SIAM J. OPTIM.
Vol. 30, No. 4, pp. 2726–2749

STOCHASTIC THREE POINTS METHOD FOR UNCONSTRAINED SMOOTH MINIMIZATION*

EL HOUCINE BERGOU†, EDUARD GORBUNOV‡, AND PETER RICHTÁRIK§

Abstract. In this paper we consider the unconstrained minimization problem of a smooth function in $\mathbb{R}^n$ in a setting where only function evaluations are possible. We design a novel randomized derivative-free algorithm—the *stochastic three points (STP)* method—and analyze its iteration complexity. At each iteration, STP generates a random search direction according to a certain fixed probability law. Our assumptions on this law are very mild: roughly speaking, all laws which do not concentrate all measures on any halfspace passing through the origin will work. For instance, we allow for the uniform distribution on the sphere and also distributions that concentrate all measures on a positive spanning set. Although our approach is designed to not explicitly use derivatives, it covers some first order methods. For instance, if the probability law is chosen to be the Dirac distribution concentrated on the sign of the gradient, then STP recovers the signed gradient descent method. If the probability law is the uniform distribution on the coordinates of the gradient, then STP recovers the randomized coordinate descent method. The complexity of STP depends on the probability law via a simple characteristic closely related to the cosine measure which is used in the analysis of deterministic direct search (DDS) methods. Unlike in DDS, where $O(n)$ (n is the dimension of x) function evaluations must be performed in each iteration in the worst case, our method only requires two new function evaluations per iteration. Consequently, while the complexity of DDS depends quadratically on n, our method depends linearly on n.

Key words. unconstrained smooth minimization, derivative-free optimization, deterministic direct search

AMS subject classifications. 90C56, 90C60, 68W20

DOI. 10.1137/19M1244378

1. Introduction.

In this paper we consider the problem

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a given smooth objective function. We assume that we do not have access to the derivatives of f and only have access to a function evaluation oracle. In other words, we assume that we work in the derivative-free optimization (DFO) setting [3]. Optimization problems of this type appear in many industrial applications where usually the objective function is evaluated through a computer simulation process, and therefore derivatives cannot be directly evaluated, e.g., shape optimization in fluid-dynamics problems [1, 10, 16].

Direct search methods of directional type [13, 3] are a popular class of methods for DFO and are among the first algorithms proposed in numerical optimization [15].

*Received by the editors February 12, 2019; accepted for publication (in revised form) August 3, 2020; published electronically October 1, 2020.
https://doi.org/10.1137/19M1244378
Funding: The first author received support from the AgreenSkills+ fellowship programme, which has received funding from the EU's Seventh Framework Programme under grant agreement FP7-609398 (AgreenSkills+ contract).

†King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, and MaIAGE, INRAE, Université Paris-Saclay, 78350 Jouy-en-Josas, France (elhoucine.bergou@inra.fr).
‡Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation (eduard.gorbunov@phystech.edu).
§King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, University of Edinburgh, Edinburgh, UK, and Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation (peter.richtarik@kaust.edu.sa).

These methods are characterized by evaluating the objective function over a number of (typically predetermined and fixed) directions to ensure descent using a sufficiently small stepsize. The directions are typically required to form a *positive spanning set* (i.e., a set of vectors whose conic hull is $\mathbb{R}^n$) in order to make sure that each point in $\mathbb{R}^n$ (and hence also the optimal solution) is achievable by a sequence of positive steps from any starting point.

For instance, the *coordinate search* method uses the coordinate (i.e., standard basis) directions $e_1, e_2, \ldots, e_n$ and their negatives $-e_1, -e_2, \ldots, -e_n$ as the set of admissible directions. Clearly, $\{\pm e_i, : i = 1, 2, \ldots, n\}$ forms a positive spanning set.

1.1. Our method and complexity results. In this paper, we study a very general *randomized* variant of direct search methods, which we call the *stochastic three points* (`STP`) method. `STP` depends on two "parameters": a distribution/probability law $\mathcal{D}$ from which we sample directions, and a stepsize selection rule. At iteration k of `STP`, we generate a random direction s_k by sampling from $\mathcal{D}$ and then choose the next iterate via

$$x_{k+1} = \arg\min \{f(x_k + \alpha_k s_k), f(x_k - \alpha_k s_k), f(x_k)\},$$

where $\alpha_k > 0$ is an appropriately chosen stepsize. That is, we pick x_{k+1} as the best of the three points $x_k + \alpha_k s_k, x_k - \alpha_k s_k$, and x_k in terms of the function values.

In the nonconvex smooth regime we prove that the number of iterations of `STP` sufficient to guarantee that

$$\min_{k=0,1,\ldots,K} \mathbf{E}\left[\|\nabla f(x_k)\|_{\mathcal{D}}\right] \leq \varepsilon$$

is $O(\mu_D^{-2}\varepsilon^{-2})$,[1] where $\mathbf{E}[\cdot]$ refers to expectation with respect to the randomness inherent in the algorithm, $\|\cdot\|_{\mathcal{D}}$ is a norm, and $\mu_D > 0$ an associated constant depending on the choice of the distribution $\mathcal{D}$ (see section 3.1). For instance, when $\|\cdot\|_{\mathcal{D}}$ coincides with the ℓ_2 norm, then $\mu_D^{-2} \sim n$, and thus the complexity is $O(n\varepsilon^{-2})$. This complexity is global since no assumption is made on the starting point. If the objective function f is convex, then the number of iterations sufficient to find point x_k for which

$$\mathbf{E}\left[f(x_k)\right] - f_* \leq \varepsilon$$

is $O(\mu_D^{-2}\varepsilon^{-1})$, where f_* is the optimal value of problem (1.1). If f is strongly convex, we obtain a global linear rate of convergence. Our results have the same dependence on ε as deterministic direct search (`DDS`). However, while the best known complexity bounds for `DDS` (when using the ℓ_2 norm) depend quadratically on the dimension n, our results depend *linearly* on n [14, 22, 5].

1.2. Related literature. While our approach shares similarities with other randomized algorithmic approaches, the differences are significant. In the 1960s, a simple random optimization approach was proposed by Matyas [15]: sample a point randomly around the current iterate and move to this new point if it decreases the objective function. This was later generalized to cover constrained problems by Baba [2]. The theoretical and numerical performances of this approach for nonconvex functions were studied by Dorea [7] and Sarma [20].

