Online Optimisation Using Zeroth Order Oracles

Iman Shames, Daniel Selvaratnam, Jonathan H. Manton

***Abstract*—This paper considers the iterative numerical optimisation of time-varying cost functions where no gradient information is available at each iteration. In this case, the proposed algorithm estimates a directional derivative by finite differences. The main contributions are the derivation of error bounds for such algorithms and proposal of optimal algorithm parameter values, e.g. step-sizes, for strongly convex cost functions. The algorithm is applied to a tackle source localisation problem using a sensing agent where the source actively evades the agent. Numerical examples are provided to illustrate the theoretical results.**

***Index Terms*—Optimization algorithms, Optimization, Autonomous systems**

I. Introduction

An optimisation problem that changes at discrete instances in time can be treated as a sequence of optimisation problems and has applications in signal processing [1] and control [2]. Objectives often change in response to new information. In general, allowing for temporal variations in the cost function and constraints offers scope for real-time optimisation in the presence of uncertainty. Assuming that every cost function and feasible set in this sequence is made available to the solver, one approach would be to solve each individual problem completely. This may not be tractable, depending on the rate at which new functions arrive. It is also unnecessary if the primary objective is to generate a sequence of iterates that 'track' the time-varying optimal points, or to ensure that the cost remains within a desired neighbourhood of the optimal value. Assuming there are bounds on the variation between consecutive cost functions, information from the current problem can be exploited to optimise its successor. A more efficient alternative, therefore, is to only solve each problem partially by limiting the number of iterations per cost function. Any proposed method has to use the information about f_k that is made available to it by a *p-th order oracle*, denoted $\mathcal{O}_p$, for a fixed $p \in \{0, 1, 2, \dots\}$. If $p = 0$ then the zeroth-order oracle makes available only the value of the function at the current iterate x_k, i.e. $f_k(x_k)$. Similarly, the first ($p = 1$), second ($p = 2$), or higher-order ($p \geq 3$) oracle make up to the p-th derivative of the function at the current iterate x_k available to the method. Thus, methods that implement gradient descent on a sequence of cost functions require a first-order oracle. For example, [3] investigates the case of smooth, strongly convex functions, with bounds on tracking error derived for unconstrained problems. Error bounds are also presented for cost functions that vary in continuous time, with the gradient descent iterates replaced by a gradient-based control law. A general framework for time-varying convex optimisation problems (with time-varying constraints) is proposed in [4], based on the theory of averaged operators. The authors develop error bounds on a variant of the Mann-Krasnosel'skii iterations with time-varying operators. Methods using a second-order oracle for solving optimisation problems on manifolds with changing co-ordinate maps are treated in [5].

The authors are with the Department of Electrical and Electronic Engineering at the University of Melbourne, Australia. e-mail: {ishames, jmanton}@unimelb.edu.au, daniel.selvaratnam@alumni.unimelb.edu.au

In this paper, in place of a first-order or a second-order oracle, a zeroth-order oracle is assumed, which only provides the optimisation algorithm with the value of the cost function at the current time. The cost function is thereby treated as a black box, with a time-varying input-output map. This corresponds to practical scenarios in which derivatives are either unavailable [6], [7], or the cost of computing them is prohibitive. In such cases, the cost function directional derivatives can still be approximated via finite-differences. This method is adopted for time-invariant problems in [8], which relies on a two-point estimate of the directional derivative in a randomly chosen direction. We have built on the techniques from [8] to deal with time-varying costs.

Literature Review: We start by briefly reviewing time-varying optimisation problems and then move on to focus on gradient-free (using zeroth order oracles) solutions to time-varying optimisation problems. We do not review time-invariant problems as it has less relevance to this paper, but refer the reader to [9] for a comprehensive review of time-invariant gradient-free optimisation. Time-varying optimisation problems have mainly been considered by the machine learning community under the term Online Convex Optimisation (OCO) [10], [11]. OCO frames the problem as a game, in which the iterates are actions selected by a player. In response, an adversary selects a cost function, which determines the cost incurred. In contrast to the standard optimisation literature, OCO is concerned with minimising *regret*, which captures the integrated cost rather than the instantaneous cost. The use of a zeroth order oracle in OCO is often termed *bandit feedback*, because it constitutes a multi-armed bandit problem [12] with convex losses. Regret bounds for convex problems under fixed compact constraints are derived in [13], [14] for iterative methods that rely on multi-point estimates of the gradient. A two-point version of this is the focus of [15], and belongs to the class of iterations considered in [8]. Methods relying on a single-point estimate of the directional derivative are also developed in [7], [16] and [12, Section 6.2]. The aforementioned works obtain bounds on *static regret*, which compares the observed cost to the cost at a fixed decision variable. *Dynamic regret* extends this concept by comparing the observed cost to the time-varying optimal cost [10]. Dynamic regret bounds on primal-dual saddle point iterations in a bandit feedback setting are derived in [6], under time-varying compact constraints. The recent work [17] examines the dynamic regret of a proximal online gradient descent algorithm, for minimising a time-varying convex but non-differentiable cost. There, the proximal operator is used in the absence of a gradient. Of greatest relevance to the work herein are [18], [19], which derive dynamic regret bounds for two-point bandit feedback methods. In [18] the directional derivative is estimated by sampling a direction from the unit sphere, while in [19] the direction is sampled from the co-ordinate axes.

