On the Convergence Time of the Drift-Plus-Penalty Algorithm for Strongly Convex Programs

Hao Yu and Michael J. Neely

***Abstract*— This paper studies the convergence time of the drift-plus-penalty algorithm for strongly convex programs. The drift-plus-penalty algorithm was originally developed to solve more general stochastic optimization and is closely related to the dual subgradient algorithm when applied to deterministic convex programs. For general convex programs, the convergence time of the drift-plus-penalty algorithm is known to be $O(\frac{1}{\epsilon^2})$. This paper shows that the convergence time for general strongly convex programs is $O(\frac{1}{\epsilon})$. This paper also proposes a new variation of the drift-plus-penalty algorithm, the drift-plus-penalty algorithm with shifted running averages, and shows that if the dual function of the strongly convex program is smooth and locally quadratic then the convergence time of the new algorithm is $O(\frac{1}{\epsilon^{2/3}})$. The convergence time analysis is further verified by numerical experiments.**

I. Introduction

Consider the following strongly convex program:

$$\min \quad f(\mathbf{x}) \tag{1}$$
$$\text{s.t.} \quad g_k(\mathbf{x}) \leq 0, \forall k \in \{1,2,\ldots,m\} \tag{2}$$
$$\mathbf{x} \in \mathcal{X} \tag{3}$$

where set $\mathcal{X} \subseteq R^n$ is closed and convex; function $f(\mathbf{x})$ is continuous and strongly convex on $\mathcal{X}$; functions $g_k(\mathbf{x}), \forall k \in \{1,2,\ldots,m\}$ are Lipschitz continuous and convex on $\mathcal{X}$. Denote $\mathbf{g}(\mathbf{x}) = [g_1(\mathbf{x}),\ldots,g_m(\mathbf{x})]^T$. The minimum of the problem (1)-(3) is assumed to exist.

A. The ϵ-Optimal Solution

Let $\mathbf{x}^*$ be an optimal solution to problem (1)-(3). For any $\epsilon > 0$, a point $\mathbf{x}^\epsilon \in \mathcal{X}$ is said to be an *ϵ-optimal solution* if $f(\mathbf{x}^\epsilon) \leq f(\mathbf{x}^*) + \epsilon$ and $g_k(\mathbf{x}^\epsilon) \leq \epsilon, \forall k \in \{1,\ldots,m\}$. Let $\mathbf{x}(t), t \in \{1,2,\ldots\}$ be the solution sequence yielded by an iterative algorithm. The *convergence time* of this algorithm is said to be $O(h(\epsilon))$ if $\mathbf{x}(t), \forall t \geq O(h(\epsilon))$ is a sequence of ϵ-optimal solutions. It is immediate that if $\tilde{h}(t) : \mathbb{R}_+ \to \mathbb{R}_+$ is a decreasing and invertible function and $f(\mathbf{x}(t)) \leq f(\mathbf{x}^*) + \tilde{h}(t)$ and $g_k(\mathbf{x}(t)) \leq \tilde{h}(t), \forall k \in \{1,\ldots,m\}$ for all $t \geq 1$, then the convergence time is $O(\tilde{h}^{-1}(\epsilon))$. For example, if the solution sequence $\mathbf{x}(t), t \in \{1,2,\ldots\}$ yielded by an iterative algorithm satisfies $f(\mathbf{x}(t)) \leq f(\mathbf{x}^*) + \frac{1}{\sqrt{t}}$ and $g_k(\mathbf{x}(t)) \leq \frac{1}{\sqrt{t}}, \forall k \in \{1,\ldots,m\}$ for all $t \geq 1$, then error decays with time like $O(\frac{1}{\sqrt{t}})$ and the convergence time of this algorithm is $O(\frac{1}{\epsilon^2})$.

This work is supported in part by the NSF grant CCF-0747525.

The authors are with the Electrical Engineering department at the University of Southern California, Los Angeles, CA.

B. The Drift-Plus-Penalty Algorithm

The *drift-plus-penalty* algorithm in [1] can be used to solve problem (1)-(3) as follows:

Algorithm 1: [The Drift-Plus-Penalty Algorithm] Let $V > 0$ be a constant parameter. Let $Q_k(0) \geq 0, \forall k \in \{1,2,\ldots,m\}$ be given constants. At each iteration t, update $\mathbf{x}(t)$, $Q_k(t+1), k \in \{1,2,\ldots,m\}$ and $\overline{\mathbf{x}}(t)$ as follows:

- $\mathbf{x}(t) = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmin}} \Big[Vf(\mathbf{x}) + \sum_{k=1}^{m} Q_k(t) g_k(\mathbf{x})\Big]$.
- $Q_k(t+1) = \max\Big\{Q_k(t) + g_k(\mathbf{x}(t)), 0\Big\}, \forall k \in \{1,2,\ldots,m\}$.
- $\overline{\mathbf{x}}(t+1) = \frac{1}{t+1}\sum_{\tau=0}^{t} \mathbf{x}(t) = \overline{\mathbf{x}}(t)\frac{t}{t+1} + \mathbf{x}(t)\frac{1}{t+1}$.

The drift-plus-penalty algorithm introduces a virtual queue $Q_k(t)$ for each constraint $g_k(\mathbf{x})$ in (2) defined as $Q_k(t+1) = \max\{Q_k(t) + g_k(\mathbf{x}(t)), 0\}$. Note that $\overline{\mathbf{x}}(t)$ is the running average of the sequence $\mathbf{x}(\tau), \tau \in \{0,1,2,\ldots,t-1\}$.

C. The Dual Subgradient Algorithm

The drift-plus-penalty algorithm was originally developed to solve more general stochastic optimization and was shown applicable to deterministic convex programs [2]. It was noted in [3], [4] that the drift-plus-penalty algorithm for convex programs is closely related to the dual subgradient algorithm with the averaged primal sequence. The classical dual subgradient algorithm for problem (1)-(3) is as follows:

Algorithm 2: [The Dual Subgradient Algorithm]Let $c > 0$ be a constant step size. Let $\lambda_k(0) \geq 0, \forall k \in \{1,2,\ldots,m\}$ be given constants. At each iteration t, update $\mathbf{x}(t)$ and $\lambda_k(t+1), k \in \{1,2,\ldots,m\}$ as follows:

- $\mathbf{x}(t) = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmin}} \Big[f(\mathbf{x}) + \sum_{k=1}^{m} \lambda_k(t) g_k(\mathbf{x})\Big]$.
- $\lambda_k(t+1) = \max\Big\{\lambda_k(t) + cg_k(\mathbf{x}(t)), 0\Big\}, \forall k \in \{1,2,\ldots,m\}$.

It can be observed that if we let $V = \frac{1}{c}$ and $\frac{Q_k(0)}{V} = \lambda_k(0), \forall k \in \{1,2,\ldots,m\}$, then Algorithm 1 and Algorithm 2 yield the same sequence of $\mathbf{x}(t)$ and $\frac{Q_k(t)}{V} = \lambda_k(t), \forall k \in \{1,2,\ldots,m\}, \forall t \geq 0$. Thus, the convergence time results of the drift-plus-penalty algorithm can be carried over to the dual subgradient algorithm and vice versa.