More recently, the works of Diniz-Ehrhardt, Martínez, and Raydan [4] and Gratton et al. [9] use random search directions while imposing a sufficient decrease condition on whether to accept the step or reject it, like in `DDS`. They update the stepsize by

[1] One can use Markov's inequality to leverage the results in expectation to high probability results.

not decreasing it if the step is accepted and decreasing it otherwise. Our approach is different from these frameworks in the sense that at each iteration we generate a single direction and choose the stepsize independently from any decrease condition. In [9], the authors require the search direction to satisfy a certain strong probabilistic property: they assume that at each iteration the random directions lead to probabilistic descent conditioned on the past. In other words, at a given iteration, independently from the past with a certain probability, at least one of the directions needs to be of a descent type. The main result of [9] is the complexity bound $O(rn\varepsilon^{-2})$ to drive the ℓ_2 norm of the gradient below ε with high probability, where $r \geq 2$ is the number of the random directions at each iteration. Our `STP` method gives a similar complexity bound for nonconvex problems (with $r = 2$). Further, while [9] does not cover the cases when the objective function is convex or strongly convex, we do.

More related to our work is the method proposed by Karmanov [11, 12] in the 1970s for convex problems, where at iteration k the step is updated via

$$x_{k+1} = x_k + \alpha_k u,$$

where u is sampled independently from the uniform distribution on the unit sphere, and

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}} f(x_k + \alpha u).$$

This method was recently improved in two ways by Stich, Muller, and Gartner [21]: (i) by allowing for an approximate line search, i.e., $\alpha_k \approx \arg\min_{\alpha\in\mathbb{R}} f(x_k + \alpha u)$, and (ii) by allowing a discrete sampling from $\{\pm e_i, i = 1, \ldots, n\}$ instead of sampling from the unit sphere.

Our approach does not need to perform any line search approximation to compute the stepsizes, and it allows different distributions (which include the uniform distribution over the unit sphere and the discrete sampling from the canonical basis of $\mathbb{R}^n$) to sample the directions. The complexity bounds given in [11, 12, 21] are worse than those obtained in this paper.

Another algorithm related to our work is due to Polyak [19, section 3.4]; this is a derivative-free approach based on forming an unbiased estimate of the gradient using Gaussian smoothing. The search direction in this method is distributed uniformly over the unit sphere, and it is premultiplied by an approximation of the directional derivative along the direction itself. More precisely, this method at iteration k computes the iteration x_{k+1} as follows:

$$x_{k+1} = x_k - \alpha_k \frac{f(x_k + \mu_k u) - f(x_k)}{\mu_k} u, \tag{1.2}$$

where $\mu_k \in (0, 1)$ is the finite differences parameter, α_k is the stepsize, and u is a random vector distributed uniformly over the unit sphere. In this work, there are no explicit rules for choosing the parameters, and there is no analysis of the worst case complexity. Nesterov and Spokoiny [18] proposed novel variants of this method by changing the way the directional derivative of f is approximated along u and performed a worst case complexity analysis of Polyak's method (1.2). While the complexity bounds in [18] are similar to those of our `STP` approach, our approach is different from the method (1.2) and its variants proposed in [18]. In particular, in our approach the search direction can follow a virtually arbitrary distribution and not merely the uniform distribution over the unit sphere. For instance, we allow a distribution that has all its mass concentrated on a discrete set of vectors, which makes a direct connection with the (deterministic) direct search methods. Moreover, the proposed stepsizes in [18] depend on the Lipschitz constant of the gradient of

the objective function. However, in our approach we also propose new stepsize rules which can be easily executed in practice.

1.3. Outline. We organize this paper as follows. In section 2 we summarize the contributions of this paper. In section 3 we present our STP method and state some of its properties. In section 3.1 we describe the main assumptions on the random directions which ensure the convergence of our method. Then, in section 3.2 we introduce the key lemma for the iteration complexity analysis. In section 4 we analyze the worst case complexity of our method for smooth nonconvex problems, in section 5 we deal with convex problems, and in section 6 we focus on strongly convex problems. Numerical tests are discussed in section 7. Finally, some concluding remarks are expressed in section 8.

1.4. Notation. Throughout this paper, $\mathcal{D}$ denotes a probability distribution over $\mathbb{R}^n$. We use $\mathbf{E}[\cdot]$ to denote the expectation, and $\langle x, y\rangle = x^\top y$ corresponds to the standard Euclidean inner product of x and y. We denote also by $\|\cdot\|_2$ the ℓ_2 norm and by $\|\cdot\|_{\mathcal{D}}$ a norm dependent on $\mathcal{D}$ which we introduce in section 3.1.

2. Summary of contributions. In this section we highlight some of the key contributions of this work.

Simplicity and flexibility. We develop and study a novel variant of direct search based on random directions, which we call the STP method. Besides the starting point x_0, our method can be described by choosing a probability distribution $\mathcal{D}$ on $\mathbb{R}^n$, from which update directions are sampled, and a sequence of stepsizes $\{\alpha_k\}$. In some cases, only the initial stepsize α_0 needs to be chosen a priori. The probability distribution $\mathcal{D}$ may be iteration dependent as long as it satisfies a certain technical (and rather weak) assumption (see Assumption 5.4). In fact, this assumption can be weakened by allowing the probability distribution to depend on the iteration k in the following way:

- The quantity $\gamma_{\mathcal{D}_k} \overset{\text{def}}{=} \mathbf{E}_{s\sim\mathcal{D}_k}\|s\|_2^2$ is uniformly (in k) bounded away from zero.
- There is a constant $\mu_{\mathcal{D}} > 0$ and norm $\|\cdot\|_{\mathcal{D}}$ (independent from k) on $\mathbb{R}^n$ such that

$$\mathbf{E}_{s\sim\mathcal{D}_k} \mid \langle g_k, s\rangle \mid \geq \mu_{\mathcal{D}}\|g_k\|_{\mathcal{D}}, \tag{2.1}$$

  where $g_k = \nabla f(x_k)$.

