Adaptive control mechanisms in gradient descent algorithms

Andrea Iannelli

***Abstract*— The problem of designing adaptive stepsize sequences for the gradient descent method applied to convex and locally smooth functions is studied. We take an adaptive control perspective and design update rules for the stepsize that make use of both past (measured) and future (predicted) information. We show that Lyapunov analysis can guide in the systematic design of adaptive parameters striking a balance between convergence rates and robustness to computational errors or inexact gradient information. Theoretical and numerical results indicate that closed-loop adaptation guided by system theory is a promising approach for designing new classes of adaptive optimization algorithms with improved convergence properties.**

I. INTRODUCTION

Convex optimization algorithms are at the core of many established methodologies in control and reinforcement learning, for example receding horizon control [1] and convex Q-learning [2]. In all these applications, one typically requires algorithms that are fast (to reduce computational time) and robust (e.g., to be less sensitive to error in the problems data). Developing systematic methods to analyze these properties and design for them is a very important step to make these tools of more widespread and dependable use in applications. Because iterative optimization algorithms can be seen as open dynamical systems, tools and viewpoints from control theory can be a valid standpoint to approach these tasks [3], [4]. Towards this goal, we consider here the basic unconstrained optimization setting where the objective f is convex and only locally smooth. We approach the design of a stepsize sequence $(\alpha_k)_{k\in\mathbb{N}}$ as an adaptive control problem where the goal is to guarantee that the interconnection between the algorithm and the stepsize law converges to the minimizer set. We do this while capturing performance and robustness trade-off of this adaptive closed-loop.

Even though gradient descent (GD) methods are standard in optimization, analysis and design of varying stepsizes is an active area of research. For the case of L_{s}-smooth objective, one line of work has developed pre-defined sequences of *large* stepsizes (i.e., with instances where these are larger than $\frac{2}{L_{\mathrm{s}}}$) and showed that they can accelerate convergence [5]. While convergence guarantees and the sequence of stepsizes generally depend on the value of the stopping time which must be selected a-priori, very recently [6] showed that this strategy provably achieves anytime convergence guarantees that strictly improves upon the classic $\mathcal{O}(\frac{1}{k})$. Besides the restriction to smooth objectives, the fundamental idea of these approaches is to pre-compute the sequence of stepsizes independently of f (except for its smoothness constant) and of the initial iterate, thus the stepsize sequence is effectively an open-loop input to the GD dynamics. In another line of work, the case of locally smooth objective has been addressed by proposing stepsizes that adapt to the local geometry [7]–[9], including extension to the proximal gradient method for composite problems. In these works, feedback from the current and past iterates is leveraged to estimate a sequence of local smoothness constants which are used in the stepsize's update. By doing so, the standard requirement of global smoothness is lifted, and progress to the optimal value is empirically much faster even though a better rate than $\mathcal{O}(\frac{1}{k})$ could not proved.

Inspired by this prior work, we make a first attempt to combine feedback and feedforward (or predictive) actions in the selection of stepsize laws for gradient-based methods applied to locally smooth objective. We approach the problem similarly to the analysis-informed design of adaptive controllers whereby appropriately constructed Lyapunov functions guide the selection of parameters so that boundedness and convergence guarantees can be established. We also investigate questions on robustness of such adaptive systems, and recognize components of the designed adaptive mechanisms that can mediate between performance and robustness, e.g., to errors in the gradient information. Besides the new technical results, we see as central contribution of this work the consideration of the role of feedback and predictive actions in adaptive optimization algorithms. Even though in a different context, connections between convex optimization and adaptive control were also studied in [10]. For space constraints and readability, the proofs of all new technical results are either in the Appendix or in the extended version of this paper [11].

Notation: We denote by $\langle x, y\rangle$ the standard Euclidean inner product in $\mathbb{R}^n$ and by $\|\cdot\|$ its induced norm. We use $\mathbb{N}$ for the set of natural numbers. Given $u, v \in \mathbb{R}^n$, we refer to the following as the Pythagoras identity

$$\|u\|^2 = \|u - v\|^2 - \|v\|^2 + 2\langle u, v\rangle$$

The convex hull of a set of points is denoted by conv.

II. PRELIMINARIES

We consider the unconstrained problem

$$\min_{x\in\mathbb{R}^n} \; f(x) \tag{1}$$

and we denote with $X^\star$ its set of minimizers and with $f^\star$ its optimal value.

The author is with the University of Stuttgart, Institute for Systems Theory and Automatic Control, 70550 Stuttgart, Germany iannelli@ist.uni-stuttgart.de.

Assumption 1: We make the following standing assumptions:

- $f : \mathbb{R}^n \to \mathbb{R}$ is *locally smooth*, i.e., f is differentiable and, for every convex and compact set $D \subset \mathbb{R}^n$, there exists $L_D \in (0, \infty)$ such that $\forall x, y \in D$:

$$\|\nabla f(y) - \nabla f(x)\| \leqslant L_D \|y - x\|. \tag{2}$$

- f is *convex*, i.e., $\forall x, y \in \mathbb{R}^n$

$$f(y) \geqslant f(x) + \langle \nabla f(x), y - x \rangle. \tag{3}$$

- $X^\star \neq \varnothing$ and $f^\star > -\infty$

Local smoothness, or equivalently local Lipschitz continuity of the gradient, defines a rather general class of functions. For example, any twice differentiable f is locally smooth since (2) holds with $L_D = \max_{v \in D} \|\nabla^2 f(v)\|$, which is finite due to continuity of the Hessian and compactness of D.

