{X}$, from the strong convexity of the function h_i, we have that $h_i(A_i x - b_i) \ge h_i(A_i x^\star - b_i) + \langle \partial h_i(y)|_{y=A_i x^\star - b_i}, A_i(x - x^\star)\rangle + \frac{\sigma_i}{2}\|A_i x - A_i x^\star\|^2 = h_i(A_i x^\star - b_i) + \langle x - x^\star, A_i^T \partial h_i(y)|_{y=A_i x^\star - b_i}\rangle + \frac{\sigma_i}{2}(x - x^\star)^T A_i^T A_i (x - x^\star)$. Adding up the above inequalities for all i, we have $H(x) \ge H(x^\star) + \langle \sum_i A_i^T \partial h_i(y)|_{y=A_i x^\star - b_i}, x - x^\star\rangle + \sum_i \frac{\sigma_i}{2}(x - x^\star)^T A_i^T A_i (x - x^\star)$. Recall that $\partial H(x^\star) = \sum_i A_i^T \partial h_i(y)|_{y=A_i x^\star - b_i}$. From the optimality of $x^\star$, we have that $\langle \sum_i A_i^T \partial h_i(y)|_{y=A_i x^\star - b_i}, x - x^\star\rangle \ge 0$ for any $x \in \mathcal{X}$. Therefore, we obtain

$$H(x) \ge H(x^\star) + \sum_i \frac{\sigma_i}{2}(x - x^\star)^T A_i^T A_i (x - x^\star). \quad (18)$$

Next, we show that the optimal solution set $X^\star = \{\tilde{x}^\star : \tilde{A}\tilde{x}^\star = t^\star; C\tilde{x}^\star \le d\}$, where $t^\star = \tilde{A}x^\star$. To do so, we decompose the space $\mathbb{R}^n$ into two mutually orthogonal subspaces, $\mathcal{V}$ and $\mathcal{U}$, where $\mathcal{V}$ is the intersection of the kernel spaces of all matrices A_i and $\mathcal{U}$ is the union of the row space of all matrices A_i. Showing that $X^\star = \{\tilde{x}^\star : \tilde{A}\tilde{x}^\star = t^\star; C\tilde{x}^\star \le d\}$ is equivalent as showing that $X^\star = \tilde{X}^\star$, where $\tilde{X}^\star = \{\tilde{x}^\star : \tilde{x}^\star = x^\star + v$, where $v \in \mathcal{V}; C\tilde{x}^\star \le d\}$. First, we show that $\tilde{X}^\star \subset X^\star$. If $\tilde{x}^\star \in \tilde{X}^\star$, we have that $\tilde{x}^\star$ is feasible and $H(\tilde{x}^\star) = H(x^\star)$ because $A_i\tilde{x}^\star = A_i(x^\star + v) = A_i x^\star$, for any i. Therefore, $\tilde{X}^\star \subset X^\star$. Second, we prove that $X^\star \subset \tilde{X}^\star$ by showing that if $\tilde{x} \notin \tilde{X}^\star$, then $\tilde{x}$ is not optimal. If $C\tilde{x} > d$, then $\tilde{x}$ is not feasible. On the other hand, if $C\tilde{x} \le d$ but $\tilde{x} = x^\star + \alpha v + u$, where α is any real number and $u \in \mathcal{U}$, then $H(\tilde{x}) > H(x^\star)$. This is due to the inequality (18) and $\sum_i \frac{\sigma_i}{2}(\tilde{x} - x^\star)^T A_i^T A_i(\tilde{x} - x^\star) > 0$. Therefore, $X^\star \subset \tilde{X}^\star$ and we have shown that $X^\star = \tilde{X}^\star$. Since $X^\star = \{\tilde{x}^\star : \tilde{A}\tilde{x}^\star = t^\star; C\tilde{x}^\star \le d\}$, according to Lemma 15 in Wang & Lin (2014), for any $x \in \mathcal{X}$, we have that $\mathtt{dist}(x, X^\star) \le \theta(\tilde{A}, C)\|\tilde{A}(x - x^\star)\|$. Furthermore, we have that $\mathtt{dist}^2(x, X^\star) \le \theta^2(\tilde{A}, C)\|\tilde{A}(x - x^\star)\|^2 \le \frac{\theta^2(\tilde{A},C)}{\min_i\{\sigma_i\}}\sum_i \sigma_i(x - x^\star)^T A_i^T A_i(x - x^\star)$. Combining this inequality with (18), we get the desired result (17). □

The constant $\theta(\tilde{A}, C)$ can be obtained as in Necoara, Nesterov & Glineur (2018). If the objective function $f(x)$ in problem (1) is of the form $h(Ax - b)$ where $h(y)$ is strongly convex and the set $\mathcal{X} = \{x : l \le x \le u\}$, according to Lemma 4.3, the subproblem objective $L_\rho(x, \lambda_k)$ satisfies the inequality (15). Therefore, according to Lemma 4.2, we have that the iterate x_t returned from the subproblem solver satisfies $L_\rho(x_t; \lambda_k) - L_\rho(x_k^\star; \lambda_k) \le B_{in}$ if $\|\partial L_\rho(x_t; \lambda_k) + s_t^\star\| \le \sqrt{\frac{\sigma}{2}B_{in}}$, where $s_t^\star = \arg\min_{s_t \in \mathcal{N}_\mathcal{X}(x_t)} \|\partial L_\rho(x_t; \lambda_k) + s_t\|$. As discussed in Nedelcu et al. (2014), $\|\partial L_\rho(x_t; \lambda_k) + s_t^\star\|$ can be efficiently evaluated on the embedded platform.