This assumption may be weakened further by letting $\mu_{\mathcal{D}}$ and norm $\|\cdot\|_{\mathcal{D}}$ depend on k and assuming (i) the uniform boundedness of $\mu_{\mathcal{D}_k}$ away from zero and (ii) uniform equivalence of $\|\cdot\|_{\mathcal{D}_k}$ to a norm independent of k. To avoid excessive notation and for the sake of clarity and simplicity of the presentation, for the analysis we choose the probability distribution to be iteration independent in this paper.

Generality. Our approach covers some very popular first order methods:

- The normalized gradient descent (NGD) method: at iteration k, $s \sim \mathcal{D}$ means that $s = \frac{g_k}{\|g_k\|_2}$ with probability 1.
- The signed gradient descent (SignGD) method: at iteration k, $s \sim \mathcal{D}$ means that $s = sign(g_k)$ with probability 1, where the *sign* operation is element-wise sign.
- The normalized randomized coordinate descent (NRCD) method:[2] at iteration k, $s \sim \mathcal{D}$ means that $s = \frac{g_k^i}{|g_k^i|}e_i$ if $g_k^i \neq 0$ and $s = 0$ otherwise, with probability $\frac{1}{n}$, where g_k^i is the ith component of g_k.

[2]This method could also be called randomized signed gradient descent.

- The normalized stochastic gradient descent (NSGD) method: at iteration k, $s \sim \mathcal{D}$ means that $s = \hat{g}_k$, where $\hat{g}_k$ is the stochastic gradient satisfying $\mathbf{E}[\hat{g}_k] = \frac{g_k}{\|g_k\|_2}$, and $\mathbf{E}[\|\hat{g}_k\|_2^2] \leq \sigma < \infty$.

The required assumption on $\mathcal{D}$ is satisfied in these cases (see Appendix B).

The probability distribution is also allowed to be either continuous or discrete so that we cover many known strategies for choosing the directions in the DFO setting in the literature. For instance, if $\mathcal{D}$ is the uniform law on the unit sphere, we recover the directions proposed in [11, 12, 19, 18]. If it is the discrete law on $\{\pm e_i, i = 1, \ldots, n\}$, we recover the directions proposed in [21]. If it is the discrete law on $\{\pm d_i, i = 1, \ldots, n\}$, where $d_i,\ i = 1, \ldots, n$ form a basis of $\mathbb{R}^n$, STP can be seen as a random variant of the simplified direct search method studied in [14].

One of the main reasons for us to allow a virtually full flexibility in choosing the probability distribution $\mathcal{D}$ is the possibility to adapt it to the characteristics of the optimization problem (1.1) in order to achieve greater efficiency:

- The dimension n could be so large that methods relying on the addition of two vectors in $\mathbb{R}^n$ in each iteration become infeasible. Therefore, one needs to rely on methods which update a small number of the components of x at each iteration.
- The objective function might not be entirely defined at the beginning of the optimization process, such as is the case in streaming optimization. In other words, the data describing the objective function arrives in real time during the optimization process. At a given iteration (time) we cannot evaluate the objective function in all points of $\mathbb{R}^n$ but can evaluate the objective function in a set of directions (only some components of x can be updated).
- Even if the entire objective function f is available at the beginning of the optimization process, for some problems the computation of the function value increases with the number of the perturbed variables. In other words, when perturbing all the components of x, the evaluation of f takes a lot of time. However, by perturbing only one parameter (or a set of parameters), the objective is evaluated in reasonable time.
- Prior knowledge about Lipschitz constants in some directions might be available.

In these kinds of situations, choosing $\mathcal{D}$ as a continuous law is not practical. On the other hand, specific choices of discrete laws $\mathcal{D}$ are suitable.

Practicality. The STP method is extremely simple to use in practice, and its analysis is also simple compared to the state-of-the-art direct search methods based on random directions/stepsizes. Perhaps the work most related to ours is that of Nesterov and Spokoiny [18]. However, their stepsizes depend on the Lipschitz constant of the gradient of the objective function, which may not be known in practice. In contrast, for STP we proposed several stepsize selection schemes, some of which can be easily implemented in practice. Moreover, our preliminary numerical experiments show that our approach is competitive in practice.

Better bounds. We obtain compact worst case complexity bounds similar to those obtained in [18]. They depend linearly on the dimension of the considered problem, while this dependence is quadratic for DDS methods [22, 5, 14]. In Table 1 we summarize selected complexity results (bounds on the number of function evaluations) obtained in this paper for STP method. In all cases we assume that f is differentiable, bounded below (by f_*), with L-Lipschitz gradient. The assumptions listed in the first column of the table are additional to this. The quantity R_0 measures the size of a

TABLE 1

Summary of the complexity results obtained in this paper for the STP method. Column "Complexity" defines the number of iterations needed to guarantee $\min_k \mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}] \leq \varepsilon$ *(second row) or* $\mathbf{E}[f(x_k) - f(x_*)] \leq \varepsilon$ *(third and fourth rows).*

Assumptions on f (additional to L-smoothness)	Stepsizes	Complexity	Theorems
none	$\alpha_k \propto \frac{1}{\sqrt{k+1}}$ $\alpha_k \propto \varepsilon$	$O\left(\frac{n}{\varepsilon^2}\right)$	4.1, 4.2
convex, R_0 finite	$\alpha_k \propto f(x_k) - f(x_*)$ $\alpha_k \propto \frac{\lvert f(x_k+ts_k)-f(x_k)\rvert}{t}$	$O\left(\frac{n}{\varepsilon}\right)$	5.3, 5.5
λ-strongly convex	$\alpha_k \propto (f(x_k) - f(x_*))^{\frac{1}{2}}$ $\alpha_k \propto \frac{\lvert f(x_k+ts_k)-f(x_k)\rvert}{t}$	$O\left(n \log\left(\frac{1}{\varepsilon}\right)\right)$	6.2, 6.3

specific level set of f. The symbol $\propto$ means "proportional to." We use this symbol often in the definition of the stepsizes; for instance, $\alpha_k \propto \frac{1}{\sqrt{k+1}}$ means that α_k is equal to some constant α_0 (independent from k) multiplied by $\frac{1}{\sqrt{k+1}}$. The optimal (or near to the optimality) α_0 usually depends on quantities such as the Lipschitz constant and/or the initial iterate x_0. More details about the definitions of all these quantities are given in the main text.