D. Related Works

Problem (1)-(3) in general can be solved via interior point methods with linear convergence time. To achieve the linear convergence time, however, the barrier parameters must be scaled carefully and the complexity associated with

each iteration can be high. In contrast, the drift-plus-penalty algorithm is a first-order method and often yields distributed implementations when the objective and constraint functions are separable. A lot of literature focuses on the convergence time of first-order methods. Work [5] considers the problem of minimizing non-smooth and strongly convex functions over convex sets based on noisy observations of gradients and shows that the stochastic gradient algorithm with the averaged primal sequence guarantees that the suboptimality decays like $O(\log(t)/t)$. For general convex programs in the form of (1)-(3), where the objective function $f(\mathbf{x})$ is convex but not necessarily strongly convex, the convergence time of the drift-plus-penalty algorithm is shown to be $O(\frac{1}{\epsilon^2})$ in [2], [6]. A similar $O(\frac{1}{\epsilon^2})$ convergence time of the dual subgradient algorithm with the averaged primal sequence is shown in [7]. A recent work [4] shows that the convergence time of the drift-plus-penalty algorithm is $O(\frac{1}{\epsilon})$ if the dual function is locally polyhedral and the convergence time is $O(\frac{1}{\epsilon^{1.5}})$ if the dual function is locally quadratic. For a special class of strongly convex programs in the form of (1)-(3), where $f(\mathbf{x})$ is second-order differentiable and strongly convex and $g_k(\mathbf{x}), \forall k \in \{1,2,\ldots,m\}$ are second-order differentiable and have bounded Jacobians[1], the convergence time of the dual subgradient algorithm is shown to be $O(\frac{1}{\epsilon})$ in [8].

This paper considers a class of strongly convex programs that is more general than those treated in [8]. Besides the strong convexity of $f(\mathbf{x})$, we only require that $g_k(\mathbf{x}), k \in \{1,2,\ldots,m\}$ are Lipschitz continuous. The function $f(\mathbf{x})$ and $g_k(\mathbf{x}), k \in \{1,2,\ldots,m\}$ can even be non-differentiable (and hence non-smooth). This paper shows that the convergence time of the drift-plus-penalty algorithm for general strongly convex programs is $O(\frac{1}{\epsilon})$. If the dual function of the strongly convex program is smooth and locally quadratic, this paper shows that the convergence time of the drift-plus-penalty algorithm can be improved to $O(\frac{1}{\epsilon^{2/3}})$ by using shifted running averages. Further improvement to $O(\log(\frac{1}{\epsilon}))$ convergence time is shown under more restrictive assumptions.

II. Preliminaries and Basic Analysis

A. Prelimaries

Definition 1 (Lipschitz Continuity): Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a convex set. Function f is said to be Lipschitz continuous on $\mathcal{X}$ with modulus L if there exists $L > 0$ such that $|f(\mathbf{y}) - f(\mathbf{x})| \leq L\|\mathbf{y} - \mathbf{x}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.

Lemma 1: Let $g_k(\mathbf{x}), k \in \{1,2,\ldots,m\}$ be Lipschitz continuous on $\mathcal{X}$ with modulus L. Then $\|\mathbf{g}(\mathbf{y}) - \mathbf{g}(\mathbf{x})\| \leq L\sqrt{m}\|\mathbf{y} - \mathbf{x}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.

Definition 2 (Strongly Convex Functions): Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a convex set. Function f is said to be strongly convex on $\mathcal{X}$ with modulus α if there exists a constant $\alpha > 0$ such that $f(\mathbf{x}) - \frac{1}{2}\alpha\|\mathbf{x}\|^2$ is convex on $\mathcal{X}$.

Define $\partial f(\mathbf{x})$ as the set of all subgradients of function f at a point $\mathbf{x}$ in $\mathcal{X}$.

[1]Note that bounded Jacobians imply Lipschitz continuity.

Lemma 2 (Theorem 6.1.2 in [9]): If $f(\mathbf{x})$ is strongly convex on convex set $\mathcal{X}$ with modulus α, then $f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{d}^T(\mathbf{y}-\mathbf{x}) + \frac{\alpha}{2}\|\mathbf{y}-\mathbf{x}\|^2$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ and all $\mathbf{d} \in \partial f(\mathbf{x})$.

Lemma 3 (Proposition B.24 (f) in [10]): If $f(\mathbf{x})$ is convex on convex set $\mathcal{X}$ and $\mathbf{x}^{opt}$ is a global minimum, then there exists $\mathbf{d} \in \partial f(\mathbf{x}^{opt})$ such that $\mathbf{d}^T(\mathbf{y} - \mathbf{x}^{opt}) \geq 0$ for all $\mathbf{y} \in \mathcal{X}$.

Combining Lemma 2 and Lemma 3 gives the following:

Corollary 1: Let $f(\mathbf{x})$ be strongly convex on convex set $\mathcal{X}$ with modulus α. If $\mathbf{x}^{opt}$ is a global minimum, then $f(\mathbf{x}^{opt}) \leq f(\mathbf{y}) - \frac{\alpha}{2}\|\mathbf{y} - \mathbf{x}^{opt}\|^2, \forall \mathbf{y} \in \mathcal{X}$.

B. Properties of the Drift

Let $\mathbf{Q}(t) = \big[Q_1(t), \ldots, Q_m(t)\big]^T$ be the vector of virtual queue backlogs. Define a Lyapunov function as $L(t) = \frac{1}{2}\|\mathbf{Q}(t)\|^2$ and the Lyapunov drift as $\Delta(t) = L(t+1) - L(t)$.

Lemma 4: At each iteration t in Algorithm 1, the Lyapunov drift is given by

$$\Delta(t) = \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}(t)) - \frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 \quad (4)$$

Proof: The virtual queue update equations $Q_k(t+1) = \max\{Q_k(t) + g_k(\mathbf{x}(t)), 0\}, \forall k \in \{1,2,\ldots,m\}$ can be rewritten as

$$Q_k(t+1) = Q_k(t) + \tilde{g}_k(\mathbf{x}(t)), \forall k \in \{1,2,\ldots,m\}, \quad (5)$$

where $\tilde{g}_k(\mathbf{x}(t)) = \begin{cases} g_k(\mathbf{x}(t)), & \text{if } Q_k(t) + g_k(\mathbf{x}(t)) \geq 0 \\ -Q_k(t), & \text{else} \end{cases}$, $\forall k \in \{1,2,\ldots,m\}$.

Fix $k \in \{1,2,\ldots,m\}$. Squaring both sides of (5) and dividing by factor 2 yields:

$$\begin{aligned}
&\frac{1}{2}(Q_k(t+1))^2 \\
=&\frac{1}{2}(Q_k(t))^2 + \frac{1}{2}\big(\tilde{g}_k(\mathbf{x}(t))\big)^2 + Q_k(t)\tilde{g}_k(\mathbf{x}(t)) \\
=&\frac{1}{2}(Q_k(t))^2 + \frac{1}{2}\big(\tilde{g}_k(\mathbf{x}(t))\big)^2 + Q_k(t)g_k(\mathbf{x}(t)) \\
&+ Q_k(t)\big(\tilde{g}_k(\mathbf{x}(t)) - g_k(\mathbf{x}(t))\big) \\
\overset{(a)}{=}&\frac{1}{2}(Q_k(t))^2 + \frac{1}{2}\big(\tilde{g}_k(\mathbf{x}(t))\big)^2 + Q_k(t)g_k(\mathbf{x}(t)) \\
&- \tilde{g}_k(\mathbf{x}(t))\big(\tilde{g}_k(\mathbf{x}(t)) - g_k(\mathbf{x}(t))\big) \\
=&\frac{1}{2}(Q_k(t))^2 - \frac{1}{2}\big(\tilde{g}_k(\mathbf{x}(t))\big)^2 + \big(Q_k(t) + \tilde{g}_k(\mathbf{x}(t))\big)g_k(\mathbf{x}(t)) \\
\overset{(b)}{=}&\frac{1}{2}(Q_k(t))^2 - \frac{1}{2}\big(Q_k(t+1) - Q_k(t)\big)^2 + Q_k(t+1)g_k(\mathbf{x}(t))
\end{aligned}$$

where (a) follows from the fact that $Q_k(t)\big(\tilde{g}_k(\mathbf{x}(t)) - g_k(\mathbf{x}(t))\big) = -\tilde{g}_k(\mathbf{x}(t))\big(\tilde{g}_k(\mathbf{x}(t)) - g_k(\mathbf{x}(t))\big)$, which can be shown by considering $\tilde{g}_k(\mathbf{x}(t)) = g_k(\mathbf{x}(t))$ and $\tilde{g}_k(\mathbf{x}(t)) \neq g_k(\mathbf{x}(t))$; and (b) follows from the fact that $Q_k(t+1) = Q_k(t) + \tilde{g}_k(\mathbf{x}(t))$.