Contributions: The works mentioned thus far have all assumed the cost function does not change between evaluations when forming the two-point estimate, i.e. function evaluations are carried out simultaneously. In contrast, our work permits the cost-function to change between the two evaluations, better respecting the time-varying nature of the problem. We also offer a different perspective on the analysis by deriving bounds on the expected instantaneous tracking error, rather than regret. This is better suited to some problems prevalent in control and signal processing such as parameter estimation or tracking. An example of such problems is studied in Section III.

II. Zeroth-order oracle

We will work with a family of continuous functions

$$\mathcal{F} := \{f_k : \mathbb{R}^n \to \mathbb{R} \mid k \in \mathbb{K}\},$$

where $\mathbb{K} = \mathbb{N} \cup \{j + \frac{1}{2} | j \in \mathbb{N}\} = \{0, 1/2, 1, 3/2, 2, 5/2, \dots\}$. For the ease of notation, we denote $k + \frac{1}{2}$ by k^+ for all $k \in \mathbb{N}$. We are interested in generating solutions to the following sequence of optimisation problems

$$\min_{x \in \mathbb{R}^n} f_k(x), \tag{1}$$

using iterates of the form

$$x_{k+1} = x_k - \alpha_k g_{\eta,k^+}(x_k, U_k), \tag{2}$$

where α_k is a positive scalar known as the *step-size*, $U_k \sim \mathcal{N}(0, I_n)$, and

$$g_{\eta,k^+}(x, u) := \frac{f_{k^+}(x + \eta u) - f_k(x)}{\eta} u. \tag{3}$$

In this update, (3) provides an approximation for the directional derivative of the function and its properties will be investigated in the following.

We do not assume that the cost function is the same at both instances needed for computing (3). Particularly, index k^+ emphasises the fact the function might change between being evaluated at x_k and $x_k + \eta U_k$. Conceptually, this corresponds to a time index "between" k and $k + 1$ at which the value function is calculated at $x_k + \eta U_k$. Throughout this paper, $\|\cdot\|$ is used to denote the Euclidean norm of its argument. Our analysis will rely on the following assumptions.

Assumption 1 (Lipschitz Gradient)**.** *Every $f \in \mathcal{F}$ is twice continuously differentiable, and $\exists L > 0$, $\forall f \in \mathcal{F}$, $\forall x, y \in \mathbb{R}^n$, $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$.*

Assumption 2 (Uniform strong convexity)**.** *There exists $\sigma \in (0, L]$ such that $\forall f \in \mathcal{F}$, $\forall x \in \mathbb{R}^n$, $\nabla^2 f(x) \succeq \sigma I$.*

Under Assumption 2 (strong convexity), each $f \in \mathcal{F}$ satisfies the restricted secant inequality [20, Appendix A]

$$\forall x \in \mathbb{R}^n,\ \nabla f(x)^\top (x - x^\star) \geq \frac{\sigma}{2}\|x - x^\star\|^2, \tag{4}$$

which will be exploited in the forthcoming analysis.

Assumption 3 (Bounded change in minimiser)**.** *Given $\mathcal{F}$, there exists $V \geq 0$ such that $\|x^\star_{k+1} - x^\star_{k^+}\| \leq V$, where*

$$x^\star_k := \arg\min\{f_k(x) \mid x \in \mathbb{R}^n\}, \quad \forall k \in \mathbb{K}. \tag{5}$$

Remark II.1. *As demonstrated in [21, Lemma 4.2], if f_{k^+} and f_{k+1} are twice continuously differentiable, and strongly convex with modulus σ, then Assumption 3 holds if there exists a positive scalar δ_1 such that $\|\nabla f_{k^+}(x) - \nabla f_{k+1}(x)\| \leq \delta_1$ with $V = \frac{\delta_1}{\sigma}$.*

The following is assumed regarding functions f_k and f_{k^+} .

Assumption 4. *There exists a positive scalar δ such that $|f_k(x) - f_{k^+}(x)| \leq \delta$, for all $x \in \mathbb{R}^n$.*

Remark II.2. *In certain scenarios one might have access to the same function value at the two points needed to compute* (3)*. For example, in the scenario where the goal is to locate the source of signal corresponding to the minimum of a time varying signal density function and one has access to two sensors that can measure the strength of the signal at different locations at the same time. Such cases are a special case of the problems we consider here where $\delta = 0$ and $g_{\eta,k^+}(x, u) = g_{\eta,k}(x, u)$.*