Experiments. We provide a number of experimental results, showing that STP is a competitive algorithm in practice. We have tested our method on a large set of problems against Polyak's method (1.2) and coordinate search method (the DDS method which uses the $2n$ coordinate directions). The experiments show that the use of tailored random directions leads to a significant improvement in terms of the number of function evaluations. Indeed, STP and Polyak's method (1.2) outperform the DDS method. Moreover, our approach exhibits better performance than the other two methods. See section 7 for a complete description of our experimental results.

3. STP method.

Our STP algorithm is formalized below as Algorithm 3.1.

Algorithm 3.1 STP Method.

Initialization

Choose starting iterate $x_0 \in \mathbb{R}^n$, positive stepsizes $\{\alpha_k\}_{k\geq 0}$, probability distribution $\mathcal{D}$ on $\mathbb{R}^n$.

For $k = 0, 1, 2, \ldots$

1. Generate a random vector $s_k \sim \mathcal{D}$
2. Let $x_+ = x_k + \alpha_k s_k$ and $x_- = x_k - \alpha_k s_k$
3. $x_{k+1} = \arg\min\{f(x_-), f(x_+), f(x_k)\}$

Due to the randomness of the search directions s_k for $k \geq 0$, the iterates are also random vectors[3] for all $k \geq 1$. Note that STP never moves to a point with a larger

[3]We do not consider the starting point x_0 to be random, but all our results hold also in this case with very minor and obvious modifications.

objective value. This monotonicity property does not depend on $\mathcal{D}$ or the properties of f. Let us formulate this simple observation as a lemma.

LEMMA 3.1 (monotonicity). *STP produces a monotonic sequence of iterates, i.e., $f(x_{k+1}) \leq f(x_k)$ for all $k \geq 0$. As a consequence,*

$$\mathbf{E}[f(x_{k+1}) \mid x_k] \leq f(x_k). \tag{3.1}$$

Throughout the paper, we assume that f is differentiable and bounded below and has L-Lipschitz gradient.

Assumption 3.2. The objective function f is L-smooth with $L > 0$ and bounded from below by $f_* \in \mathbb{R}$. That is, f has a Lipschitz continuous gradient with Lipschitz constant L:

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2 \qquad \forall x, y \in \mathbb{R}^n$$

and $f(x) \geq f_*$ for all $x \in \mathbb{R}^n$.

3.1. Random search directions. Our analysis in the rest of the paper is based on the following key assumption.

Assumption 3.3. The probability distribution $\mathcal{D}$ on $\mathbb{R}^n$ has the following properties:

1. The quantity $\gamma_{\mathcal{D}} \overset{\text{def}}{=} \mathbf{E}_{s\sim\mathcal{D}} \|s\|_2^2$ is positive and finite.
2. There is a constant $\mu_{\mathcal{D}} > 0$ and norm $\|\cdot\|_{\mathcal{D}}$ on $\mathbb{R}^n$ such that for all $g \in \mathbb{R}^n$,

$$\mathbf{E}_{s\sim\mathcal{D}} \mid \langle g, s\rangle \mid \geq \mu_{\mathcal{D}}\|g\|_{\mathcal{D}}. \tag{3.2}$$

Note that since all norms in $\mathbb{R}^n$ are equivalent, the second part of the above assumption is satisfied if and only if

$$\inf_{\|g\|_2=1} \mathbf{E}_{s\sim\mathcal{D}} \mid \langle g, s\rangle \mid > 0.$$

However, as the next lemma illustrates, it will be convenient to work with norms that are allowed to depend on $\mathcal{D}$. We now give some examples of distributions for which the above assumption is satisfied.

LEMMA 3.4. *Let $g \in \mathbb{R}^n$.*

1. *If $\mathcal{D}$ is the uniform distribution on the unit sphere in $\mathbb{R}^n$, then*

$$\gamma_{\mathcal{D}} = 1 \quad \text{and} \quad \mathbf{E}_{s\sim\mathcal{D}} \mid \langle g, s\rangle \mid \sim \frac{1}{\sqrt{2\pi n}}\|g\|_2. \tag{3.3}$$

   Hence, $\mathcal{D}$ satisfies Assumption 3.3 *with $\gamma_{\mathcal{D}} = 1$, $\|\cdot\|_{\mathcal{D}} = \|\cdot\|_2$, and $\mu_{\mathcal{D}} \sim \frac{1}{\sqrt{2\pi n}}$.*
2. *If $\mathcal{D}$ is the normal distribution with zero mean and $n \times n$ identity as the covariance matrix, i.e., $s \sim N(0, \frac{I}{n})$, then*

$$\gamma_{\mathcal{D}} = 1 \quad \text{and} \quad \mathbf{E}_{s\sim\mathcal{D}} \mid \langle g, s\rangle \mid = \frac{\sqrt{2}}{\sqrt{n\pi}}\|g\|_2. \tag{3.4}$$

   Hence, $\mathcal{D}$ satisfies Assumption 3.3 *with $\gamma_{\mathcal{D}} = 1$, $\|\cdot\|_{\mathcal{D}} = \|\cdot\|_2$, and $\mu_{\mathcal{D}} = \frac{\sqrt{2}}{\sqrt{n\pi}}$.*
3. *If $\mathcal{D}$ is the uniform distribution over standard unit basis vectors $\{e_1, \ldots, e_n\}$ in $\mathbb{R}^n$, then*

$$\gamma_{\mathcal{D}} = 1 \quad \text{and} \quad \mathbf{E}_{s\sim\mathcal{D}} \mid \langle g, s\rangle \mid = \frac{1}{n}\|g\|_1. \tag{3.5}$$

   Hence, $\mathcal{D}$ satisfies Assumption 3.3 *with $\gamma_{\mathcal{D}} = 1$, $\|\cdot\|_{\mathcal{D}} = \|\cdot\|_1$, and $\mu_{\mathcal{D}} = \frac{1}{n}$.*

4. *If $\mathcal{D}$ is a distribution on $D = \{d_1, \dots, d_n\}$ where $d_1, \dots, d_n$ form an orthonormal basis of $\mathbb{R}^n$ and $P(s = d_i) = p_i$, then*

$$\gamma_{\mathcal{D}} = 1 \quad \text{and} \quad \mathbf{E}_{s\sim\mathcal{D}} \,|\, \langle g, s\rangle \,| = \|g\|_{\mathcal{D}} \overset{def}{=} \sum_{i=1}^{n} p_i |g_i|. \tag{3.6}$$

Hence, $\mathcal{D}$ satisfies Assumption 3.3 with $\gamma_{\mathcal{D}} = 1$ and $\mu_{\mathcal{D}} = 1$.