Summing over $k \in \{1,2,\ldots,m\}$ yields $\frac{1}{2}\|\mathbf{Q}(t+1)\|^2 = \frac{1}{2}\|\mathbf{Q}(t)\|^2 - \frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 + \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}(t))$. Rearranging the terms yields the desired result. ■

C. The Drift-Plus-Penalty Algorithm

Let $V > 0$ be a constant parameter. By Lemma 4, the *drift-plus-penalty* is defined as $\Delta(t) + Vf(\mathbf{x}(t)) = Vf(\mathbf{x}(t)) + \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}(t)) - \frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2$. To solve problem (1)-(3), the idea is to minimize this expression at each iteration. Since $\mathbf{Q}(t+1)$ is unavailable at iteration t, we use $Vf(\mathbf{x}(t)) + \mathbf{Q}^T(t)\mathbf{g}(\mathbf{x}(t))$ as a "reasonable" approximation of the actual drift-plus-penalty. The corresponding algorithm is described in Algorithm 1.

For general convex programs in the form of (1)-(3), where the objective function$f(\mathbf{x})$ is convex but not necessarily strongly convex, it is shown in [2], [6] that the sequence $\overline{\mathbf{x}}(t)$ yielded by Algorithm 1 with $V = 1/\epsilon$ satisfies $f(\overline{\mathbf{x}}(t)) \leq f(\mathbf{x}^*) + O(\epsilon)$ and $g_k(\overline{\mathbf{x}}(t)) \leq O(\epsilon), \forall k \in \{1, 2, \ldots, m\}$ for all $t \geq \frac{1}{\epsilon^2}$. That is, the convergence time for general convex programs is $O(\frac{1}{\epsilon^2})$.

III. Convergence Time Analysis

This section analyzes the convergence time of the drift-plus-penalty algorithm for strongly convex program (1)-(3).

A. Problem Assumptions

Throughout this paper, we require the following assumptions on problem (1)-(3):

Assumption 1: Objective function $f(\mathbf{x})$ is strongly convex on $\mathcal{X}$ with modulus α. Each constraint function $g_k(\mathbf{x}), k \in \{1, 2, \ldots, m\}$ is Lipschitz continuous on $\mathcal{X}$ with modulus β.

Assumption 2 (Strong Duality): The strong duality holds for problem (1)-(3). That is, there exists a Lagrange multiplier vector $\boldsymbol{\lambda}^* = [\lambda_1^*, \lambda_2^*, \ldots, \lambda_m^*] \geq \mathbf{0}$ such that $q(\boldsymbol{\lambda}^*) = \min_{\mathbf{x}\in\mathcal{X}} \Big\{ f(\mathbf{x}) : g_k(\mathbf{x}) \leq 0, \forall k \in \{1, 2, \ldots, m\} \Big\}$,where $q(\boldsymbol{\lambda}) = \min_{\mathbf{x}\in\mathcal{X}} \{f(\mathbf{x}) + \sum_{k=1}^m \lambda_k g_k(\mathbf{x})\}$ is the *Lagrangian dual function* of problem (1)-(3).

If the strong duality holds for a convex program in the form of (1)-(3), then $\lambda_k^* g_k(\mathbf{x}^*) = 0, \forall k \in \{1, 2, \ldots, m\}$ and $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} \Big[f(\mathbf{x}) + \sum_{k=1}^m \lambda_k^* g_k(\mathbf{x}) \Big]$, where $\boldsymbol{\lambda}^*$ is defined in Assumption 2 and $\mathbf{x}^*$ is the optimal solution to problem (1)-(3). (See also Theorem 6.2.5 in [11].)

For convex programs, strong duality is implied by Slater's condition [11]. However, there are convex programs where strong duality holds but Slater's condition does not hold. Thus, the strong duality assumption is a mild assumption.

B. Objective Value Violations

Lemma 5: Let $\mathbf{x}^* \in \mathcal{X}$ be the optimal solution to problem (1)-(3). At each iteration t in Algorithm 1, the drift-plus-penalty satisfies $\Delta(t) + Vf(\mathbf{x}(t)) \leq Vf(\mathbf{x}^*) + B(t), \forall t \geq 0$, where $B(t) = -\frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2 + \mathbf{Q}^T(t)\big[\mathbf{g}(\mathbf{x}^*) - \mathbf{g}(\mathbf{x}(t))\big] + \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}(t))$.

Proof: Fix $t \geq 0$. Since $f(\mathbf{x})$ is strongly convex with modulus α; $g_k(\mathbf{x}), \forall k \in \{1, 2, \ldots, m\}$ are convex; and $Q_k(t), \forall k \in \{1, 2, \ldots, m\}$ are non-negative at each iteration t, the function $Vf(\mathbf{x}) + \sum_{k=1}^m Q_k(t) g_k(\mathbf{x})$ is also strongly convex with modulus $V\alpha$ at each iteration t. Note that $\mathbf{x}(t) = \operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} \big[Vf(\mathbf{x}) + \sum_{k=1}^m Q_k(t) g_k(\mathbf{x}) \big]$. By Corollary 1 with $\mathbf{x}^{opt} = \mathbf{x}(t)$ and $\mathbf{y} = \mathbf{x}^*$, we have $\big[Vf(\mathbf{x}(t)) + \sum_{k=1}^m Q_k(t) g_k(\mathbf{x}(t))\big] \leq \big[Vf(\mathbf{x}^*) + \sum_{k=1}^m Q_k(t) g_k(\mathbf{x}^*)\big] - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2$.

Hence, $Vf(\mathbf{x}(t)) \leq Vf(\mathbf{x}^*) + \mathbf{Q}^T(t)\big[\mathbf{g}(\mathbf{x}^*) - \mathbf{g}(\mathbf{x}(t))\big] - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2$.

Adding the above inequality to equation (4) and using the definition of $B(t)$ yields the result. ∎

Lemma 6: If $V \geq \frac{m\beta^2}{\alpha}$ in Algorithm 1, then $B(t) \leq 0, \forall t \geq 0$.