Let the *smoothed cost* be defined as $f_{\eta,k^+}(x) := \mathbb{E}[f_{k^+}(x + \eta u)]$ where $u \sim \mathcal{N}(0, I_n)$. One obtains

$$\begin{aligned}\nabla f_{\eta,k^+}(x) &= \mathbb{E}\left[\frac{f_{k^+}(x + \eta u)}{\eta} u\right] \\ &= \mathbb{E}\left[\frac{f_{k^+}(x + \eta u) - f_k(x)}{\eta} u\right] = \mathbb{E}\left[g_{\eta,k^+}(x, u)\right].\end{aligned} \tag{6}$$

Define the *tracking error* as

$$e_k := \|x_k - x^\star_{k^+}\| \tag{7}$$

and the *estimation error* as

$$\bar{e}_k := \|x_{k+1} - x^\star_{k^+}\|, \tag{8}$$

which yields:

$$\begin{aligned}\bar{e}_k^2 &= \|x_{k+1} - x^\star_{k^+}\|^2 = \|(x_k - \alpha_k g_{k^+}) - x^\star_{k^+}\|^2 \\ &= \left[(x_k - x^\star_{k^+}) - \alpha_k g_{k^+}\right]^\top \left[(x_k - x^\star_k) - \alpha_k g_{k^+}\right] \\ &= e_k^2 - 2\alpha_k g_{k^+}^\top (x_k - x^\star_{k^+}) + \alpha_k^2 \|g_{k^+}\|^2.\end{aligned} \tag{9}$$

where $g_{k^+} := g_{\eta,k^+}(x_k, U_k)$. Assumption 3 implies

$$\begin{aligned}e_{k+1} &:= \|x_{k+1} - x^\star_{k+1}\| \\ &\leq \|x_{k+1} - x^\star_{k^+}\| + \|x^\star_{k^+} - x^\star_{k+1}\| \\ &\leq \bar{e}_k + V,\end{aligned} \tag{10}$$

$$\mathbb{E}[e_{k+1} \mid x_k] \leq \mathbb{E}[\bar{e}_k \mid x_k] + V. \tag{11}$$

Using the fact that $\nabla f_{k^+}(x) = \mathbb{E}\left[\left(\nabla f_{k^+}(x)^T u\right) u\right]$ a bounding inequality for the difference in gradients between the original and the smoothed cost can computed as (Its derivation is given in (24).):

$$\|\nabla f_{\eta,k^+}(x) - \nabla f_{k^+}(x)\| \leq \frac{\eta L}{2}\left[(n+3)^{\frac{3}{2}} + \delta n^{1/2}\right]. \quad (12)$$

One can also obtain a bound on $\mathbb{E}[\|g_{\eta,k^+}(x,u)\|^2]$. Note

$$\begin{aligned}&\left(f_{k^+}(x+\eta u) - f_k(x)\right)^2\\&= \left(f_{k^+}(x+\eta u) - f_k(x) - \eta\nabla f_{k^+}(x)^\top u + \eta \nabla f_{k^+}(x)^\top u\right)^2\\&\leq 2\left(\frac{\eta^2}{2}L\|u\|^2 + \delta\right)^2 + 2\eta^2\left(\nabla f_{k^+}(x)^\top u\right)^2.\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbb{E}[\|g_{\eta,k^+}(x,u)\|^2] \leq \frac{\eta^2}{2}L^2\mathbb{E}[\|u\|^6] + 2L\delta\mathbb{E}[\|u\|^4]\\ + 2\frac{\delta}{\eta^2}\mathbb{E}[\|u\|^2]\\ + 2\mathbb{E}\left[\left\|\left(\nabla f_{k^+}(x)^\top u\right)u\right\|^2\right].\end{aligned}$$

Applying Propositions 6 and 7 (in the appendix):

$$\begin{aligned}\mathbb{E}[\|g_{\eta,k^+}(x,u)\|^2] \leq \frac{\eta^2}{2}L^2(6+n)^3 + 2L\delta(n+4)^2 + 2\frac{\delta^2}{\eta^2}n\\ + 2(n+4)\|\nabla f_{k^+}(x)\|^2. \quad (13)\end{aligned}$$

The main result is presented below.

Theorem 1. *Let $\mathcal{F}$ satisfy Assumptions 1 – 4, and consider the iterations $x_{k+1} = x_k - \alpha_k g_{\eta,k^+}(x_k, U_k)$, where $U_k \sim \mathcal{N}(0, I_n)$ and $\alpha_k > 0$. Then for all $k \geq 0$,*

$$\mathbb{E}[e_{k+1} \mid x_k] \leq e_k\sqrt{2L^2(n+4)\alpha_k^2 - \sigma\alpha_k + 1} + D_k + V, \quad (14)$$

where

$$D_k := \alpha_k \max\left\{\frac{\eta L\left((n+3)^{3/2} + \delta n^{1/2}\right)}{2\sqrt{2L^2(n+4)\alpha_k^2 - \sigma\alpha_k + 1}},\right. \left.\sqrt{\frac{(\eta L)^2}{2}(n+6)^3 + 2L\delta(n+4)^2 + \frac{2\delta^2 n}{\eta^2}}\right\}. \quad (15)$$