Proof. See Appendix A. ∎

Without loss of generality, in the rest of this paper we assume that $\gamma_{\mathcal{D}} = 1$. This can be achieved by considering distribution $\mathcal{D}'$ instead, where $s' \sim \mathcal{D}'$ is obtained by first sampling s' from $\mathcal{D}$ and then normalizing via either (i) $s = s'/\|s'\|_2$ or (ii) $s = s'/\sqrt{\mathbf{E}_{s'\sim\mathcal{D}}\|s'\|_2^2}$.

3.2. Key lemma. We now establish a key result which will be used to prove the main properties of our algorithm. An analogous result in the case of DDS methods states that the gradient of the objective function for unsuccessful iterations is bounded by a constant multiplied by the stepsize. See, for instance, [14, Lemma 10].

LEMMA 3.5. *If Assumptions 3.2 and 3.3 hold, then for all $k \geq 0$,*

$$\mathbf{E}\left[f(x_{k+1}) \mid x_k\right] \leq f(x_k) - \mu_{\mathcal{D}}\alpha_k \|\nabla f(x_k)\|_{\mathcal{D}} + \frac{L}{2}\alpha_k^2 \tag{3.7}$$

and

$$\theta_{k+1} \leq \theta_k - \mu_{\mathcal{D}}\alpha_k g_k + \frac{L}{2}\alpha_k^2, \tag{3.8}$$

where $\theta_k = \mathbf{E}[f(x_k)]$ and $g_k = \mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}]$.

Proof. Notice that from L-smoothness of f we have

$$\begin{aligned} f(x_k + \alpha_k s_k) &\leq f(x_k) + \langle \nabla f(x_k), \alpha_k s_k\rangle + \frac{L}{2}\|\alpha_k s_k\|_2^2 \\ &= f(x_k) + \alpha_k \langle \nabla f(x_k), s_k\rangle + \frac{L}{2}\alpha_k^2 \|s_k\|_2^2, \end{aligned}$$

and, similarly, $f(x_k - \alpha_k s_k) \leq f(x_k) - \alpha_k \langle \nabla f(x_k), s_k\rangle + \frac{L}{2}\alpha_k^2\|s_k\|_2^2$. Hence,

$$\begin{aligned} f(x_{k+1}) &\leq \min\{f(x_k + \alpha_k s_k), f(x_k - \alpha_k s_k)\} \\ &\leq f(x_k) - \alpha_k |\langle \nabla f(x_k), s_k\rangle| + \frac{L}{2}\alpha_k^2\|s_k\|_2^2. \end{aligned}$$

To conclude (3.7), we only need to take expectation in the above inequality with respect to $s_k \sim \mathcal{D}$, conditional on x_k, and use inequality (3.2). By taking the expectation in (3.7), we get (3.8). ∎

Note that (3.7) can equivalently be written in the following form:

$$\|\nabla f(x_k)\|_{\mathcal{D}} \leq \frac{1}{\mu_{\mathcal{D}}}\left(\frac{f(x_k) - \mathbf{E}\left[f(x_{k+1}) \mid x_k\right]}{\alpha_k} + \frac{L}{2}\alpha_k\right),$$

which makes it possible to compare the result with a key result used in the analysis of DDS. Indeed, if we assume that the opposite of the sufficient *expected* decrease condition

$$f(x_k) - \mathbf{E}\left[f(x_{k+1}) \mid x_k\right] \geq c\alpha_k^2 \tag{3.9}$$

holds for some $c > 0$, then we obtain

$$\|\nabla f(x_k)\|_{\mathcal{D}} \leq \frac{1}{\mu_{\mathcal{D}}}\left(c + \frac{L}{2}\right)\alpha_k. \tag{3.10}$$

In DDS, condition (3.9) is equivalent to the sufficient decrease condition $f(x_k) - f(x_{k+1}) \geq c\alpha_k^2$. If this condition does not hold, then the step is declared unsuccessful. Inequality (3.10) is similar to the result in [14, Lemma 10]. In DDS methods, one can check the sufficient decrease condition, and this drives the analysis and allows simple stepsize update rules to be implemented. In STP, we typically cannot evaluate $\mathbf{E}[f(x_{k+1}) \mid x_k]$ (we can if $\mathcal{D}$ has all its mass on a discrete set, but in that case we would need to do more work per iteration).

4. Nonconvex problems. In this section, we state our most general complexity result where we do not make any additional assumptions on f besides smoothness and boundedness (see Assumption 3.2).

4.1. Decreasing stepsize. We first state a complexity theorem for STP used with a decreasing stepsize.

THEOREM 4.1 (decreasing stepsize). *Let Assumptions* 3.2 *and* 3.3 *hold. Choose* $\alpha_k = \frac{\alpha_0}{\sqrt{k+1}}$, *where* $\alpha_0 > 0$. *If*

$$K \geq \frac{2\left(\frac{\sqrt{2}(f(x_0)-f_*)}{\alpha_0} + \frac{L\alpha_0}{2}\right)^2}{\mu_{\mathcal{D}}^2\varepsilon^2}, \tag{4.1}$$

then $\min_{k=0,1,\ldots,K} \mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}] \leq \varepsilon$.