One of the most popular methods to solve (1) is gradient descent (GD), which generates a sequence of iterate $(x_k)_{k \in \mathbb{N}}$ by applying the following simple recursion

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k), \quad k \in \mathbb{N}, \tag{4}$$

starting from a given x_0 and choosing an appropriate sequence of stepsizes $(\alpha_k)_{k \in \mathbb{N}}$. We will denote by $F_k := f(x_k) - f^\star$ the optimality gap at iteration k.

Standard convergence analyses of GD assume that f is (globally) L_s-*smooth*, i.e., f is differentiable and there exists $L_\mathrm{s} \in (0, \infty)$ such that $\forall x, y \in \mathbb{R}^n$

$$\|\nabla f(y) - \nabla f(x)\| \leqslant L_\mathrm{s} \|y - x\|, \tag{5}$$

or equivalently $\forall x, y \in \mathbb{R}^n$

$$f(y) \leqslant f(x) + \langle \nabla f(x), y - x \rangle + \frac{L_\mathrm{s}}{2} \|y - x\|^2. \tag{6}$$

In this case, one can guarantee global convergence of (4) to an element of $X^\star$ by restricting the choice of stepsize [12]. Most of the analyses show convergence with constant stepsizes in the ranges $L_\mathrm{s}\alpha \in (0, 1]$ or $L_\mathrm{s}\alpha \in [1, 2)$. As a summary of the available analysis results, we provide a Lyapunov-based analysis that encompasses any time-varying stepsize sequence satisfying[1]: $L_\mathrm{s}\alpha_k \in (0, 2)$; $\limsup_{k \to +\infty} L_\mathrm{s}\alpha_k \neq 2$; $\liminf_{k \to +\infty} \alpha_k \neq 0$.

Theorem 1: Let $(x_k)_{k \in \mathbb{N}}$ be a sequence generated by the GD method (4) applied to a L_s-smooth function f also satisfying Assumption 1 with any time-varying stepsize sequence satisfying $L_\mathrm{s}\alpha_k \in (0, 2), k \in \mathbb{N}$. Define the function

$$V_k^\mathrm{s}(x^\star) := \|x_k - x^\star\|^2, \quad x^\star \in X^\star. \tag{7}$$

Then for any $x^\star \in X^\star$, $k \in \mathbb{N}$ and $x_0 \in \mathbb{R}^n$, it holds[2]

$$V_{k+1}^\mathrm{s} - V_k^\mathrm{s} \leqslant -\frac{\alpha_k}{L_\mathrm{s}} (2 - \alpha_k L_\mathrm{s}) (1 + \alpha_k L_\mathrm{s}) \|\nabla f(x_{k+1})\|^2 \tag{8a}$$

$$F_k \leqslant \frac{2 F_0 V_0^\mathrm{s}}{2 V_0^\mathrm{s} + \frac{c_1}{L_\mathrm{s}} F_0 k}, \tag{8b}$$

$$\|\nabla f(x_k)\| \leqslant \frac{L_\mathrm{s}}{c_2} \frac{\|x_0 - x^\star\|}{k}, \tag{8c}$$

where $c_1, c_2 \in (0, \infty)$ are problem-independent constants. The proof builds on standard results [12], [13], but an analysis encompassing arbitrary sequences $(\alpha_k)_{k \in \mathbb{N}}$, and yielding (8) is not present in the literature. Precisely: (8a) gives the existence of a Lyapunov function for (4) which can be used to show boundedness of the iterates and global convergence to the set $X^\star$; (8b) and (8c) show convergence rates for function values and gradient which have no worse dependencies on L_s and k than those found in the literature and focusing on constant or smaller ranges of stepsizes [12], [13].

The goal of this work is to develop a GD algorithm achieving similar guarantees to Theorem 1 under the standing Assumption 1 only. The only degree of freedom in (4) is the stepsize sequence, and we approach its design as an adaptive control problem where $(\alpha_k)_{k \in \mathbb{N}}$ is an input that can be chosen based on feedback and feedforward information to steer the iterate towards the set $X^\star$.

III. Adapting stepsize to local smoothness

A. A local smoothness estimate

The intuitive idea for using the GD method without assuming (global) smoothness of the objective is to adapt the stepsize sequence to the local geometry of the cost function. A natural measure of it along the GD iterates is the local smoothness estimate

$$\begin{aligned} L_k = L(x_{k+1}, x_k) &:= \frac{\|\nabla f(x_{k+1}) - \nabla f(x_k)\|}{\|x_{k+1} - x_k\|}, \\ &= \frac{\|\nabla f(x_k - \alpha_k \nabla f(x_k)) - \nabla f(x_k)\|}{\alpha_k \|\nabla f(x_k)\|}. \end{aligned} \tag{9}$$

At iterate k, this estimate depends both on past information through x_k (feedback) and one-step ahead future information through x_{k+1} (feedforward).

While after Theorem 1 it would be tempting to conjecture that a stepsize sequence satisfying $L_k\alpha_k \in (0, 2)$ could satisfy our goal, the following results instill caution.