Proof. From (9):

$$\bar{e}_k^2 \leq e_k^2 - 2\alpha_k g_{\eta,k^+}(x_k,U_k)^\top(x_k - x_k^\star) + \alpha_k^2\|g_{\eta,k^+}(x_k,U_k)\|^2.$$

Taking conditional expectations,

$$\begin{aligned}\mathbb{E}[\bar{e}_k^2 \mid x_k] \leq e_k^2 - 2\alpha_k\mathbb{E}[g_{\eta,k^+}(x_k,U_k) \mid x_k]^\top(x_k - x_k^\star)\\ + \alpha_k^2\mathbb{E}\left[\|g_{\eta,k^+}(x_k,U_k)\|^2 \mid x_k\right],\end{aligned}$$

and then applying (6) and (13),

$$\begin{aligned}\mathbb{E}[\bar{e}_k^2 \mid x_k] \leq e_k^2 - 2\alpha_k\nabla f_{\eta,k^+}(x_k)^\top(x_k - x_{k^+}^\star)\\ + 2\alpha_k^2(n+4)\|\nabla f_k(x_k)\|^2 + \frac{(\eta\alpha_k L)^2}{2}(n+6)^3\\ + 2L\delta\alpha_k^2(n+4)^2 + 2\alpha_k^2\frac{\delta^2}{\eta^2}n. \quad (16)\end{aligned}$$

Equations (4) and (12) then imply

$$\begin{aligned}&\nabla f_{\eta,k^+}(x_k)^\top(x_k - x_{k^+}^\star)\\&\quad= [\nabla f_{\eta,k^+}(x_k) - \nabla f_k(x_k)]^\top(x_k - x_{k^+}^\star)\\&\qquad + \nabla f_k(x_k)^\top(x_k - x_{k^+}^\star)\\&\quad\geq \nabla f_k(x_k)^\top(x_k - x_{k^+}^\star)\\&\qquad - \|\nabla f_{\eta,k^+}(x_k) - \nabla f_k(x_k)\|e_k\\&\quad\geq \frac{\sigma}{2}e_k^2 - \frac{\eta L}{2}\left((n+3)^{3/2} + \delta n^{1/2}\right)e_k. \quad (17)\end{aligned}$$

Recall that gradient magnitude is bounded by tracking error in unconstrained problems (see Lemma 4):

$$\|\nabla f_{k^+}(x_k)\| \leq L e_k. \quad (18)$$

Applying (18) and (17) to (16) yields

$$\begin{aligned}\mathbb{E}[\bar{e}_k^2 \mid x_k] \leq e_k^2 - \alpha_k\sigma e_k^2 + \alpha_k\eta L\left((n+3)^{3/2} + \delta n^{1/2}\right)e_k\\ + 2\alpha_k^2(n+4)\|\nabla f_k(x_k)\|^2\\ + \alpha_k^2\left(\frac{(\eta L)^2}{2}(n+6)^3 + 2L\delta(n+4)^2 + \frac{2\delta^2 n}{\eta^2}\right)\end{aligned}$$

$$\begin{aligned}\mathbb{E}[\bar{e}_k^2 \mid x_k] \leq \overbrace{[2L^2(n+4)\alpha_k^2 - \sigma\alpha_k + 1]}^{:=a_k}e_k^2\\ + \overbrace{\left[\eta L\alpha_k\left((n+3)^{3/2} + \delta n^{1/2}\right)\right]}^{:=b_k}e_k\\ + \overbrace{\left[\frac{(\eta L)^2}{2}(n+6)^3 + 2L\delta(n+4)^2 + \frac{2\delta^2 n}{\eta^2}\right]\alpha_k^2}^{:=c_k}.\end{aligned} \quad (19)$$

Applying Jensen's inequality and Lemma 5 (in the appendix) to (19) results in

$$\begin{aligned}\mathbb{E}[\bar{e}_k \mid x_k] \leq \sqrt{\mathbb{E}[\bar{e}_k^2 \mid x_k]} \leq e_k\sqrt{a_k} + D_k\\ \leq e_k\sqrt{2L^2(n+4)\alpha_k^2 - \sigma\alpha_k + 1} + D_k,\end{aligned}$$

where $D_k := \max\left\{\frac{b_k}{2\sqrt{a_k}}, \sqrt{c_k}\right\}$ is given by (15). Finally applying (11), completes the proof. □

Remark II.3. *Inequality* (14) *implies*

$$\mathbb{E}[e_{k+1}] \leq \mathbb{E}[e_k]\sqrt{2L^2(n+4)\alpha_k^2 - \sigma\alpha_k + 1} + D_k + V. \quad (20)$$

For the rest of this section, we limit the choice of the step-size to constant step-sizes, i.e., $\alpha_k = \alpha \in (0, \frac{\sigma}{2L^2(n+4)})$, $\forall k$.