Proof. We base the proof on the analysis of the recursion (3.8). In particular, it is useful to write it in the following form:

$$g_k \leq \frac{1}{\mu_{\mathcal{D}}}\left(\frac{\theta_k - \theta_{k+1}}{\alpha_k} + \frac{L}{2}\alpha_k\right) = \frac{1}{\mu_{\mathcal{D}}}\left(\frac{(\theta_k - \theta_{k+1})\sqrt{k+1}}{\alpha_0} + \frac{L\alpha_0}{2\sqrt{k+1}}\right). \tag{4.2}$$

We know from (3.1) and the assumption that f is bounded below that $f_* \leq \theta_{k+1} \leq \theta_k \leq f(x_0)$ for all k. Letting $l = \lfloor K/2 \rfloor$, this implies that

$$\sum_{j=l}^{2l}(\theta_j - \theta_{j+1}) = \theta_l - \theta_{2l+1} \leq f(x_0) - f_* \stackrel{\text{def}}{=} C,$$

from which we conclude that there must exist $j \in \{l, \ldots, 2l\}$ such that $\theta_j - \theta_{j+1} \leq C/(l+1)$. This implies that

$$\begin{aligned} g_j &\overset{(4.2)}{\leq} \frac{1}{\mu_{\mathcal{D}}}\left(\frac{(\theta_j - \theta_{j+1})\sqrt{j+1}}{\alpha_0} + \frac{L\alpha_0}{2\sqrt{j+1}}\right) \leq \frac{1}{\mu_{\mathcal{D}}}\left(\frac{C\sqrt{j+1}}{\alpha_0(l+1)} + \frac{L\alpha_0}{2\sqrt{j+1}}\right) \\ &\leq \frac{1}{\mu_{\mathcal{D}}}\left(\frac{C\sqrt{2l+1}}{\alpha_0(l+1)} + \frac{L\alpha_0}{2\sqrt{l+1}}\right) \leq \frac{1}{\mu_{\mathcal{D}}\sqrt{l+1}}\left(\frac{\sqrt{2}C}{\alpha_0} + \frac{L\alpha_0}{2}\right) \\ &\leq \frac{1}{\mu_{\mathcal{D}}\sqrt{K/2}}\left(\frac{\sqrt{2}C}{\alpha_0} + \frac{L\alpha_0}{2}\right) \overset{(4.1)}{\leq} \varepsilon, \end{aligned}$$

which finishes the proof. ∎

Let us now give some insights into the above theorem.

- **Sphere setup.** If $\mathcal{D}$ is the uniform distribution on the Euclidean sphere, then $\mu_{\mathcal{D}} \sim \frac{1}{\sqrt{2\pi n}}$, and hence the above theorem gives a complexity guarantee of the form
$$O\left(\frac{n}{\varepsilon^2}\right).$$
This is an improvement on DDS, where the best known complexity bound is $O(n^2/\varepsilon^2)$ [22, 14]. The same conclusion holds for the normal distribution setup.
- **Coordinate setup.** If $\mathcal{D}$ is the uniform distribution on $\{e_1, \ldots, e_n\}$, then $\mu_{\mathcal{D}} = 1/n$, and hence the bound is of the form
$$O\left(\frac{n^2}{\varepsilon^2}\right).$$
However, this is for the ℓ_1 norm of the gradient of f, which is *larger* than the ℓ_2 norm. Indeed, for all x we have $\sqrt{n}\|\nabla f(x)\|_2 \geq \|\nabla f(x)\|_1 \geq \|\nabla f(x)\|_2$, and the first inequality can be tight (for the vector of all ones, for instance). Hence, if we are interested to achieve $\|\nabla f(x)\|_2 \leq \varepsilon'$, in certain situations it may be sufficient to push the ℓ_1 norm of the gradient below $\varepsilon = \sqrt{n}\varepsilon'$ instead. So, the iteration bound can be as good as
$$O\left(\frac{n^2}{(\sqrt{n}\varepsilon')^2}\right) = O\left(\frac{n}{(\varepsilon')^2}\right).$$
- **Quality of the final iterate.** Theorem 4.1 does not guarantee the gradient of f at the *final* point x_K to be small (in expectation). Instead, it guarantees that the gradient of f at *some* point produced by the method will be small. Notice however, that the method is monotonic. Hence, all subsequent points produced by the method will have better function values than the one which has gradient of minimum norm (in expectation). So, we can say that $f(x_K) \leq f(x_j)$, where $\mathbf{E}[\|\nabla f(x_j)\|_{\mathcal{D}}] \leq \varepsilon$.
- **Optimal stepsize.** Note that the complexity depends on α_0. The optimal choice (minimizing the complexity bound) is
$$\alpha^* = 8^{1/4}\sqrt{\frac{f(x_0) - f_*}{L}},$$
in which case the complexity bound (4.1) takes the form
$$\frac{4\sqrt{2}(f(x_0) - f_*)L}{\mu_{\mathcal{D}}^2 \varepsilon^2}. \tag{4.3}$$
Assume that the lower bound f_* is achieved by some point $x_* \in \mathbb{R}^n$. Necessarily, $\nabla f(x_*) = 0$. Moreover, since f is L-smooth, we can write
$$f(x_0) \leq f(x_*) + \langle \nabla f(x_*), x_0 - x_* \rangle + \frac{L}{2}\|x_0 - x_*\|_2^2.$$
Hence, the optimal initial stepsize is no larger than
$$\alpha^* \leq 2^{1/4}\|x_0 - x_*\|_2.$$

Of course, we cannot use this initial optimal stepsize as we usually do not know L and/or f_*. So, we are paying for the lack of knowledge by an increased complexity bound. This makes intuitive sense: the stepsize should not be much larger than the distance of the initial point to an optimal point. On the other hand, there are examples of nonconvex functions for which the ratio $(f(x_0) - f_*)/L$ is arbitrarily small and the distance between x_0 and x_* arbitrarily high. This cannot happen for convex functions with bounded level sets or for strongly convex functions, as then $f(x) - f(x_*)$ can be lower bounded by a quantity proportional to $\|x - x_*\|_2$ with some positive power.