Proof: Since $\mathbf{x}^*$ is the optimal solution to problem (1)-(3), we have $g_k(\mathbf{x}^*) \leq 0, \forall k \in \{1, 2, \ldots, m\}$. Note that $\mathbf{Q}_k(t+1) \geq 0, \forall k \in \{1, 2, \ldots, m\}, \forall t \geq 0$. Thus,

$$\mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}^*) \leq 0, \quad \forall t \geq 0 \tag{6}$$

Now we have,

$$\begin{aligned}
B(t) =& -\frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2 \\
&+ \mathbf{Q}^T(t)\big[\mathbf{g}(\mathbf{x}^*) - \mathbf{g}(\mathbf{x}(t))\big] + \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}(t)) \\
\overset{(a)}{\leq}& -\frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2 \\
&+ \mathbf{Q}^T(t)\big[\mathbf{g}(\mathbf{x}^*) - \mathbf{g}(\mathbf{x}(t))\big] + \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}(t)) \\
&- \mathbf{Q}^T(t+1)\mathbf{g}(\mathbf{x}^*) \\
=& -\frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2 \\
&+ \big[\mathbf{Q}^T(t) - \mathbf{Q}^T(t+1)\big]\big[\mathbf{g}(\mathbf{x}^*) - \mathbf{g}(\mathbf{x}(t))\big] \\
\overset{(b)}{\leq}& -\frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2 \\
&+ \|\mathbf{Q}(t) - \mathbf{Q}(t+1)\|\|\mathbf{g}(\mathbf{x}(t)) - \mathbf{g}(\mathbf{x}^*)\| \\
\overset{(c)}{\leq}& -\frac{1}{2}\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\|^2 - \frac{V\alpha}{2}\|\mathbf{x}(t) - \mathbf{x}^*\|^2 \\
&+ \sqrt{m}\beta\|\mathbf{Q}(t) - \mathbf{Q}(t+1)\|\|\mathbf{x}(t) - \mathbf{x}^*\| \\
=& -\frac{1}{2}\Big(\|\mathbf{Q}(t+1) - \mathbf{Q}(t)\| - \sqrt{m}\beta\|\mathbf{x}(t) - \mathbf{x}^*\|\Big)^2 \\
&- \frac{1}{2}(V\alpha - m\beta^2)\|\mathbf{x}(t) - \mathbf{x}^*\|^2 \\
\overset{(d)}{\leq}& 0
\end{aligned}$$

where (a) follows from (6); (b) follows from the Cauchy-Schwartz inequality; (c) follows from Lemma 1; and (d) follows from $V \geq \frac{m\beta^2}{\alpha}$. ∎

Lemmas 5 and 6 immediately imply the following results.

Corollary 2: If $V \geq \frac{m\beta^2}{\alpha}$ in Algorithm 1, then the drift-plus-penalty satisfies

$$\Delta(t) + Vf(\mathbf{x}(t)) \leq Vf(\mathbf{x}^*), \forall t \geq 0. \tag{7}$$

Theorem 1 (Objective Value Violations): Let $\mathbf{x}^* \in \mathcal{X}$ be the optimal solution to problem (1)-(3). If $V \geq \frac{m\beta^2}{\alpha}$ in Algorithm 1, then $f(\overline{\mathbf{x}}(t)) \leq f(\mathbf{x}^*) + \frac{\|\mathbf{Q}(0)\|^2}{2Vt}, \forall t \geq 1$.

Proof: By Corollary 2, we have $\Delta(\tau) + Vf(\mathbf{x}(\tau)) \leq Vf(\mathbf{x}^*)$ for all $\tau \in \{0, 1, \ldots, t-1\}$. Summing over $\tau \in$

$\{0,1,\ldots,t-1\}$, we have

$$\sum_{\tau=0}^{t-1}\Delta(\tau)+V\sum_{\tau=0}^{t-1}f(\mathbf{x}(\tau))\leq Vtf(\mathbf{x}^*)$$

$$\Rightarrow\quad L(t)-L(0)+V\sum_{\tau=0}^{t-1}f(\mathbf{x}(\tau))\leq Vtf(\mathbf{x}^*)$$

$$\Rightarrow\quad \frac{1}{t}\sum_{\tau=0}^{t-1}f(\mathbf{x}(\tau))\leq f(\mathbf{x}^*)+\frac{L(0)-L(t)}{Vt}\leq f(\mathbf{x}^*)+\frac{L(0)}{Vt}$$

Finally, Jensen's inequality implies that $f(\overline{\mathbf{x}}(t))\leq \frac{1}{t}\sum_{\tau=0}^{t-1}f(\mathbf{x}(\tau))\leq f(\mathbf{x}^*)+\frac{L(0)}{Vt}=f(\mathbf{x}^*)+\frac{\|\mathbf{Q}(0)\|^2}{2Vt}$ ■

Corollary 3: If $V\geq\frac{m\beta^2}{\alpha}$ and $Q_k(0)=0,\forall k\in\{1,2,\ldots,m\}$ in Algorithm 1, then $f(\overline{\mathbf{x}}(t))\leq f(\mathbf{x}^*),\forall t\geq 1$.

C. Constraint Violations

The analysis of constraint violations is similar to that in [6] for general convex programs. However, using the improved upper bound of the drift-plus-penalty expression in Corollary 2, the convergence time of constraint violations in strongly convex programs is order-wise better than that in [6].

Lemma 7: For any $t_2>t_1\geq 0$, $Q_k(t_2)\geq Q_k(t_1)+\sum_{\tau=t_1}^{t_2-1}g_k(\mathbf{x}(\tau)),\forall k\in\{1,2,\ldots,m\}$. In particular, for any $t>0$, $Q_k(t)\geq Q_k(0)+\sum_{\tau=0}^{t-1}g_k(\mathbf{x}(\tau)),\forall k\in\{1,2,\ldots,m\}$

Proof: Fix $k\in\{1,2,\ldots,m\}$. Note that $Q_k(t_1+1)=\max\{Q_k(t_1)+g_k(\mathbf{x}(t_1)),0\}\geq Q_k(t_1)+g_k(\mathbf{x}(t_1))$. By induction, this lemma follows. ■

Lemma 8: Let $\boldsymbol{\lambda}^*\geq\mathbf{0}$ be given in Assumption 2. If $V\geq\frac{m\beta^2}{\alpha}$ in Algorithm 1, then the virtual queue vector satisfies

$$\|\mathbf{Q}(t)\|\leq\sqrt{\|\mathbf{Q}(0)\|^2+V^2\|\boldsymbol{\lambda}^*\|^2}+V\|\boldsymbol{\lambda}^*\|,\forall t\geq 1. \quad (8)$$

Proof: Let $\mathbf{x}^*$ be the optimal solution to problem (1)-(3). The strong duality assumption implies that $f(\mathbf{x}^*)=f(\mathbf{x}^*)+\sum_{k=1}^m\lambda_k^*g_k(\mathbf{x}^*)$ and $f(\mathbf{x}^*)+\sum_{k=1}^m\lambda_k^*g_k(\mathbf{x}^*)\leq f(\mathbf{x}(\tau))+\sum_{k=1}^m\lambda_k^*g_k(\mathbf{x}(\tau)),\forall\tau\in\{0,1,\ldots\}$. Thus, for all $\tau\geq 0$, we have $f(\mathbf{x}^*)-f(\mathbf{x}(\tau))\leq\sum_{k=1}^m\lambda_k^*g_k(\mathbf{x}(\tau))$. Summing over $\tau\in\{0,1,\ldots,t-1\}$ yields

$$tf(\mathbf{x}^*)-\sum_{\tau=0}^{t-1}f(\mathbf{x}(\tau))\leq\sum_{\tau=0}^{t-1}\sum_{k=1}^m\lambda_k^*g_k(\mathbf{x}(\tau))$$

$$=\sum_{k=1}^m\lambda_k^*\Big[\sum_{\tau=0}^{t-1}g_k(\mathbf{x}(\tau))\Big]\overset{(a)}{\leq}\sum_{k=1}^m\lambda_k^*[Q_k(t)-Q_k(0)]$$

$$\leq\sum_{k=1}^m\lambda_k^*Q_k(t)\overset{(b)}{\leq}\|\boldsymbol{\lambda}^*\|\|\mathbf{Q}(t)\|,$$

where (a) follows from Lemma 7 and (b) follows from Cauchy-Schwartz inequality.