Corollary 2. *If $\alpha \in (0, \frac{\sigma}{2L^2(n+4)})$ for all k, then*

$$\limsup_{k\to\infty}\mathbb{E}[e_k] \leq \Delta, \quad (21)$$

where

$$\Delta = \frac{D+V}{1-\rho}, \quad (22)$$

$\rho := \sqrt{2L^2(n+4)\alpha^2 - \sigma\alpha + 1}$, *and*

$$D := \alpha \max \left\{ \overbrace{\frac{\eta L\left((n+3)^{3/2} + \delta n^{1/2}\right)}{2\rho}}^{:=\theta_1}, \overbrace{\sqrt{\frac{(\eta L)^2}{2}(n+6)^3 + 2L\delta(n+4)^2 + \frac{2\delta^2 n}{\eta^2}}}^{:=\theta_2} \right\}.$$

Proof. If $0 < \alpha < \frac{\sigma}{2L^2(n+4)}$, then $2L^2(n+4)\alpha^2 - \sigma\alpha + 1 < 1$. Consequently, (21) follows from (20). □

In the following, we shift our focus to study the impact of different values of η and α on the performance of the algorithm. We choose (22), the tracking error bound as $k \to \infty$, as a proxy for the performance of the proposed algorithm. One might expect that setting η to be arbitrarily small improves the precision of the finite difference (3) and ultimately results in a smaller (22). After a closer inspection this is revealed to be false. The value of D tends to infinity as η tends to zero. Hence, one needs to select both α and η to minimise Δ as defined in (22). While the optimisation problem corresponding to these choices is small, in general, providing a closed-form solution for it is not possible. However, under an extra assumption on δ, such a solution can be computed:

Theorem 3. *If* $\delta \leq \dfrac{(n+6)^{3/2}}{n^{1/2}}$, *then* $\bar{\eta}$ *and* $\bar{\alpha}$ *minimise* Δ *where* $\bar{\eta} = \left(\dfrac{4\delta^2 n}{L^2(n+6)^3}\right)^{1/4}$ *and* $\bar{\alpha}$ *is the root of* $A\alpha^2 + B\alpha + C = 0$,*in the interval* $(0, \frac{\sigma}{2L^2(n+4)})$ *with* $A := 8(\bar{\theta}\sigma + 4VL^2(n+4))^2\bar{\theta}^2 L^2(n+4)$, $B := -2V(\bar{\theta}\sigma^2 + 4VL^2(n+4)\sigma + 8\bar{\theta}L^2(n+4))$, $C := (V\sigma + 2\bar{\theta})^2 - 4\bar{\theta}^2$, *and* $\bar{\theta} = \sqrt{\frac{(\bar{\eta}L)^2}{2}(n+6)^3 + 2L\delta(n+4)^2 + \frac{2\delta^2 n}{\bar{\eta}^2}}$.

Proof. Note that $\sigma \leq L < 2L\sqrt{(n+4)}$, and

$$\rho^2 = 2L^2(n+4)\alpha^2 - \sigma\alpha + 1 \geq \frac{1}{2} \tag{23}$$

for all α. Now (23) implies $2\rho \geq \frac{2}{\sqrt{2}} = \sqrt{2}$. Thus,

$$\theta_1 \leq \frac{\eta L\left((n+3)^{3/2} + \delta n^{1/2}\right)}{\sqrt{2}}.$$

By inspection, it can be seen that if $\eta \leq \sqrt{\frac{2}{L}}$, then

$$\frac{\eta L\left((n+3)^{3/2} + \delta n^{1/2}\right)}{\sqrt{2}} \leq \theta_2.$$

Hence, $D = \alpha\theta_2$ and $\Delta = \frac{\theta_2\alpha + V}{1-\rho}$ Moreover, $\bar{\eta} = \left(\dfrac{4\delta^2 n}{L^2(n+6)^3}\right)^{1/4}$ minimises θ_2 and $\left(\dfrac{4\delta^2 n}{L^2(n+6)^3}\right)^{1/4} \leq \sqrt{\frac{2}{L}}$ if $\delta \leq \dfrac{(n+6)^{3/2}}{n^{1/2}}$. As the denominator of Δ is independent of η, Δ attains its minimum at $\bar{\eta}$. The optimal step-size $\bar{\alpha}$ is computed by taking the derivative of Δ evaluated at $\bar{\eta}$ with respect to α and setting it equal to zero. □

Remark II.4 (Convergence and Computational Complexity)**.** *The expected value of the tracking error enters a ball of radius* $\Delta + \epsilon$ *centred at the origin in finitely-many steps where* ϵ *is an arbitrary positive scalar. The expected value of the tracking error remains inside this ball thereafter. In other words,* $\exists K \in \mathbb{N}$ *such that for all* $k \geq K$, $\mathbb{E}[e_k] \leq \Delta + \epsilon$. *Assuming that* $(1-\rho)\mathbb{E}[e_0] - V > 0$, *it can be observed that*[1]

$$K \leq \log\left[\frac{D + (1-\rho)\epsilon}{(1-\rho)\mathbb{E}[e_0] - V}\right](\log\rho)^{-1}.$$

All the terms on the RHS of the above equation are independent of n *except for* D *which is of order* $O(n)$ *if* δ *satisfies the hypothesis of Theorem 3 and* $\eta = \bar{\eta}$. *Each step of the proposed method requires two function evaluations, an* n*-dimensional Gaussian random variable generation, and* $O(n)$ *operations required to compute two scalar multiplication of two* n*-dimensional vectors and two* n*-dimensional vector summations. Assuming that all the necessary random vectors are sampled prior to running the algorithm (to amortise the computation cost) the complexity of entering the aforementioned ball will be of order* $O(n \log n)$.