Lemma 1: Consider a convex and differentiable f.

(i) Given $L(y, x)$ defined in (9), $\forall y, x \in \mathbb{R}^n$

$$f(y) \leqslant f(x) + \langle \nabla f(x), y - x \rangle + L(y, x) \|y - x\|^2. \tag{10}$$

(ii) The sequence generated by (4) GD satisfies

$$F_{k+1} \leqslant F_k - (1 - L_k\alpha_k) \alpha_k \|\nabla f(x_k)\|^2. \tag{11}$$

(iii) If $L_k\alpha_k \in \left(0, \frac{1}{2}\right)$, then

$$V_{k+1}^\mathrm{s}(x^\star) - V_k^\mathrm{s}(x^\star) \leqslant -2\alpha_k F_{k+1}. \tag{12}$$

[1] We make the last two technical restrictions to simplify some steps in the derivation of the rates in view of the very unrestricted range of time-varying stepsizes; this is without loss of generality, e.g., [13, Section 4] consider the specific case where $L_\mathrm{s}\alpha_k \in [1, 2)$, $\lim_{k \to +\infty} L_\mathrm{s}\alpha_k \to 2$.

[2] We omit the argument of V^s for brevity and formatting reasons.

Item (i) shows that, compared to the global smoothness constant L_{s}, the local smoothness estimate L_k can be larger up to a factor of two, compare (6) and (10). A direct consequence of this is item (ii), where Eq. (11) clearly implies that a guaranteed function value decrease holds if $L_k\alpha_k \in (0,1)$. Finally, item (iii) shows that setting the stepsize to $L_k\alpha_k \in \left(0,\frac{1}{2}\right)$ guarantees the existence of the same Lyapunov function V^{s} in (7).

We note that the sufficient condition $L_k\alpha_k \in (0,1)$ for function value decrease in item (ii) is, in general, also necessary. Indeed (11) follows immediately from (10), which has recently been shown to be tight [14].

Lemma 2: [14, Proposition 2.3] Given $\beta \in (0,1)$, there exist a convex and differentiable f_β, $y, x \in \mathbb{R}^n$ such that

$$f_\beta(y) \geqslant f_\beta(x) + \langle \nabla f_\beta(x), y - x\rangle + \beta L(y,x)\|y-x\|^2. \quad (13)$$

B. Adaptive feedback-feedforward gradient descent

We propose here a novel stepsize update law that uses L_k to adapt to the local geometry by combining feedback and feedforward mechanisms.

The update law reads as:

$$\begin{aligned} &\alpha_k = \min\left\{\alpha_k^{(1)}, \alpha_k^{(2)}\right\}, \\ \text{where}\quad &\alpha_k^{(1)} = \frac{\gamma_k}{L_k}, \quad \alpha_k^{(2)} = \frac{\alpha_{k-1}}{\gamma_k^2}\left(\frac{1-\gamma_k^2}{1-\gamma_{k-1}^2}\right), \\ &(\gamma_k)_{k\in\mathbb{N}} \subset (0,1). \end{aligned} \quad (14)$$

where $(\gamma_k)_{k\in\mathbb{N}}$ is a scalar sequence of parameters inside the specified range. The stepsize α_k is chosen as the smallest between the two upper bounds $\alpha_k^{(1)}$ and $\alpha_k^{(2)}$. The former one is the intuitive choice discussed in III-A. It is worth observing that, whenever $\gamma_k \in \left(\frac{1}{2},1\right)$, $L_k\alpha_k^{(1)} \in (0,\gamma_k)$. That is, α_k in (14) can be up to two times larger than the bound in item (iii) of Lemma 1 guaranteeing the existence of V^{s} in item (iii), and can become as large as the fundamental limit in item (ii) of Lemma 1. The bound $\alpha_k^{(2)}$ instead limits the increase of stepsize across two consecutive iterations and does not depend directly on the local geometry, but only on the last value of the stepsize. As it will be shown in Section III-C, this bound also gives some inherent robustness to the algorithm. Intuitively, the first constraint is active in regions of the variable space where f changes rapidly (or is *less smooth*), whereas the second is active when, due to the function's *flatness*, the stepsize would tend otherwise to overly increase. Finally, the parameters $(\gamma_k)_{k\in\mathbb{N}}$ are a tuning knob to navigate the speed of convergence vs. robustness trade-off discussed later. While any value (constant or time-varying) in $(0,1)$ is valid, one intriguing option is to use them as additional adaptive parameters. For example, they could be modified online so that the two upper bounds are as close as possible and thus α_k is maximized at every iteration. This option will be further explored in Section IV.

The interconnection between the classic GD recursion (4) and the adaptation law (14) is shown schematically in Figure 1 and we will refer to in the following for brevity as AFFGD (adaptive feedback-feedforward gradient descent).

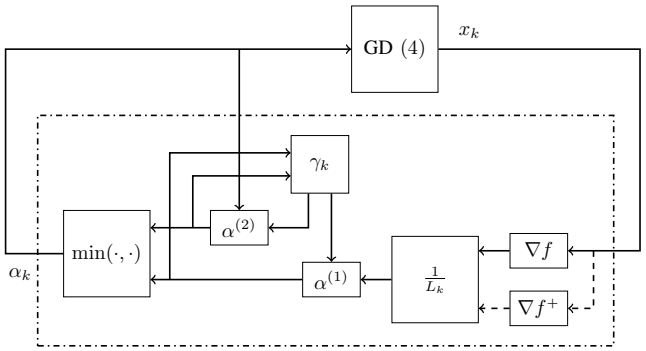

Fig. 1. The adaptive feedback-feedforward gradient descent algorithm (feedforward paths with dashed line).