On the other hand, summing (7) in Corollary 2 over $\tau\in\{0,1,\ldots,t-1\}$ and dividing by V yield $tf(\mathbf{x}^*)-\sum_{\tau=0}^{t-1}f(\mathbf{x}(\tau))\geq\frac{L(t)-L(0)}{V}=\frac{\|\mathbf{Q}(t)\|^2-\|\mathbf{Q}(0)\|^2}{2V}$.

Combining the last two inequalities yields $\frac{\|\mathbf{Q}(t)\|^2-\|\mathbf{Q}(0)\|^2}{2V}\leq\|\boldsymbol{\lambda}^*\|\|\mathbf{Q}(t)\|\Rightarrow\big(\|\mathbf{Q}(t)\|-V\|\boldsymbol{\lambda}^*\|\big)^2\leq\|\mathbf{Q}(0)\|^2+V^2\|\boldsymbol{\lambda}^*\|^2\Rightarrow\|\mathbf{Q}(t)\|\leq\sqrt{\|\mathbf{Q}(0)\|^2+V^2\|\boldsymbol{\lambda}^*\|^2}+V\|\boldsymbol{\lambda}^*\|$. ■

Theorem 2 (Constraint Violations): Let $\boldsymbol{\lambda}^*\geq\mathbf{0}$ be defined in Assumption 2. If $V\geq\frac{m\beta^2}{\alpha}$ in Algorithm 1, then the constraint functions satisfy $g_k(\overline{\mathbf{x}}(t))\leq\frac{\sqrt{\|\mathbf{Q}(0)\|^2+V^2\|\boldsymbol{\lambda}^*\|^2}+V\|\boldsymbol{\lambda}^*\|}{t},\forall k\in\{1,2,\ldots,m\},\forall t\geq 1$.

Proof: Fix $t\geq 1$ and $k\in\{1,2,\ldots,m\}$. Recall that $\overline{\mathbf{x}}(t)=\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbf{x}(\tau)$. Thus,

$$g_k(\overline{\mathbf{x}}(t))\overset{(a)}{\leq}\frac{1}{t}\sum_{\tau=0}^{t-1}g_k(\mathbf{x}(\tau))\overset{(b)}{\leq}\frac{Q_k(t)-Q_k(0)}{t}$$

$$\leq\frac{Q_k(t)}{t}\leq\frac{\|\mathbf{Q}(t)\|}{t}\overset{(c)}{\leq}\frac{\sqrt{\|\mathbf{Q}(0)\|^2+V^2\|\boldsymbol{\lambda}^*\|^2}+V\|\boldsymbol{\lambda}^*\|}{t}$$

where (a) follows from the convexity of $g_k(\mathbf{x}),k\in\{1,2,\ldots,m\}$; (b) follows from Lemma 7; and (c) follows from Lemma 8. ■

Theorems 1 and 2 show that the drift-plus-penalty algorithm provides an ϵ-optimal solution to a general strongly convex program with convergence time $O(1/\epsilon)$.

IV. Extensions

This section proposes a new variation of the drift-plus-penalty algorithm and shows that the convergence time is improved to $O(\frac{1}{\epsilon^{2/3}})$ when the dual function of problem (1)-(3) satisfies additional assumptions.

A. Smoothness and Locally Quadratic

Definition 3 (Smooth Functions): Let $\mathcal{X}\subseteq\mathbb{R}^n$ and function $f(\mathbf{x})$ be continuously differentiable on $\mathcal{X}$. Function $f(\mathbf{x})$ is said to be smooth on $\mathcal{X}$ with modulus L if $\nabla_{\mathbf{x}}f(\mathbf{x})$ is Lipschitz continuous on $\mathcal{X}$ with modulus L.

Assumption 3 (Smooth Dual Functions): The dual function of problem (1)-(3) is smooth on $\mathbb{R}_+^m$ with modulus γ.

The next lemma from [8] provides a sufficient condition of Assumption 3.

Lemma 9 (Theorem 2.1 in [8]): In problem (1)-(3), if the objective function $f(\mathbf{x})$ is second-order differentiable and is strongly convex with modulus σ_F on $\mathcal{X}$ and constraints $\mathbf{g}(\mathbf{x})$ are second-order differentiable and have a bounded Jacobian, i.e., $\|\nabla\mathbf{g}(\mathbf{x})\|_F\leq c_h$ for all $\mathbf{x}\in\mathcal{X}$, then the dual function q is smooth on $\mathbb{R}_+^m$ with modulus $\frac{c_h^2}{\sigma_F}$.

Assumption 4 (Locally Quadratic Dual Functions): Let $\boldsymbol{\lambda}^*$ be a Lagrange multiplier of problem (1)-(3) defined in Assumption 2. There exists $D_q>0$ and $L_q>0$ such that for any $\boldsymbol{\lambda}\in\{\boldsymbol{\lambda}\in\mathbb{R}_+^m:\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^*\|\leq D_q\}$, the dual function $q(\boldsymbol{\lambda})=\min_{\mathbf{x}\in\mathcal{X}}\Big\{f(\mathbf{x})+\sum_{k=1}^m\lambda_kg_k(\mathbf{x})\Big\}$ satisfies $q(\boldsymbol{\lambda}^*)\geq q(\boldsymbol{\lambda})+L_q\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^*\|^2$.

Assumption 4 is introduced in [3] and further studied in [4]. It has been shown in [4] that if a convex program satisfies this assumption, then the drift-plus-penalty algorithm with restarted running averages can have better convergence time compared with the standard drift-plus-penalty algorithm. Inspired by this result, the next subsection shows that if problem (1)-(3) satisfies Assumptions 1-4, then the convergence time of another variation of the drift-plus-penalty algorithm, called the drift-plus-penalty algorithm with shifted running average, is $O(\frac{1}{\epsilon^{2/3}})$.

B. Convergence Time Analysis

Lemma 10: Suppose problem (1)-(3) satisfies Assumption 4. Let $q(\boldsymbol{\lambda}), \boldsymbol{\lambda}^*, D_q$ and L_q be defined in Assumption 4. We have the following properties:

1) If $\boldsymbol{\lambda} \in \mathbb{R}^m_+$ and $q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}) \leq L_q D_q^2$, then $\|\boldsymbol{\lambda} - \boldsymbol{\lambda}^*\| \leq D_q$.
2) The Lagrange multiplier defined in Assumption 2 is unique.

Proof: See [12] for the details. ∎

Let $\boldsymbol{\lambda}(t) = \frac{\mathbf{Q}(t)}{V}$. By the strong convexity of $f(\mathbf{x})$ and Proposition B.25 in [10], the dual function $q(\boldsymbol{\lambda})$ is differentiable and has gradient $\nabla_{\boldsymbol{\lambda}} q(\boldsymbol{\lambda}(t)) = \mathbf{g}(\mathbf{x}(t))$. It can be observed that the dynamic of $\boldsymbol{\lambda}(t)$ is the same as that in the projected gradient method with constant step size $\frac{1}{V}$ to solve $\max_{\boldsymbol{\lambda} \in \mathbb{R}^m_+} \big\{q(\boldsymbol{\lambda})\big\}$. Thus, we have the next theorem.

Theorem 3: Assume problem (1)-(3) satisfies Assumptions 1-3. Let $\{\boldsymbol{\lambda}(t) = \frac{\mathbf{Q}(t)}{V}, t \geq 0\}$ be the sequence yielded by Algorithm 1 with fixed $\mathbf{Q}(0) \geq \mathbf{0}$ and $V > 0$. Let $\theta = \max\Big\{\frac{4V^2\|\boldsymbol{\lambda}(0) - \boldsymbol{\lambda}^*\|^2}{(2V-\gamma)}, q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(0))\Big\}$. If $V \geq \gamma$, then $q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(t)) \leq \frac{\theta}{t}, \forall t \geq 1$.