III. Localisation of an Evading Source Using a Mobile Agent and Other Numerical Examples

This section considers the problem of steering a mobile agent to the vicinity of a drifting source is considered. The motions of the agent and the source are assumed to be modelled by single integrators. Particularly, let $x_k \in \mathbb{R}^n$ and $z_k \in \mathbb{R}^n$ be the position of the agent and the source at time index k, respectively. Their motions are governed by $x_{k+1} = x_k + \xi_k$, $z_{k+1} = z_k + \zeta_k$, where ξ_k and ζ_k are the velocities of the agent and the evading source and $\|\zeta_k\| \leq V$. While in principle no further assumption on ζ_k is required, to make the scenario interesting we assume that the source knows x_k exactly and always choose to maximise its distance from the agent by setting $\zeta_k = V(z_k - x_k)/\|z_k - x_k\|$. In other words, the source actively evades the agent with maximum speed. An example for this scenario is the case where the agent can measure $f_k(x_k) = \dfrac{1}{2}\|x_k - z_k\|^2$ and $f_{k^+}(x_k + \eta U_k) = \dfrac{1}{2}\|x_k + \eta U_k - (z_k + 0.5\zeta_k)\|^2$. These functions satisfy Assumptions 1 – 4 with $L = \sigma = 1$, $V = 0.1$, $\delta = 1$, and $\xi_k = \alpha_k g_{\eta,k^+}(x_k, U_k)$, where $v_\xi = 1$, and $\alpha_k = 0.03$ with the value 0.03 being obtained from Theorem 3. We have compared the performance of the proposed algorithm for three different values for η with a modified algorithm in which U_k is uniformly drawn from the surface of a sphere as in [18]. The average tracking errors after 1000 runs for each scenario where the starting position of the agents were selected uniformly in a square of 100×100 are presented in Fig. 1. In the simulation the choice of $\eta = 0.35$ is equal to $\bar{\eta}$ given in Theorem 3. The negative impact of choosing small η can be observed from the figure. The spherical sampling strategies seems to result in a larger

[1]If $(1-\rho)\mathbb{E}[e_0] - V < 0$, then $K = 1$.

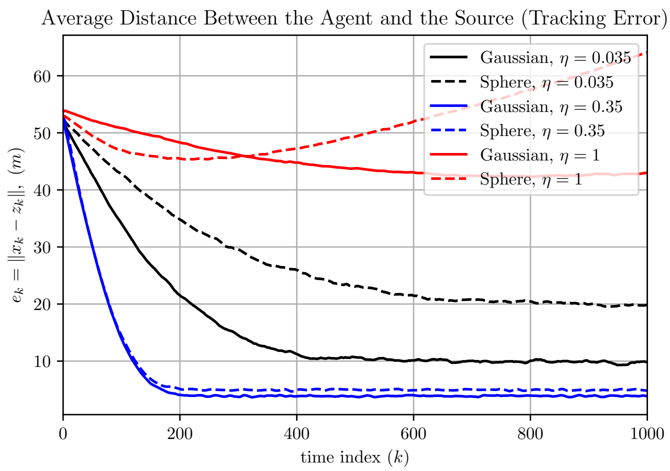

Fig. 1: The average tracking error for different choices of η and methodologies for selecting U_k (Gaussian versus uniformly from the surface of a sphere) after running 1000 random scenarios.

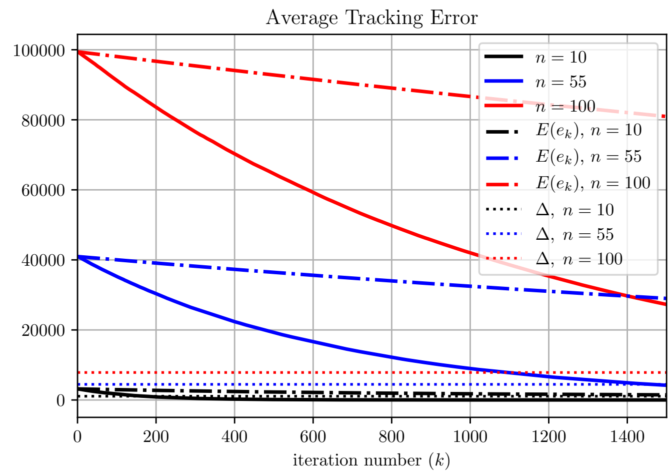

Fig. 2: Algorithm tracking error for different values of n. The dashed-dot bounds asymptotically converge to the dotted values. The algorithms and the bounds are tested for $\mathbb{E}[e_0] = 100n$.

tracking error and asymptotic bound compared to the Gaussian sampling strategy. For larger choices of η, the algorithm based on sampling from a the surface of a sphere seems to fail to converge to a ball around the source.