The following result gives convergence guarantees for AFFGD by using a Lyapunov analysis which, albeit departing from the one used in Theorem 1 and under weaker requirements, yield qualitatively similar results.

Theorem 2: Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by the GD method (4) applied to a function f satisfying Assumption 1 and with stepsize law (14). Define the function

$$V_k^{\mathrm{a}}(x^\star) := \|x_k - x^\star\|^2 + \frac{2\alpha_{k-1}}{1-\gamma_{k-1}^2}F_k, \quad x^\star \in X^\star. \quad (15)$$

Then for any $x^\star \in X^\star$ and $k \in \mathbb{N}$, it holds that

$$V_{k+1}^{\mathrm{a}} - V_k^{\mathrm{a}} \leqslant -\frac{2\gamma_k^2}{1-\gamma_k^2}\left(\alpha_k^{(2)} - \alpha_k\right)F_k - v_k \quad (16a)$$

$$F_k \leqslant \frac{\|x_0 - x^\star\|^2 + 2\alpha_0\frac{\gamma_0^2}{1-\gamma_0^2}F_0}{2\sum_{i=1}^{k-1}\alpha_i} \quad (16b)$$

$$\|\nabla f(x_k)\| \xrightarrow{k\to\infty} 0 \quad (16c)$$

where

$$v_k := \frac{\alpha_k}{L_{D_k}}\|\nabla f(x_k)\|^2 + \frac{\alpha_k^2}{1-\gamma_k^2}\|\nabla f(x_{k+1})\|^2 \quad (17)$$

and L_{D_k} is the local smoothness constant over a convex and compact set D_k containing x_k and $x^\star$.

Eq. (16a) shows that function V^{a} is a valid Lyapunov function for the closed-loop dynamics (4)-(14) which gives boundedness of the iterates, asymptotic optimality (16c) and global convergence to the set $X^\star$. Eq. (16b) gives a guaranteed last iterate convergence. As shown in the proof (cf. Eq. 36), the stepsize sequence $(\alpha_k)_{k\in\mathbb{N}}$ is separated from 0, and thus (16b) yields immediately a guaranteed convergence rate of $\mathcal{O}(\frac{1}{k})$, as in the standard smooth case. However, the denominator of (16b) points out that we can accelerate convergence by maximizing the sum of stepsizes. We can achieve this by adapting online the parameters $(\gamma_k)_{k\in\mathbb{N}}$ to make $\alpha_k^{(1)}$ and $\alpha_k^{(2)}$ as close as possible. Compared to the recent literature on adaptive gradient descent [7], [9], AFFGD provides convergence rate guarantees on the last iterate (16b), a Lyapunov function with the two standard terms relating to suboptimality distances (15), and a larger available upper bound on the stepsize with respect to the local geometry, compare with [8, Table 1]. On the other hand, it is also important to recognize that the computation

of $\alpha_k^{(1)}$ is a disadvantage of the proposed formulation as it involves forward prediction. While for some special cases this can be done without extra computation (e.g., when f is quadratic with Hessian M, then L_k only depends on M and the current gradient), in general this requires a linesearch procedure that can be easily automated but might result in a more expensive per-iteration cost. Most importantly, we show next that limiting the growth rate of α_k (e.g., as currently done via $\alpha_k^{(2)}$) provides robustness to inexact gradients, for example due to errors in the linesearch procedure.

C. Robustness

The system theoretic view on AFFGD (Figure 1) prompts the question of robustness of the closed-loop. For example, one can consider stepsize updates where the first upper bound in (14), involving forward prediction, is not exactly satisfied

$$\alpha_k \leqslant \tilde{\alpha}_k^{(1)} := \frac{\gamma_k}{a_k L_k}, \quad a_k \in (0,1). \tag{18}$$

Because $\tilde{\alpha}_k^{(1)} > \alpha_k^{(1)}$, this can capture errors in the gradient information (e.g., noisy evaluation, inexactness of the linesearch) inversely proportional to the parameter a_k.

Let us define the scaled sequence $(\tilde{\gamma}_k)_{k\in\mathbb{N}}$ with $\tilde{\gamma}_k := \frac{\gamma_k}{a_k} > \gamma_k$. If $a_k \in (\gamma_k, 1)$, then $\tilde{\gamma}_k \in (\gamma_k, 1) \forall k \in \mathbb{N}$. Then we can simply observe that we can still guarantee the results of Theorem 2 if we tighten the second upper bound in (14) correspondingly, that is we impose

$$\tilde{\alpha}_k^{(2)} := \frac{\alpha_{k-1}}{\tilde{\gamma}_k^2}\left(\frac{1-\tilde{\gamma}_k^2}{1-\tilde{\gamma}_{k-1}^2}\right). \tag{19}$$

Indeed the conditions prescribed for the stepsize (14) are satisfied with respect to the scaled sequence $(\tilde{\gamma}_k)_{k\in\mathbb{N}} \subset (0,1)$. This observation provides two insights. First, limiting the growth rate of α_k (through $\alpha_k^{(2)}$) adds robustness to inexact gradient information, which also contributes to understanding the role of the second upper bound (14). Second, the tuning parameters $(\gamma_k)_{k\in\mathbb{N}}$ provide a means to navigate the trade-off between speed of convergence (when it is chosen adaptively to make $\alpha_k^{(1)}$ and $\alpha_k^{(2)}$ close and thus maximize the rate of convergence) and robustness (when it is chosen away from 1 to have robustness against perturbations $a_k \in (\gamma_k, 1)$).