Proof: This is essentially the same as the convergence time proof of projected gradient methods for unconstrained smooth optimization in [13]. See [12] for details. ∎

Lemma 11: Assume problem (1)-(3) satisfies Assumptions 1-4. Let $\{\boldsymbol{\lambda}(t) = \frac{\mathbf{Q}(t)}{V}, t \geq 0\}$ be the sequence yielded by Algorithm 1 with fixed $\mathbf{Q}(0) \geq \mathbf{0}$ and $V > 0$. If $V \geq \gamma$, then $\|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\| \leq D_q$ for all $t > \max\Big\{\frac{4V^2\|\boldsymbol{\lambda}(0) - \boldsymbol{\lambda}^*\|}{(2V-\gamma)L_q D_q^2}, \frac{q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(0))}{L_q D_q^2}\Big\}$. That is, there exists a constant $T_q = \Big\lceil \max\Big\{\frac{4V^2\|\boldsymbol{\lambda}(0) - \boldsymbol{\lambda}^*\|}{(2V-\gamma)L_q D_q^2}, \frac{q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(0))}{L_q D_q^2}\Big\}\Big\rceil$ such that $\boldsymbol{\lambda}(T_q)$ arrives where Assumption 4 holds and Assumption 4 holds for all $\boldsymbol{\lambda}(t), t \geq T_q$.

Proof: By Theorem 3 and Lemma 10, if $\frac{\theta}{t} < L_q D_q^2$, then $\|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\| \leq D_q$. It can be checked that if $t > \max\Big\{\frac{4V^2\|\boldsymbol{\lambda}(0) - \boldsymbol{\lambda}^*\|}{(2V-\gamma)L_q D_q^2}, \frac{q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(0))}{L_q D_q^2}\Big\}$, then $\frac{\theta}{t} < L_q D_q^2$. ∎

In the remaining part of this subsection, let θ be defined in Theorem 3 and T_q be defined in Lemma 11.

Lemma 12: Assume problem (1)-(3) satisfies Assumptions 1-4. Let $\mathbf{Q}(t), t \in \{1, 2, \dots\}$ be the sequence yielded by Algorithm 1 with fixed $\mathbf{Q}(0) \geq \mathbf{0}$ and $V > 0$. Let $\boldsymbol{\lambda}(t) = \frac{\mathbf{Q}(t)}{V}, \forall t \geq 0$. If $V \geq \gamma$, $\|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\| \leq \frac{1}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}, \forall t \geq T_q$.

Proof: By Theorem 3, we have $q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(t)) \leq \frac{\theta}{t}, \forall t \in \{1, 2, \dots\}$. By Lemma 11, we have $q(\boldsymbol{\lambda}^*) - q(\boldsymbol{\lambda}(t)) \geq L_q\|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\|^2, \forall t \geq T_q$. Thus, for all $t \geq T_q$, we have $\|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\| \leq \frac{1}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}$. ∎

Corollary 4: Under the assumption of Lemma 12, $\|\boldsymbol{\lambda}(2t) - \boldsymbol{\lambda}(t)\| \leq \frac{2}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}, \forall t \geq T_q$; or equivalently, $\|\mathbf{Q}(2t) - \mathbf{Q}(t)\| \leq \frac{2V}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}, \forall t \geq T_q$.

Proof: By Lemma 12, we have $\|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\| \leq \frac{1}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}$ and $\|\boldsymbol{\lambda}(2t) - \boldsymbol{\lambda}^*\| \leq \frac{1}{\sqrt{2t}}\sqrt{\frac{\theta}{L_q}}$. Thus, $\|\boldsymbol{\lambda}(2t) - \boldsymbol{\lambda}(t)\| \leq \|\boldsymbol{\lambda}(2t) - \boldsymbol{\lambda}^*\| + \|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\| \leq \frac{1}{\sqrt{2t}}\sqrt{\frac{\theta}{L_q}} + \frac{1}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}} \leq \frac{2}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}$. ∎

The drift-plus-penalty algorithm with shifted running averages is summarized as follows:

Algorithm 3: [The Drift-Plus-Penalty Algorithm with Shifted Running Averages] Let $V > 0$ and $Q_k(0) \geq 0, \forall k \in \{1, 2, \dots, m\}$ be given constant parameters. At each iteration t, update $\mathbf{x}(t)$, $Q_k(t+1)$ and $\overline{\mathbf{x}}(t+1)$ as follows:

- $\mathbf{x}(t) = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} \Big[Vf(\mathbf{x}) + \sum_{k=1}^{m} Q_k(t) g_k(\mathbf{x})\Big]$.
- $Q_k(t+1) = \max\big\{Q_k(t) + g_k(\mathbf{x}(t)), 0\big\}, \forall k \in \{1, 2, \dots, m\}$.
- $$\overline{\mathbf{x}}(t+1) = \begin{cases} \frac{1}{\frac{t+1}{2}} \sum_{\tau = \frac{t+1}{2}}^{t} \mathbf{x}(\tau) & \text{if } t+1 \text{ is even} \\ \overline{\mathbf{x}}(t) & \text{if } t+1 \text{ is odd} \end{cases}$$

The only difference between Algorithm 3 and Algorithm 1 is the step of updating running averages. At each iteration t, the running averages in Algorithm 1 are started from iteration 0 while the running averages in Algorithm 3 are started from iteration $\lfloor \frac{t+1}{2} \rfloor$ and are updated only for even iterations.

The next two theorems show that if problem (1)-(3) satisfies Assumptions 1-4, then Algorithm 3 provides an ϵ-optimal solution with convergence time $O(\frac{1}{\epsilon^{2/3}})$.

Theorem 4: Assume problem (1)-(3) satisfies Assumptions 1-4. Let $\mathbf{x}^*$ be the optimal solution and $\boldsymbol{\lambda}^*$ be defined in Assumption 4. Let $\overline{\mathbf{x}}(t), \mathbf{Q}(t)$ be sequences yielded by Algorithm 3. If $V \geq \max\{\frac{m\beta^2}{\alpha}, \gamma\}$, then $f(\overline{\mathbf{x}}(2t)) \leq f(\mathbf{x}^*) + \frac{1}{t^{3/2}}\Big(\frac{2V\theta}{t^{1/2}L_q} + \frac{2\sqrt{\theta}\eta}{\sqrt{L_q}}\Big), \forall t \geq T_q$, where $\eta = \sqrt{\|\mathbf{Q}(0)\|^2 + V^2\|\boldsymbol{\lambda}^*\|^2} + V\|\boldsymbol{\lambda}^*\|$.

Proof: Fix $t > T_q$. By Corollary 2, we have $\Delta(\tau) + Vf(\mathbf{x}(\tau)) \leq Vf(\mathbf{x}^*)$ for all $\tau \in \{0, 1, \dots\}$. Summing over $\tau \in \{t, t+1, \dots, 2t-1\}$ yields $\sum_{\tau=t}^{2t-1}\Delta(\tau) + V\sum_{\tau=t}^{2t-1} f(\mathbf{x}(\tau)) \leq Vtf(\mathbf{x}^*)$. Dividing by factor Vt yields

$$\frac{1}{t}\sum_{\tau=t}^{2t-1} f(\mathbf{x}(\tau)) \leq f(\mathbf{x}^*) + \frac{L(t) - L(2t)}{Vt} \tag{9}$$