In the next numerical example, we use a sequence of cost functions of the form $f_k(x) = \frac{1}{2}(x - z_k)^\top Q_k(x - z_k)$, where $Q_k \in \mathbb{R}^{n\times n}$ and $z_k \in \mathbb{R}^n$ are randomly generated at each iteration k and satisfy $\sigma I \preceq Q_k = Q_k^\top \preceq LI$, and $\|z_{k+1} - z_k\| \leq V$, for all k with parameter values $\sigma = 1$, $L = 2$, $V = 1$, $\delta = (n+6)^{3/2}/n^{1/2}$, $\eta = \bar{\eta}$, and $\alpha = \bar{\alpha}$ as described in Theorem 3. The average performance of the proposed optimisation method after 100 random runs, the expected tracking error bound at each iteration, $\mathbb{E}[e_k]$ given by (20), and the asymptotic value of the bound, Δ, given by (22), for different problem sizes (different values of n) are depicted in Fig. 2.

IV. Conclusion

A gradient-free optimisation algorithm for solving time-varying problems along with bounds in the expectation of the tracking error was proposed. Tracking error bounds for unconstrained strongly convex problems are first derived for a randomised algorithm that relies on a zeroth-order oracle. The results are then applied to the problem of localisation of a drifting source. Extensive numerical studies to illustrate the results are presented.

The results in this paper can be extended to constrained and non-smooth time-varying optimisation problems. So far, the bounds for gradient-free methods hold only in expectation. Another future direction is to investigate the possibility of obtaining bounds that hold almost surely. The work of Duchi *et al.* [22] on gradient-free stochastic time-invariant optimisation also offers a useful analysis that can be exploited for time-varying problems. Similar bounds for other gradient-free optimisation methods, e.g. those that rely on choosing U_k uniformly from the surface of a sphere, can be derived. To this aim, computing bounds analogous to (12) and (13) is necessary. The detailed derivation of such bounds and conducting an analytical comparison of the performance of different optimisation algorithms using different sampling strategies is left as a future direction. In the context of drifting source localisation, the problem of incorporating the dynamical constraints of the agent in the proposed motion strategy is an obvious future direction.

References

[1] F. Y. Jakubiec and A. Ribeiro, "D-MAP: Distributed Maximum a Posteriori Probability Estimation of Dynamic Systems," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 450–466, Jan. 2013.

[2] B. Gutjahr, L. Gröll, and M. Werling, "Lateral Vehicle Trajectory Optimization Using Constrained Linear Time-Varying MPC," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1586–1595, Jun. 2017.

[3] A. Y. Popkov, "Gradient Methods for Nonstationary Unconstrained Optimization Problems," *Automation and Remote Control*, vol. 66, no. 6, pp. 883–891, Jun. 2005.

[4] A. Simonetto, "Time-Varying Convex Optimization via Time-Varying Averaged Operators," *arXiv:1704.07338 [math]*, Apr. 2017.

[5] J. H. Manton, "A framework for generalising the Newton method and other iterative methods from Euclidean space to manifolds," *Numerische Mathematik*, vol. 129, no. 1, pp. 91–125, Jan. 2015.

[6] T. Chen and G. B. Giannakis, "Bandit Convex Optimization for Scalable and Dynamic IoT Management," *arXiv:1707.09060 [cs]*, Jul. 2017.

[7] S. J. Kim and G. B. Giannakis, "An Online Convex Optimization Approach to Real-Time Energy Pricing for Demand Response," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2784–2793, Nov. 2017.

[8] Y. Nesterov and V. Spokoiny, "Random Gradient-Free Minimization of Convex Functions," *Foundations of Computational Mathematics*, vol. 17, no. 2, pp. 527–566, Apr. 2017.

$$
\begin{aligned}
\|\nabla f_{\eta,k^+}(x)-\nabla f_{k^+}(x)\| &= \left\|\mathbb{E}\left[\left(\frac{f_{k^+}(x+\eta u)-f_k(x)}{\eta}-\nabla f_{k^+}(x)^T u\right)u\right]\right\| \\
&= \left\|\frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}^n}\left(\frac{f_{k^+}(x+\eta u)-f_k(x)}{\eta}-\nabla f_{k^+}(x)^T u\right)ue^{-\frac{1}{2}\|u\|^2}du\right\| \\
&\le \frac{1}{\eta\sqrt{2\pi}}\int_{\mathbb{R}^n}|f_{k^+}(x+\eta u)-f_k(x)-\eta\nabla f_{k^+}(x)^T u|\|u\|e^{-\frac{1}{2}\|u\|^2}du \\
&\le \frac{1}{\eta\sqrt{2\pi}}\int_{\mathbb{R}^n}\left(|f_{k^+}(x+\eta u)-f_{k^+}(x)-\eta\nabla f_{k^+}(x)^T u|+\delta\right)\|u\|e^{-\frac{1}{2}\|u\|^2}du \qquad \text{(Assumption 4)} \\
&\le \frac{L\eta}{2\sqrt{2\pi}}\int_{\mathbb{R}^n}\left(\|u\|^3+\delta\|u\|\right)e^{-\frac{1}{2}\|u\|^2}du \qquad \text{(Assumption 1)} \\
&\le \frac{\eta}{2}L\left((n+3)^{\frac{3}{2}}+\delta n^{1/2}\right) \qquad \text{(Proposition 6)}
\end{aligned} \tag{24}
$$