It is natural to ask what happens when $a_k \in (0, \gamma_k]$, which models scenarios where perturbations are large or γ_k is chosen close to 1. In this case the analysis in Theorem 2 does not apply but the following result provides a first answer.

Lemma 3: Let $(x_k)_{k\in\mathbb{N}}$ be a sequence generated by the GD method (4) applied to a function f satisfying Assumption 1 and with stepsize law

$$\alpha_k = \frac{\gamma_k}{a_k L_k}, \quad \gamma_k \in (0,1),\ a_k \in (0, \gamma_k]. \tag{20}$$

Define the function

$$V_k^{\mathrm{p}}(x^\star) := \|x_k - x^\star\|^2 + \frac{\alpha_{k-1}}{\alpha_k}\|x_{k+1} - x_k\|^2, \quad x^\star \in X^\star. \tag{21}$$

Then for any $x^\star \in X^\star$ and $k \in \mathbb{N}$, it holds that

$$V_{k+1}^{\mathrm{p}} - V_k^{\mathrm{p}} \leqslant -\left(\frac{\alpha_{k-1}^2}{\alpha_k^2} - \frac{\gamma_k^2}{a_k^2}\right)\|x_{k+1} - x_k\|^2 - 2\alpha_k F_{k+1}. \tag{22}$$

Note that Eq. (20) allows perturbations to even determine stepsizes that results in $L_k\alpha_k > 1$. Condition (22) shows that, even in such extreme scenarios, limiting the growth rate of α_k guarantees *robust* convergence. Indeed, if in addition to (20) it holds

$$\alpha_k \leqslant \frac{a_k}{\gamma_k}\alpha_{k-1} \tag{23}$$

then the decrease of V^{p} is guaranteed. While (23) is restrictive as for large perturbations it effectively prevents α_k from increasing, it provides another important characterization of the robustifying effect of limiting the growth rate even in this large perturbations regime. Moreover, we observe that requiring (23) is not necessary because we are ignoring the second negative term on the r.h.s. of (22). We conjecture that positive growth rate conditions allowing $L_k\alpha_k > 1$ are possible, but for space reasons we leave this for future work.

IV. Numerical Study

We study and compare numerically the performance of the proposed AFFGD algorithm[3]. We consider logistic regression, that is, given N features $s_i \in \mathbb{R}^n$ and labels $y_i = \pm 1$, the goal is to find a linear classifier $x^\star \in \mathbb{R}^n$ by solving

$$\min_{x\in\mathbb{R}^n} \ \frac{1}{N}\sum_1^N \log(1 + \exp(-y_i x^\top s_i)). \tag{24}$$

The objective is convex and globally smooth with $L_{\mathrm{s}} = \frac{1}{4N}\sigma_{\max}(S)^2$, where $S \in \mathbb{R}^{N\times n}$ is the feature matrix. The application of GD to (24) has recently received attention [15] due to the complex behavior of the iterates for *large* stepsizes (i.e., greater than $\frac{2}{L_{\mathrm{s}}}$) with not linearly-separable data. To test this regime, we generate random data with $N = 50$, $n = 2$.

In a first set of results displayed in Figure 2, we compare five GD algorithms (4) that differ for the step-size: *GD* uses the classical choice $\alpha_k = \frac{1}{L_{\mathrm{s}}}$ ($\simeq 1$ in this case); *GD TV* is the dynamic update rule proposed in [13, Theorem 4] whereby $L_{\mathrm{s}}\alpha_k \in [1, 2)$ and the stepisze is monotonically increased according to a pre-determined law with $L_{\mathrm{s}}\alpha_k \to 2$; *AdGD* [9, Algorithm 1] and *AdaGM* [7, Algorithm 2] are recently proposed adaptive GD schemes which also adapt to the local geometry by only using past information. Finally, *AFFGD* is the update rule proposed in this work (14) with a constant tuning parameter $\gamma = 0.7$ and arbitrary initialization α_{-1}.

The results show the clear impact that adaptation has on accelerating convergence. Even though the problem is globally smooth and the first two methods are guaranteed to asymptotically converge, they are slow compared to the adaptive ones. In that regard, the right plot shows that the sequence of adaptive stepsizes, during the iterations before convergence, sum up at a rate which is faster than linear. We also observe that *AFFGD*, implemented here using a simple

[3]Codes to reproduce the results are available at: `https://github.com/col-tasas/2025-AFFGD`

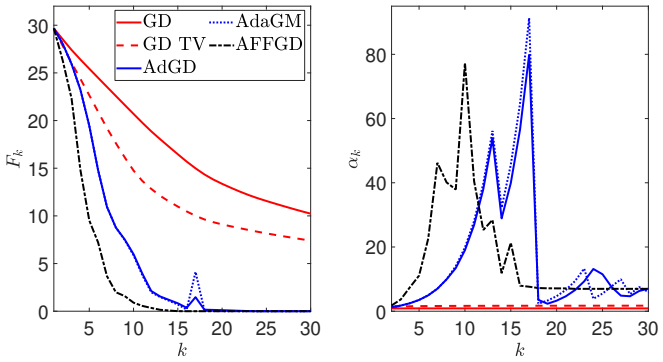

Fig. 2. Optimality gap and stepsizes of algorithms solving (24).