5 Numerical Simulations

In this section, we present simulation results for a fairness constrained logistic regression problem to verify the theoretical results in Section 3 and the fixed-point implementation in Section 4. The simulations are conducted using the Fixed-Point Designer in Matlab R2020a on a Macbook Pro with 2.3GHz Quad-Core Intel Core i5. Specifically, we consider the dataset $\{d_i, z_i, y_i\}_{i=1:N}$, where $d_i \in \mathbb{R}^n$ is a vector of features at sample i, z_i is a sensitive attribute associated with sample i, e.g., gender or age, and $y_i = \{-1, +1\}$ is the corresponding label. As shown in Zafar et al. (2019), to find a classifier with parameter x that can both predict the label for a sample with feature d_i and mitigate the correlation between the sample's attribute and label, we can formulate the following optimization problem

$$\min_{x,c} \; -\frac{1}{N}\sum_{i=1}^{N} \log\big(\Pr(y_i|d_i, x)\big) \tag{19}$$

$$\text{s.t.} \;\; \frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})d_i^T x = c,\; x_l \le x \le x_u,\; c_l \le c \le c_u,$$

where $\Pr(y_i|d_i, x) = 1/\big(1+\exp(-y_i d_i^T x)\big)$, $\bar{z}$ is the mean of the attributes in the dataset, and c represents the covariance between the labels and the attributes. In addition, x_l, x_u, c_l and c_u denote the bounds on these decision variables. [1] Note that problem (19) cannot be directly solved using the results in Jerez et al. (2014), which assume a quadratic objective function. The method in Jerez et al. (2014) can be extended to solve Problem (19), if combined with Sequential Quadratic Programming (SQP). However, doing so requires the inversion of a new KKT matrix at each iteration of SQP and the solutions to the QP problems will be inexact on a fixed-point platform. The effect of this inexactness on the iteration complexity and final solution accuracy of SQP under fixed-point arithmetics has not been theoretically studied.

Table 1. Primal optimality and feasiblity achieved by Algorithm 2 for Problem (19)

ϵ	*fl-wl*	low. (10),	opt.,	up. (10)	feas.,	(11)
1	15-19	-0.7008,	0.0218,	0.6608	0.0198,	0.9633
0.1	19-23	-0.0715,	0.0071,	0.0672	0.0019,	0.0996
0.01	22-26	-0.0072,	0.0045,	0.0067	0.0004,	0.0100

For our numerical simulations, we generate 10 datasets $\{d_i, z_i, y_i\}$. Each dataset contains 100 samples. The feature vectors $d_i \in \mathbb{R}^5$ and the true decision parameter $x^* \in \mathbb{R}^5$ in each dataset are sampled from a uniform distribution $[-1, 1]^5$. For a sample with the feature vector d_i, the corresponding label is $y_i = +1$ if $d_i^T x \ge 0$. Each sample is associated with a binary attribute $z_i = \{+1, -1\}$ with probability $(0.5, 0.5)$. The bounds on $\|\lambda^\star\|$ for each problem instance were obtained by solving this problem using the Matlab function *fmincon*. Table 1 shows the achieved optimality and feasibility for the worst-case scenario and for three desired solution accuracies, $\epsilon = 1, 0.1$ and 0.01. The fractional and word lengths in Table 1 are chosen using the bounds (10) and (11) in Theorem 3.5. In Table 1, we observe that the optimality achieved by Algorithm 2 is bounded by the lower and upper bounds in (10), and the achieved feasibility is upper bounded by the bound in (11). Finally, notice

[1] The logarithmic objectives as in (19) also appear in networked control problems studied, e.g., in Chang et al. (2015); Chatzipanagiotis et al. (2015); Chatzipanagiotis & Zavlanos (2016, 2017); Lee et al. (2018). In these networked control problems, fixed-point methods can improve the efficiency of solving the local subproblems as well as quantize the information shared among the agents to reduce communication overhead while maintaining the solution accuracy. Extension of the proposed fixed-point optimization method to this setting is part of our future research agenda.

that the bounds in (10) and (11) are at most 50 times larger than the actual algorithm performance. This suggests that the bounds can be used in practice to design the proposed fixed-point implementation.

6 Conclusion

In this paper we proposed an Augmented Lagrangian Method to solve convex and non-smooth optimization problems using fixed-point arithmetic. To avoid data overflow, we introduced a projection operation in the multiplier update. Moreover, we present a stopping criterion to terminate early the primal subproblem iteration, while ensuring a desired accuracy of the solution. We presented convergence rate results as well as bounds on the optimality and feasibility gaps. To the best of our knowledge, this is the first fixed-point ALM that can handle non-smooth problems, data overflow, and can efficiently and systematically utilize iterative solvers in the primal update.