Thus, we have

$$\begin{aligned} f(\overline{\mathbf{x}}(2t)) &\overset{(a)}{\leq} \frac{1}{t}\sum_{\tau=t}^{2t-1} f(\mathbf{x}(\tau)) \overset{(b)}{\leq} f(\mathbf{x}^*) + \frac{L(t) - L(2t)}{Vt} \\ &= f(\mathbf{x}^*) + \frac{\|\mathbf{Q}(t)\|^2 - \|\mathbf{Q}(2t)\|^2}{2Vt} \\ &= f(\mathbf{x}^*) + \frac{\|\mathbf{Q}(t) - \mathbf{Q}(2t) + \mathbf{Q}(2t)\|^2 - \|\mathbf{Q}(2t)\|^2}{2Vt} \\ &\overset{(c)}{\leq} f(\mathbf{x}^*) + \frac{\|\mathbf{Q}(t) - \mathbf{Q}(2t)\|^2 + 2\|\mathbf{Q}(2t)\|\|\mathbf{Q}(t) - \mathbf{Q}(2t)\|}{2Vt} \\ &\overset{(d)}{\leq} f(\mathbf{x}^*) + \frac{\frac{4V^2\theta}{tL_q} + 2\eta\frac{2V}{\sqrt{t}}\frac{\sqrt{\theta}}{\sqrt{L_q}}}{2Vt} \\ &= f(\mathbf{x}^*) + \frac{1}{t^{3/2}}\Big(\frac{2V\theta}{t^{1/2}L_q} + \frac{2\sqrt{\theta}\eta}{\sqrt{L_q}}\Big) \end{aligned}$$

where (a) follows from the fact that $\overline{\mathbf{x}}(2t) = \frac{1}{t}\sum_{\tau=t}^{2t-1}\mathbf{x}(\tau)$ and the convexity of $f(\mathbf{x})$; (b) follows from (9); (c) follows from the Cauchy-Schwartz inequality; and (d) is true because

$\|\mathbf{Q}(2t) - \mathbf{Q}(t)\| \leq \frac{2V}{\sqrt{t}}\sqrt{\frac{\theta}{L_q}}, \forall t \geq T_q$ by Corollary 4 and $\|\mathbf{Q}(2t)\| \leq \sqrt{\|\mathbf{Q}(0)\|^2 + V^2\|\boldsymbol{\lambda}^*\|^2} + V\|\boldsymbol{\lambda}^*\| = \eta$ by Lemma 8. ■

Theorem 5: Assume problem (1)-(3) satisfies Assumptions 1-4. Let $\overline{\mathbf{x}}(t), t \in \{1, 2, \dots\}$ be the sequence yielded by Algorithm 3. If $V \geq \gamma$, then $g_k(\overline{\mathbf{x}}(2t)) \leq \frac{1}{t^{3/2}} \frac{2V\sqrt{\theta}}{\sqrt{L_q}}, \forall k \in \{1, 2, \dots, m\}, \forall t \geq T_q$.

Proof: Fix $t > T_q$ and $k \in \{1, 2, \dots, m\}$. We have $g_k(\overline{\mathbf{x}}(2t)) \overset{(a)}{\leq} \frac{1}{t}\sum_{\tau=t}^{2t-1} g_k(\mathbf{x}(\tau)) \overset{(b)}{\leq} \frac{1}{t}\big(Q_k(2t) - Q_k(t)\big) \leq \frac{1}{t}\|\mathbf{Q}(2t) - \mathbf{Q}(t)\| \overset{(c)}{\leq} \frac{2V}{t^{3/2}}\sqrt{\frac{\theta}{L_q}}$, where (a) follows from the convexity of g_k; (b) follows from Lemma 7; and (c) follows from Corollary 4. ■

C. Network Utility Maximizations

Consider a network with m links and n flow streams. Let $\{b_1, b_2, \dots, b_m\}$ be the capacities of each link and $\{x_1, x_2, \dots, x_n\}$ be the rates of each flow stream. Let $\mathcal{N}(k) \subseteq 1, 2, \dots, n, 1 \leq k \leq m$ be the set of flow streams that use link k. This problem is to maximize the utility function $\sum_{i=1}^n c_i \log(x_i)$ with $c_i > 0, \forall 1 \leq i \leq n$, which represents a measure of network fairness [14], subject to the capacity constraint of each link. This problem is known as the network utility maximization (NUM) problem and can be formulated as follows:

$$\min \quad \sum_{i=1}^{n} -c_i \log(x_i) \tag{10}$$

$$\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b} \tag{11}$$

$$\mathbf{x} \geq \mathbf{0} \tag{12}$$

where $c_i > 0, \forall 1 \leq i \leq n$ and $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ represents the link capacity constraints. Note that $\mathbf{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_n]$ is a 0-1 matrix of size $m \times n$ such that $a_{ij} = 1$ if and only if flow x_j uses link i; and $\mathbf{b} > \mathbf{0}$.

Note that problem (10)-(12) satisfies Assumptions 1 and 2. By Theorems 1 and 2, Algorithm 1 solves this problem with $O(\frac{1}{\epsilon})$ convergence time. The next theorem shows sufficient conditions under which Algorithm 3 can be applied to solve this problem and yields a better $O(\frac{1}{\epsilon^{2/3}})$ convergence time.

Theorem 6: The network utility maximization problem (10)-(12) has the following properties:

1) Let $b^{\max} = \max_{1 \leq i \leq n} b_i$ and $\mathbf{x}^{\max} > \mathbf{0}$ such that $x_i^{\max} > b^{\max}, \forall i \in \{1, \dots, n\}$. The network utility maximization problem (10)-(12) is equivalent to the following problem

$$\min \quad \sum_{i=1}^{n} -c_i \log(x_i) \tag{13}$$

$$\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b} \tag{14}$$

$$\mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{\max} \tag{15}$$

2) Let $\mathbf{x}^*$ be an optimal solution. Assume $\mathbf{A}\mathbf{x}^* \leq \mathbf{b}$ has m' rows that hold with equality, and let $\mathbf{A}'$ be the $m' \times n$ submatrix of $\mathbf{A}$ corresponding to these active rows. If $\text{rank}(\mathbf{A}') = m'$, then Assumptions 1-4 hold for this problem. That is, Algorithm 3 for problem (13)-(15) ensures error decays like $O(\frac{1}{t^{3/2}})$ and provides an ϵ-optimal solution with convergence time $O(\frac{1}{\epsilon^{2/3}})$.
3) [2] If $\text{rank}(\mathbf{A}) = m$, then Algorithm 3 for problem (13)-(15) ensures error decays like $O\big(\frac{1}{t}\big(1 - \frac{L_c}{V}\big)^{t/4}\big)$ and provides an ϵ-optimal solution with convergence time $O(\log(\frac{1}{\epsilon}))$.

Proof: See [12] for details. ■

V. Numerical Results

Consider the simple NUM problem described in Figure 1. Let x_1, x_2 and x_3 be the data rates of stream $1, 2$ and 3 and the network utility be minimizing $-\log(x_1) - 2\log(x_2) - 3\log(x_3)$. It can be checked that capacity constraints other than $x_1 + x_2 + x_3 \leq 10, x_1 + x_2 \leq 8$, and $x_2 + x_3 \leq 8$ are redundant. By Theorem 6, the NUM problem can be formulated as follows:

$$\begin{aligned} \min \quad & -\log(x_1) - 2\log(x_2) - 3\log(x_3) \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{\max} \end{aligned}$$

where $\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$, $\mathbf{b} = [10, 8, 8]^T$ and $\mathbf{x}^{\max} = [11, 11, 11]^T$. The optimal solution to this NUM problem is $x_1^* = 2, x_2^* = 3.2, x_3^* = 4.8$ and the optimal value is -7.5253. Note that the second capacity constraint $x_1 + x_2 \leq 8$ is loose and the other two capacity constraints are tight.