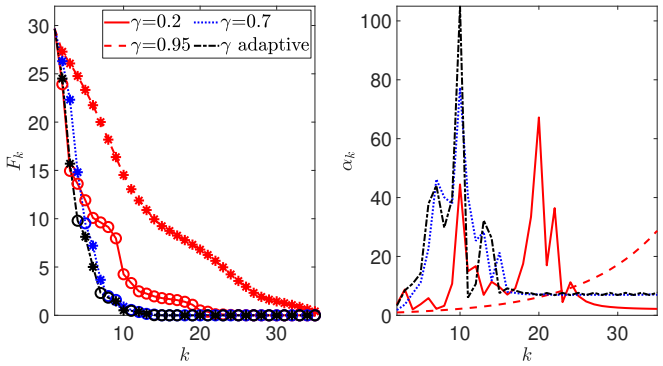

Fig. 3. Analysis of AFFGD for different choices of γ_k. Asterisk and circle markers denote points at which $\alpha_k = \alpha_k^{(1)}$ and $\alpha_k = \alpha_k^{(2)}$, respectively.

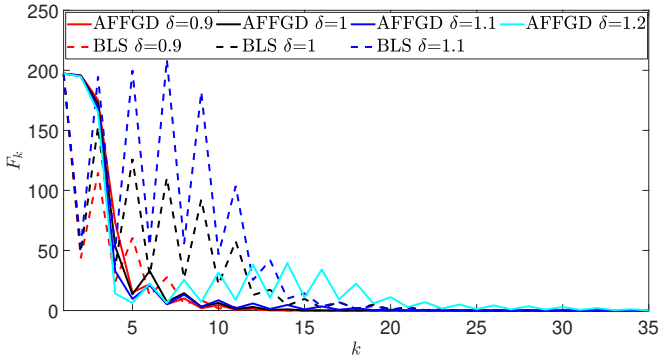

Fig. 4. Robustness of AFFGD vs. a simple backtracking strategy.

linesearch to verify the condition imposed by $\alpha_k^{(1)}$, markedly outperforms the other two methods at the cost of a slightly increased computational time ($0.13s$ against $0.1s$). We finally note that, when simulating *GD* with $\alpha > 20\frac{1}{L_s}$ and *GD TV* with $L_s\alpha_k \to 40$, we observed non-converging behaviours as described in [15]. This is interesting because these stepsizes are still quite smaller than those (succesfully) employed in many iterations by the adaptive schemes (see the y scale of the right plot in Figure 2).

We focus next in Figure 3 on AFFGD and investigate the effect of the sequence of tuning parameters $(\gamma_k)_{k\in\mathbb{N}}$. We compare three scenarios where this parameter is kept constant at some pre-defined value ($\gamma = \{0.2, 0.7, 0.95\}$) with the adaptive case where $\gamma_0 = 0.95$ and then it is changed adaptively using the simple recursion

$$\gamma_k = \begin{cases} \frac{1}{\theta}\gamma_{k-1}; & \alpha_{k-1} = \alpha_{k-1}^{(1)}, \\ \theta\gamma_{k-1}; & \alpha_{k-1} = \alpha_{k-1}^{(2)}, \end{cases} \quad k \in \mathbb{Z}_+ \qquad (25)$$

where $\theta = 0.9$ is a free parameter defining the strength of adaptation of γ_k. The rationale is to recursively update γ_k based on the last active constraint in order to determine similar values for the two upper bounds $\alpha_k^{(1)}$ and $\alpha_k^{(2)}$, and by doing so maximize the sum of α_k which, as shown by our analyses (16b), accelerates the convergence rate. The left plot shows, as expected, that $\gamma = \{0.2, 0.95\}$ yield low performance as they increase one of the bounds at the cost of strongly decreasing the other (which is then always active). On the contrary, the dynamic update rule (25) is able to recover from the bad initialization γ_0 and determine larger values of stepsize and faster progress than those achieved with $\gamma = 0.7$ (which was fine tuned offline).

Finally, we take a numerical perspective on the robustness of AFFGD. Guided by the analytical results in Section III-C, showing that limiting the growth rate of α_k through $\alpha_k^{(2)}$ robustifies the algorithm, we compare AFFGD with a simple backtracking line search (BLS) that sets $\alpha_k = \frac{\gamma_k}{L_k}$. We implement the update rule as $x_{k+1} = x_k - (1+\delta)\alpha_k\nabla f(x_k)$ to analyze the effect of numerical or gradient estimation errors quantified by the positive scalar δ. Figure 4 shows, in agreement with Lemma 3, that AFFGD (solid) is only marginally affected by such errors, while the convergence of BLS (dashed) degrades and is lost for $\delta > 1.1$.