References

Chang, T.-H., Hong, M., & Wang, X. (2015). Multi-agent distributed optimization via inexact consensus ADMM. *IEEE Transactions on Signal Processing*, *63*, 482–497.

Chatzipanagiotis, N., Dentcheva, D., & Zavlanos, M. M. (2015). An augmented Lagrangian method for distributed optimization. *Mathematical Programming*, *152*, 405–434.

Chatzipanagiotis, N., & Zavlanos, M. M. (2016). A distributed algorithm for convex constrained optimization under noise. *IEEE Transactions on Automatic Control*, *61*, 2496–2511.

Chatzipanagiotis, N., & Zavlanos, M. M. (2017). On the convergence of a distributed augmented Lagrangian method for nonconvex optimization. *IEEE Transactions on Automatic Control*, *62*, 4405–4420.

Devolder, O., Glineur, F., & Nesterov, Y. (2012). Double smoothing technique for large-scale linearly constrained convex optimization. *SIAM Journal on Optimization*, *22*, 702–727.

Devolder, O., Glineur, F., & Nesterov, Y. (2014). First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, *146*, 37–75.

Eckstein, J., & Silva, P. J. (2013). A practical relative error criterion for augmented Lagrangians. *Mathematical Programming*, *141*, 319–348.

Hansen, E., & Walster, G. (1993). Bounds for Lagrange multipliers and optimal points. *Computers & Mathematics with Applications*, *25*, 59–69.

Jerez, J. L., Constantinides, G. A., & Kerrigan, E. C. (2015). A low complexity scaling method for the Lanczos kernel in fixed-point arithmetic. *IEEE Transactions on Computers*, *64*, 303–315.

Jerez, J. L., Goulart, P. J., Richter, S., Constantinides, G. A., Kerrigan, E. C., & Morari, M. (2014). Embedded online optimization for model predictive control at megahertz rates. *IEEE Transactions on Automatic Control*, *59*, 3238–3251.

Lan, G., & Monteiro, R. D. (2016). Iteration-complexity of first-order augmented Lagrangian methods for convex programming. *Mathematical Programming*, *155*, 511–547.

Lee, S., Chatzipanagiotis, N., & Zavlanos, M. M. (2018). Complexity certification of a distributed augmented Lagrangian method. *IEEE Transactions on Automatic Control*, *63*, 827–834.

Mangasarian, O. (1985). Computable numerical bounds for Lagrange multipliers of stationary points of non-convex differentiable non-linear programs. *Operations Research Letters*, *4*, 47–48.

Necoara, I., Nesterov, Y., & Glineur, F. (2018). Linear convergence of first order methods for non-strongly convex optimization. *Mathematical Programming*, (pp. 1–39).

Necoara, I., & Patrascu, A. (2016). Iteration complexity analysis of dual first-order methods for conic convex programming. *Optimization Methods and Software*, *31*, 645–678.

Necoara, I., Patrascu, A., & Glineur, F. (2017). Complexity of first-order inexact Lagrangian and penalty methods for conic convex programming. *Optimization Methods and Software*, (pp. 1–31).

Nedelcu, V., Necoara, I., & Tran-Dinh, Q. (2014). Computational complexity of inexact gradient augmented Lagrangian methods: application to constrained MPC. *SIAM Journal on Control and Optimization*, *52*, 3109–3134.

Nedić, A., & Ozdaglar, A. (2009). Approximate primal solutions and rate analysis for dual subgradient methods. *SIAM Journal on Optimization*, *19*, 1757–1780.

Patrinos, P., & Bemporad, A. (2014). An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *IEEE Transactions on Automatic Control*, *59*, 18–33.

Patrinos, P., Guiggiani, A., & Bemporad, A. (2015). A dual gradient-projection algorithm for model predictive control in fixed-point arithmetic. *Automatica*, *55*, 226–235.

Richter, S., Geyer, T., & Morari, M. (2017). Resource-efficient gradient methods for model predictive pulse pattern control on an FPGA. *IEEE Transactions on Control Systems Technology*, *25*, 828–841.

Richter, S., Morari, M., & Jones, C. N. (2011). Towards computational complexity certification for constrained MPC based on Lagrange relaxation and the fast gradient method. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on* (pp. 5223–5229). IEEE.

Rockafellar, R. T. (1976a). Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of operations research*, *1*, 97–116.

Rockafellar, R. T. (1976b). Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, *14*, 877–898.

Ruszczyński, A. (2006). Nonlinear optimization. *Princeton University Press, Princeton*, .

Schmidt, M., Roux, N. L., & Bach, F. R. (2011). Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in neural information processing systems* (pp. 1458–1466).

Wang, P., & Lin, C. (2014). Iteration complexity of feasible descent methods for convex optimization. *The Journal of Machine Learning Research*, *15*, 1523–1548.

Zafar, M. B., Valera, I., Gomez-Rodriguez, M., & Gummadi, K. P. (2019). Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, *20*, 1–42.