4.2. Constant stepsize. We now state a complexity theorem for STP used with a constant stepsize.

THEOREM 4.2 (constant stepsize). *Let Assumption* 3.2 *hold. Choose a fixed stepsize* $\alpha_k = \alpha$ *with* $0 < \alpha < 2\mu_{\mathcal{D}}\varepsilon/L$. *If*

$$K \geq k(\varepsilon) \overset{def}{=} \left\lceil \frac{f(x_0) - f_*}{(\mu_{\mathcal{D}}\varepsilon - \frac{L}{2}\alpha)\alpha} \right\rceil - 1, \tag{4.4}$$

then $\min_{k=0,1,\ldots,K} \mathbf{E}\left[\|\nabla f(x_k)\|_{\mathcal{D}}\right] \leq \varepsilon$. *In particular, if* $\alpha = \mu_{\mathcal{D}}\varepsilon/L$, *then*

$$k(\varepsilon) = \left\lceil \frac{2L(f(x_0) - f_*)}{\mu_{\mathcal{D}}^2\varepsilon^2} \right\rceil - 1.$$

Proof. If $g_k \leq \varepsilon$ for some $k \leq k(\varepsilon)$, then we are done. Assume hence by contradiction that $g_k > \varepsilon$ for all $k \leq k(\varepsilon)$. By taking expectation in Lemma 3.5, we get

$$\theta_{k+1} \leq \theta_k - \mu_{\mathcal{D}}\alpha g_k + \frac{L}{2}\alpha^2,$$

where $\theta_k = \mathbf{E}[f(x_k)]$ and $g_k = \mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}]$. Hence,

$$f_* \leq \theta_{K+1} < \theta_0 - (K+1)\left(\mu_{\mathcal{D}}\alpha\varepsilon - \frac{L}{2}\alpha^2\right) \overset{(4.4)}{\leq} \theta_0 - (f(x_0) - f_*) = f_*,$$

which is a contradiction. ∎

We now offer some comments:

- In some situations, when L is not available, it is impossible to compute optimal $\alpha = \frac{\mu_{\mathcal{D}}\varepsilon}{L}$.
- If we can guess α close to the optimal, then the method depends linearly on n if $1/\mu_{\mathcal{D}}^2 = O(n)$.
- Also, by using the optimal α, we get complexity that depends on $L(f(x_0) - f_*)$, which is similar to the setup with variable stepsizes and optimal α_0.
- As before, we only get a guarantee on the best of the points in terms of the gradient norm, not on the final point.

5. Convex problems. In this section we estimate the complexity of STP in the case of convex f. In this case we need an additional technical assumption.

Assumption 5.1. We assume that f is convex, has a minimizer x_*, and has bounded level set at x_0:

$$R_0 \overset{\text{def}}{=} \max\left\{\|x - x_*\|_{\mathcal{D}}^* \ : \ f(x) \leq f(x_0)\right\} < +\infty,$$

where $\|\xi\|_{\mathcal{D}}^* \overset{\text{def}}{=} \max\{\langle \xi, x\rangle \mid \|x\|_{\mathcal{D}} \leq 1\}$ defines the dual norm to $\|\cdot\|_{\mathcal{D}}$.

Note that if the above assumption holds, then whenever $f(x) \leq f(x_0)$, we get $f(x) - f(x_*) \leq \langle \nabla f(x), x - x_* \rangle \leq \|\nabla f(x)\|_{\mathcal{D}} \|x - x_*\|_{\mathcal{D}}^* \leq R_0 \|\nabla f(x)\|_{\mathcal{D}}$. That is,

$$\|\nabla f(x)\|_{\mathcal{D}} \geq \frac{f(x) - f(x_*)}{R_0}. \tag{5.1}$$

Now, we state our main complexity result of this section.

5.1. Constant stepsize. We start with the analysis of STP with constant stepsizes.

THEOREM 5.2 (constant stepsize). *Let Assumptions* 3.2, 3.3, *and* 5.1 *be satisfied. Let* $0 < \varepsilon < \frac{LR_0^2}{\mu_{\mathcal{D}}^2}$, *and choose the constant stepsize* $\alpha_k = \alpha = \frac{\varepsilon \mu_{\mathcal{D}}}{LR_0}$. *If*

$$K \geq \frac{LR_0^2}{\mu_{\mathcal{D}}^2 \varepsilon} \log\left(\frac{2(f(x_0) - f(x_*))}{\varepsilon}\right), \tag{5.2}$$

then $\mathbf{E}[f(x_K)] - f(x_*) \leq \varepsilon$.

Proof. Let us substitute (5.1) into Lemma 3.5 and take expectations. We get

$$\theta_{k+1} \leq \theta_k - \frac{\mu_{\mathcal{D}} \alpha}{R_0}(\theta_k - f(x_*)) + \frac{L}{2}\alpha^2. \tag{5.3}$$

Let $r_k = \theta_k - f(x_*)$ and $c = 1 - \frac{\mu_{\mathcal{D}} \alpha}{R_0} \in (0, 1)$. Subtracting $f(x_*)$ from both sides of (5.3), we obtain

$$\begin{aligned} r_K &\leq c r_{K-1} + \frac{L}{2}\alpha^2 \leq c^K r_0 + \frac{L}{2}\alpha^2 \sum_{i=0}^{K-1} c^i \\ &\leq \exp(-\mu_{\mathcal{D}} \alpha K / R_0) r_0 + \frac{L\alpha^2}{2(1-c)} = \exp(-\mu_{\mathcal{D}} \alpha K / R_0) r_0 + \frac{\varepsilon}{2} \overset{(5.2)}{\leq} \varepsilon \end{aligned}$$

which finishes the proof. ∎

If $\mu_{\mathcal{D}} \sim \frac{1}{\sqrt{n}}$, then the above theorem gives a complexity guarantee of the form

$$O\left(\frac{n}{\varepsilon} \log\left(\frac{1}{\varepsilon}\right)\right).$$

Comparing this to the best known complexity bound for DDS, which is $O(\frac{n^2}{\varepsilon})$ [5, 14], we improve the dependence on n. However, we deteriorate the $\frac{1}{\varepsilon}$ dependence on ε because of the presence of the term $\log(\frac{1}{\varepsilon})$.

5.2. Variable stepsize. In the next theorem we show how one can get rid of the $\log \frac{1}{\varepsilon}$ term using a variable stepsize.

THEOREM 5.3 (variable stepsize). *Let Assumptions* 3.2, 3.3, *and* 5.1 *be satisfied. Let* $\alpha_k = \alpha_0(f(x_k) - f(x_*))$, *where* $0 < \alpha_0 < \frac{2\mu_{\mathcal{D}}}{R_0 L}$. *Define* $a = \frac{\mu_{\mathcal{D}} \alpha_0}{R_0} - \frac{L\alpha_0^2}{2} > 0$. *If* $k \geq k(\varepsilon) \overset{def}{=} \frac{1}{a}(\frac{1}{\varepsilon} - \frac{1}{r_0})$, *then* $\mathbf{E}[f(x_k) - f(x_*)] \leq \varepsilon$.