V. CONCLUSIONS

We consider the design of adaptive stepsize sequences for gradient descent methods driven by local properties of the objective. We frame the problem as an instance of adaptive controller where the input (stepsize) to the plant (GD method) is computed as a combination of feeedback and predicted information. Theoretical results and numerical experiments support the idea of pursuing closed-loop adaptation to accelerate convergence while increasing robustness. Future directions include online optimization problems, where adaptation should capture both local geometry and instantaneous variations of the objective. We view the system theoretic framework in [16] as a promising starting point.

APPENDIX

Lemma 4: [17, Theorem 5.8] For any L_s-smooth and convex function f it holds $\forall x, y \in \mathbb{R}^n$

$$f(y) \geqslant f(x) + \langle \nabla f(x), y - x\rangle + \frac{1}{2L_s}\|\nabla f(y) - \nabla f(x)\|^2. \qquad (26)$$

Proof of Theorem 2

Eq. (16a). We start off by applying Pythagoras identity (for the equality) and using Lemma 4 (for the inequality)

$$\|x_{k+1} - x^\star\|^2 - \|x_k - x^\star\|^2 = -2\alpha_k\langle\nabla f(x_k), x_k - x^\star\rangle + \alpha_k^2\|\nabla f(x_k)\|^2 \leqslant -2\alpha_k F_k + \left(-\frac{\alpha_k}{L_{D_k}} + \alpha_k^2\right)\|\nabla f(x_k)\|^2 \qquad (27)$$

where $x^\star \in X^\star$. While we could have simply used convexity in the inequality (and drop the second term), local smoothness implies that Lemma 4 holds $\forall x, y \in D_k$, where D_k is

a convex and compact set containing x_k and $x^\star$, and L_{D_k} is the associated local smoothness constant (2). We now work on the third term, where we use again Pythagoras identity but now with $u = x_{k+1} - x_k$ and $v = x_{k+2} - x_{k+1}$ yielding

$$\begin{aligned}\|x_{k+1} - x_k\|^2 &= \alpha_k^2\|\nabla f(x_k)\|^2 = \alpha_k^2\|\nabla f(x_{k+1}) - \nabla f(x_k)\|^2 \\ &- \alpha_k^2\|\nabla f(x_{k+1})\|^2 + 2\alpha_k\langle \nabla f(x_{k+1}), x_k - x_{k+1}\rangle \\ &\leqslant \alpha_k^2 L_k^2\|x_{k+1} - x_k\|^2 - \alpha_k^2\|\nabla f(x_{k+1})\|^2 + 2\alpha_k(F_k - F_{k+1})\end{aligned} \tag{28}$$

where for the inequality we used the definition of L_k (first term) and convexity (third term). Using now the upper bound on α_k due to $\alpha_k^{(1)}$ (14), we get

$$(1-\gamma_k^2)\|x_{k+1} - x_k\|^2 \leqslant -\alpha_k^2\|\nabla f(x_{k+1})\|^2 + 2\alpha_k(F_k - F_{k+1}) \tag{29}$$

Because $(\gamma_k)_{k\in\mathbb{N}} \subset (0,1)$, we can divide the latter expression by $(1-\gamma_k^2)$ and plug this bound in (27) to obtain

$$\begin{aligned}\|x_{k+1} - x^\star\|^2 - \|x_k - x^\star\|^2 &\leqslant -\frac{2\alpha_k}{1-\gamma_k^2}F_{k+1} \\ &+ \left(\frac{2\alpha_k}{1-\gamma_k^2} - 2\alpha_k\right)F_k - v_k\end{aligned} \tag{30}$$

where v_k, defined in (17), is a positive term for all $x_k \notin X^\star$. Simple manipulations of (30) yield

$$\begin{aligned}V_{k+1}^{\mathrm{a}} - V_k^{\mathrm{a}} &\leqslant -2\left(\frac{\alpha_{k-1}}{1-\gamma_{k-1}^2} - \frac{\alpha_k\gamma_k^2}{1-\gamma_k^2}\right)F_k - v_k \\ &\overset{(14)}{=} -\frac{2\gamma_k^2}{1-\gamma_k^2}\left(\alpha_k^{(2)} - \alpha_k\right)F_k - v_k\end{aligned} \tag{31}$$

where the upper bound on α_k due to $\alpha_k^{(2)}$ (14) guarantees negativity of the first term.

Eq. (16b). We sum (30) for $k = 0, 1, ..., n-1$ and obtain

$$\begin{aligned}\|x_n - x^\star\|^2 + \frac{2\alpha_{n-1}}{1-\gamma_{n-1}^2}F_n + \sum_{k=0}^{n-2} w_k F_{k+1} \\ \leqslant \|x_0 - x^\star\|^2 + \left(\frac{2\alpha_0}{1-\gamma_0^2} - 2\alpha_0\right)F_0\end{aligned} \tag{32}$$

with

$$w_k := 2\left(\frac{\alpha_k}{1-\gamma_k^2} - \frac{\alpha_{k+1}}{1-\gamma_{k+1}^2} + \alpha_{k+1}\right). \tag{33}$$