[9] A. Conn, K. Scheinberg, and L. Vicente, *Introduction to Derivative-Free Optimization*, ser. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, Jan. 2009.

[10] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 2003, pp. 928–936.

[11] E. Hazan, "Introduction to online convex optimization," *Foundations and Trends® in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2015, oCLC: 995130886.

[12] S. Bubeck, "Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems," *Foundations and Trends® in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.

[13] A. Agarwal, O. Dekel, and L. Xiao, "Optimal Algorithms for Online Convex Optimization with Multi-Point Bandit Feedback." in *COLT*. Citeseer, 2010, pp. 28–40.

[14] O. Shamir, "An optimal algorithm for bandit and zero-order convex optimization with two-point feedback," *Journal of Machine Learning Research*, vol. 18, no. 52, pp. 1–11, 2017.

[15] C.-K. Chiang, C.-J. Lee, and C.-J. Lu, "Beating Bandits in Gradually Evolving Worlds," in *Proceedings of the 26th Annual Conference on Learning Theory*, 2013, pp. 210–227.

[16] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, "Online convex optimization in the bandit setting: Gradient descent without a gradient," in *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2005, pp. 385–394.

[17] R. Dixit, A. S. Bedi, R. Tripathi, and K. Rajawat, "Online Learning With Inexact Proximal Online Gradient Descent Algorithms," *IEEE Transactions on Signal Processing*, vol. 67, no. 5, pp. 1338–1352, Mar. 2019.

[18] T. Yang, L. Zhang, R. Jin, and J. Yi, "Tracking Slowly Moving Clairvoyant: Optimal Dynamic Regret of Online Learning with True and Noisy Gradient," in *International Conference on Machine Learning (ICML)*, 2016, p. 12.

[19] O. Besbes, Y. Gur, and A. Zeevi, "Non-Stationary Stochastic Optimization," *Operations Research*, vol. 63, no. 5, pp. 1227–1244, Sep. 2015.

[20] H. Karimi, J. Nutini, and M. Schmidt, "Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-{\L}ojasiewicz Condition," *arXiv:1608.04636 [cs, math, stat]*, Aug. 2016.

[21] D. D. Selvaratnam, I. Shames, J. H. Manton, and M. Zamani, "Numerical optimisation of time-varying strongly convex functions subject to time-varying constraints," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 849–854.

[22] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono, "Optimal Rates for Zero-Order Convex Optimization: The Power of Two Function Evaluations," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2788–2806, May 2015.

APPENDIX

Lemma 4. *Let $X \subset \mathbb{R}^n$ be closed and convex. If $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and satisfies*

$$\exists L > 0,\ \forall x, y \in X,\ \|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|,$$

then for any $x^\star \in \arg\min\{f(x) \mid x \in X\}$ and $x \in X$,

$$|\|\nabla f(x^\star)\| - \|\nabla f(x)\|| \le L\|x - x^\star\|$$

Proof. The result follows directly from the reverse triangle inequality. ☐

Lemma 5. *Let $x, y, a, c \ge 0$, and $b \in \mathbb{R}$. Then*

$$x^2 \le ay^2 + by + c \implies x \le \sqrt{a}y + D,$$

where $D := \max\left\{\frac{b}{2\sqrt{a}}, \sqrt{c}\right\}$.

Proof. The definition of D implies both $2\sqrt{a}D \ge b$ and $D^2 \ge c$. Assuming the LHS of the implication holds,

$$
\begin{aligned}
x^2 &\le ay^2 + by + c \\
&\le ay^2 + 2\sqrt{a}Dy + D^2 \\
&= (\sqrt{a}y + D)^2.
\end{aligned}
$$

☐

Proposition 6 (Lemma 1, [8])**.** *Let $M_p := \frac{1}{\sqrt{\pi}}\int_{\mathbb{R}^n}\|u\|^p e^{-\frac{1}{2}\|u\|^2}du$. For $p \in [0, 2]$, $M_p \le n^{p/2}$. If $p \ge 2$, then $n^{p/2} \le M_p \le (p+n)^{p/2}$.*

Proposition 7 (Thereom 3, [8])**.** *If f is differentiable at x, then $\mathbb{E}\left[\left\|\left(\nabla f(x)^T u\right)u\right\|\right] \le (n+4)\|\nabla f(x)\|$, where $u \sim \mathcal{N}(0, I_n)$.*