Since the objective function is decomposable, the drift-plus-penalty algorithm can yield a distributed solution. This is why the drift-plus-penalty algorithm, or equivalent, the dual subgradient algorithm (which makes similar decisions but traditionally does not take primal averages), is widely used to solve NUM problems [15]. It can be checked that the objective function is strongly convex with modulus $\alpha = \frac{2}{121}$ on $\mathcal{X} = \{\mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{\max}\}$ and each constraint $g_k(\cdot), 1 \leq k \leq 3$ is Lipschitz continuous with modulus $\beta = \sqrt{3}$ on $\mathcal{X}$. Figure 2 shows the values of objective and constraint functions yielded by Algorithm 1 with $V = \frac{2\beta^2}{\alpha} = 363$ and $Q_1(0) = Q_2(0) = Q_3(0) = 0$. By Theorem 1, we know $f(\overline{\mathbf{x}}(t)) \leq f(\mathbf{x}^*), \forall t > 0$ and this is verified in Figure 2. To verify the convergence time of constraint violations, Figure 3 plots $g_1(\overline{\mathbf{x}}(t))$, $g_2(\overline{\mathbf{x}}(t))$, $g_3(\overline{\mathbf{x}}(t))$ and $1/t$ with both x-axis and y-axis in $\log_{10}$ scales. We observe that the curves of $g_1(\overline{\mathbf{x}}(t))$ and $g_3(\overline{\mathbf{x}}(t))$ are parallel to the curve of $1/t$ for large t. Note that constraint $g_1(\overline{\mathbf{x}}(t)) \leq 0$ is satisfied early because it is loose. Figure 3 verifies that the error of Algorithm 1 decays like $O(1/t)$ and suggests that it is actually $\Theta(1/t)$ for this NUM problem.

[2] In this part, the dual function of problem (13)-(15) is locally strongly concave, which implies locally quadratic. In this case, Algorithm 3 ensures error decays like $O\big(\frac{1}{t}\big(1 - \frac{L_c}{V}\big)^{t/4}\big)$, where L_c is the radius of the locally strongly concave property, i.e., a parameter similar to L_q in Assumption 4. See [12] for detailed discussions.

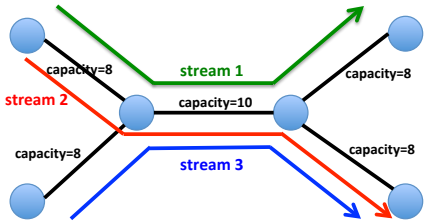

Fig. 1. A simple NUM problem with 3 flow streams

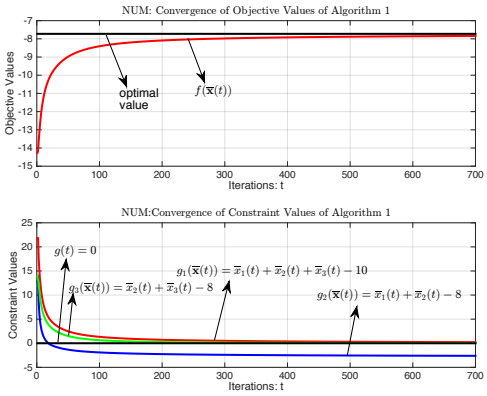

Fig. 2. The convergence of Algorithm 1 for a NUM problem.

Note that $\text{rank}(\mathbf{A}) = 3$. By Theorem 6, this NUM problem satisfies Assumption 1-4. By Lemma 9, the dual function of this NUM problem is smooth with modulus $\gamma = 422$. So we apply Algorithm 3 with $V = \max\{\frac{2\beta^2}{\alpha}, \gamma\} = 422$ and $Q_1(0) = Q_2(0) = Q_3(0) = 0$ to this NUM problem. Figure 4 verifies the convergence of the objective and constraint functions. Figure 5 verifies that the error of Algorithm 3 decays at least exponentially as proven in Theorem 6.

REFERENCES

[1] M. J. Neely, *Stochastic network optimization with application to communication and queueing systems*. Morgan & Claypool Publishers, 2010, vol. 3, no. 1.

[2] ——, "Distributed and secure computation of convex programs over a network of connected processors," in *DCDIS Conference Guelph*, July 2005.

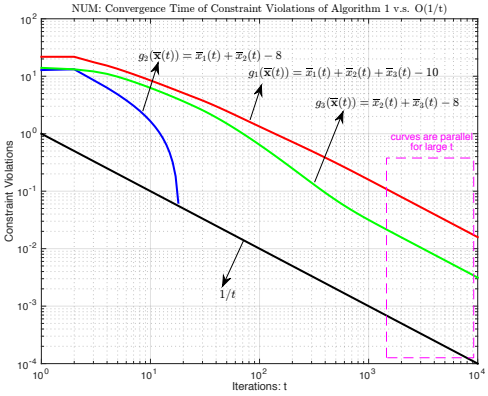

Fig. 3. The convergence time of Algorithm 1 for a NUM problem.

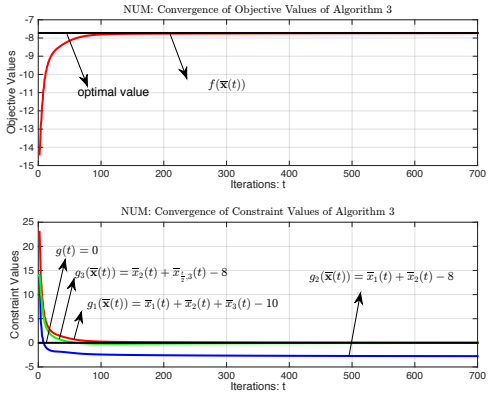

Fig. 4. The convergence of Algorithm 3 for a NUM problem.

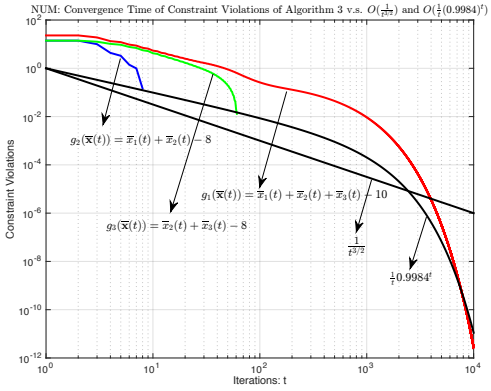

Fig. 5. The convergence time of Algorithm 3 for to a NUM problem.

[3] L. Huang and M. J. Neely, "Delay reduction via lagrange multipliers in stochastic network optimization," *IEEE Trans. Autom. Control*, vol. 56, no. 4, pp. 842–857, 2011.

[4] S. Supittayapornpong, L. Huang, and M. J. Neely, "Time-average optimization with nonconvex decision set and its convergence," in *IEEE Conference on Decision and Control (CDC)*, 2014.

[5] O. Shamir and T. Zhang, "Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes," in *ICML*, 2013.

[6] M. J. Neely, "A simple convergence time analysis of drift-plus-penalty for stochastic optimization and convex programs," *arXiv preprint arXiv:1412.079*, 2014.

[7] A. Nedic and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1757–1780, 2009.

[8] I. Necoara and V. Nedelcu, "Rate analysis of inexact dual first-order methods application to dual decomposition," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1232–1243, May 2014.

[9] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of Convex Analysis*. Springer, 2001.

[10] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[11] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*. Wiley-Interscience, 2006.

[12] H. Yu and M. J. Neely, "On the convergence time of the drift-plus-penalty algorithm for strongly convex programs," *arXiv preprint arXiv:1503.06235*, 2015.

[13] Y. Nesterov, *Introductory lectures on convex optimization*. Springer Science & Business Media, 2004.

[14] F. P. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, no. 1, pp. 33–37, 1997.

[15] S. H. Low and D. E. Lapsley, "Optimization flow control—I: basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861–874, 1999.