Observe now that, because of the upper bound on α_k due to $\alpha_k^{(1)}$ (14) and $(\gamma_k)_{k\in\mathbb{N}} \subset (0,1)$, from item (ii) of Lemma 1 we have that for all iterations $F_k \geqslant F_{k+1}$. Note also that

$$\frac{2\alpha_{n-1}}{1-\gamma_{n-1}^2} + \sum_{k=0}^{n-2} w_k = \frac{2\alpha_0}{1-\gamma_0^2} + 2\sum_{k=0}^{n-2}\alpha_{k+1}. \tag{34}$$

Using these facts in (32) we finally obtain

$$F_n \leqslant \frac{\|x_0 - x^\star\|^2 + 2\alpha_0\frac{\gamma_0^2}{1-\gamma_0^2}F_0}{2\sum_{k=1}^{n-1}\alpha_k}. \tag{35}$$

Note that the denominator of (35) grows at least as fast as k because we can show that the stepsize sequence $(\alpha_k)_{k\in\mathbb{N}}$ is separated from 0. Indeed, the existence of the Lyapunov function V^{a} for system (4)-(14) implies boundedness of the sequence $(x_k)_{k\in\mathbb{N}}$. For any $x^\star \in X^\star$, define $D^\star := \overline{\mathrm{conv}}(x^\star, x_0, x_1, ..)$, which is closed and convex. Therefore, by local smoothness there exists bounded $L_{D^\star}$ such that (2) holds on $D^\star$. It is then

$$\alpha_k \geqslant \frac{\gamma_k}{L_k} \geqslant \frac{\gamma_k}{L_{D^\star}} > 0, \quad \forall k \in \mathbb{N} \tag{36}$$

which shows the desired property.

Eq. (16c). We sum again (30) for $k = 0, 1, ..., n-1$ but this time also keep the last term v_k (17). We obtain

$$\sum_{k=1}^{n-1}\frac{\alpha_k}{L_{D_k}}\|\nabla f(x_k)\|^2 \leqslant c_3 \tag{37}$$

where the constant $c_3 \in (0, \infty)$ exists due to the bounded quantities involved in (30) and discussed in the previous item. Because $L_{D_k} \leqslant L_{D^\star} < \infty \forall k$, (37) gives summability of $\|\nabla f(x_k)\|^2$ and thus the result is proven. This shows that all cluster points of the sequences generated by (4)-(14) belong to $X^\star$. Using the Lyapunov condition (16a) and classic fixed point arguments, one can then conclude that any sequence generated by the algorithm converges to a solution.

REFERENCES

[1] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

[2] F. Lu and S. P. Meyn, "Convex q learning in a stochastic environment," in *62nd IEEE Conference on Decision and Control (CDC)*, 2023.

[3] C. W. Scherer, C. Ebenbauer, and T. Holicki, "Optimization algorithm synthesis based on integral quadratic constraints: A tutorial," in *62nd IEEE Conference on Decision and Control (CDC)*, 2023.

[4] F. Dörfler, Z. He, G. Belgioioso, S. Bolognani, J. Lygeros, and M. Muehlebach, "Toward a systems theory of algorithms," *IEEE Control Systems Letters*, vol. 8, pp. 1198–1210, 2024.

[5] J. M. Altschuler and P. A. Parrilo, "Acceleration by stepsize hedging: Silver stepsize schedule for smooth convex optimization," *Mathematical Programming*, 2024.

[6] Z. Zhang, J. Lee, S. Du, and Y. Chen, "Anytime acceleration of gradient descent," *arXiv preprint arXiv:2411.17668*, 2024.

[7] P. Latafat, A. Themelis, L. Stella, and P. Patrinos, "Adaptive proximal algorithms for convex optimization under local lipschitz continuity of the gradient," *Mathematical Programming*, 2024.

[8] P. Latafat, A. Themelis, and P. Patrinos, "On the convergence of adaptive first order methods: proximal gradient and alternating minimization algorithms," in *6th Annual Learning for Dynamics and Control Conference*, 2024.

[9] Y. Malitsky and K. Mishchenko, "Adaptive proximal gradient method for convex optimization," in *Advances in Neural Information Processing Systems*, 2024.

[10] M. Raginsky, A. Rakhlin, and S. Yüksel, "Online convex programming and regularization in adaptive control," in *49th IEEE Conference on Decision and Control (CDC)*, 2010.

[11] A. Iannelli, "Adaptive control mechanisms in gradient descent algorithms," in *arXiv preprint arXiv:2508.19100*, 2025.

[12] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 2014.

[13] M. Teboulle and Y. Vaisbourd, "An elementary approach to tight worst case complexity analysis of gradient based methods," *Mathematical Programming*, vol. 201, no. 1–2, p. 63–96, 2022.

[14] A. Mishkin, A. Khaled, Y. Wang, A. Defazio, and R. M. Gower, "Directional smoothness and gradient methods: Convergence and adaptivity," in *Advances in Neural Information Processing Systems*, 2024.

[15] S. Meng, A. Orvieto, D. Cao, and C. D. Sa, "Gradient descent on logistic regression with non-separable data and large step sizes," *arXiv preprint arXiv:2406.05033*, 2024.

[16] F. Jakob and A. Iannelli, "Online convex optimization and integral quadratic constraints: An automated approach to regret analysis," in *IEEE Conference on Decision and Control*, 2025.

[17] A. Beck, *First-Order Methods in Optimization*. SIAM-Society for Industrial and Applied Mathematics, 2017.