Proof. Let us substitute (5.1) into (3.7) of Lemma 3.5. Subtracting $f(x_*)$ from both sides, we get

$$\mathbf{E}\left[f(x_{k+1}) \mid x_k\right] - f(x_*) \leq f(x_k) - f(x_*) - \mu_{\mathcal{D}} \alpha_k \frac{f(x_k) - f(x_*)}{R_0} + \frac{L}{2}\alpha_k^2.$$

Let $r_k = \mathbf{E}[f(x_k)] - f(x_*)$. By using our choice of α_k in the previous equation and then taking expectation, we get $r_{k+1} \leq r_k - (\frac{\mu_{\mathcal{D}}\alpha_0}{R_0} - \frac{L\alpha_0^2}{2})r_k^2 = r_k - ar_k^2$. Therefore,

$$\frac{1}{r_{k+1}} - \frac{1}{r_k} = \frac{r_k - r_{k+1}}{r_k r_{k+1}} \geq \frac{r_k - r_{k+1}}{r_k^2} \geq a.$$

From this we have $\frac{1}{r_k} \geq \frac{1}{r_0} + ka$ and hence $r_k \leq \frac{1}{\frac{1}{r_0}+ka}$. It remains to notice that for $k \geq \frac{1}{a}(\frac{1}{\varepsilon} - \frac{1}{r_0})$ we have $r_k \leq \frac{1}{\frac{1}{r_0}+ka} \leq \varepsilon$. ∎

If $\alpha_0 = \frac{\mu_{\mathcal{D}}}{R_0 L}$, then a is maximal as a function of α_0, for which we get the optimal bound

$$k(\varepsilon) = \frac{2R_0^2 L}{\mu_{\mathcal{D}}^2}\left(\frac{1}{\varepsilon} - \frac{1}{r_0}\right).$$

If $\mu_{\mathcal{D}} \sim \frac{1}{\sqrt{n}}$, then the above theorem gives a complexity guarantee of the form $O\left(\frac{n}{\varepsilon}\right)$.

5.3. Practical stepsize. The stepsizes in the previous theorem depend on the (in general) unknown quantity $f(x_*)$. The next theorem gives a more practical way of defining stepsizes for which we get the same complexity as in the previous theorem. We start by stating an extra assumption on the probability distribution $\mathcal{D}$ and show that this assumption is satisfied for all the probability distributions given in Lemma 3.4.

Assumption 5.4. All samples $s \sim \mathcal{D}$ are of unit Euclidean norm ($\|s\|_2 = 1$) with probability 1.

Let $C_{\mathcal{D}}$ be the positive constant for which the inequality $\|x\|_2 \leq C_{\mathcal{D}}\|x\|_{\mathcal{D}}$ holds for all $x \in \mathbb{R}^n$. Such a constant exists due to the equivalence of all norms in $\mathbb{R}^n$.

THEOREM 5.5 (practical stepsize). *Let Assumptions* 3.2, 3.3, 5.1, *and* 5.4 *be satisfied. Let* $\alpha_k = \frac{|f(x_k+ts_k)-f(x_k)|}{Lt}$, *where*

$$0 < t \leq \frac{\sqrt{2}\mu_{\mathcal{D}}\left(\mathbf{E}\left[f(x_{K-1})\right] - f_*\right)}{LR_0}.$$

Define $a = \frac{\mu_{\mathcal{D}}^2}{4LR_0^2}$. *If* $K \geq k(\varepsilon) \overset{def}{=} \frac{1}{a}(\frac{1}{\varepsilon} - \frac{1}{r_0})$, *then* $\mathbf{E}\left[f(x_K)\right] - f(x_*) \leq \varepsilon$.

Proof. From Lemma 3.5 we have

$$f(x_{k+1}) \leq f(x_k) - \alpha_k|\langle\nabla f(x_k), s_k\rangle| + \frac{L\alpha_k^2}{2}. \tag{5.4}$$

We know that $\alpha_k^{\text{opt}} = \frac{|\langle\nabla f(x_k), s_k\rangle|}{L}$ minimizes the right-hand side of (5.4). But it depends on $\nabla f(x_k)$ which we cannot compute *exactly*, because we have a zeroth order oracle. Actually, we do not need to know the whole gradient; it is enough to know the directional derivative of f, which we can approximate by finite differences, using function values. It is the main idea behind our choice of $\alpha_k^{\text{opt}} = \frac{|f(x_k+ts_k)-f(x_k)|}{Lt}$, which does not depend any more on $f(x_*)$ and can be easily computed in practice. We can rewrite $\alpha_k = \frac{|f(x_k+ts_k)-f(x_k)|}{Lt} = \frac{|\langle\nabla f(x_k), s_k\rangle|}{L} + \frac{|f(x_k+ts_k)-f(x_k)|}{Lt} - \frac{|\langle\nabla f(x_k), s_k\rangle|}{L} \overset{\text{def}}{=} \alpha_k^{\text{opt}} + \delta_k$. Therefore, we have

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) - \frac{|\langle\nabla f(x_k), s_k\rangle|^2}{L} - \delta_k|\langle\nabla f(x_k), s_k\rangle| + \frac{|\langle\nabla f(x_k), s_k\rangle|^2}{2L} \\ &\quad + \delta_k|\langle\nabla f(x_k), s_k\rangle| + \frac{L}{2}(\delta_k)^2 \\ &= f(x_k) - \frac{|\langle\nabla f(x_k), s_k\rangle|^2}{2L} + \frac{L}{2}(\delta_k)^2. \end{aligned}$$

Next, we estimate $|\delta_k|$ using the L-smoothness of f:

$$|\delta_k| = \frac{1}{Lt}\,||f(x_k + ts_k) - f(x_k)| - |\langle \nabla f(x_k), ts_k\rangle|| \\ \leq \frac{1}{Lt}\,|f(x_k + ts_k) - f(x_k) - \langle \nabla f(x_k), ts_k\rangle| \leq \frac{1}{Lt}\cdot\frac{L}{2}\|ts_k\|_2^2 = \frac{t}{2}.$$

From this we obtain

$$f(x_{k+1}) \quad \leq f(x_k) - \tfrac{|\langle \nabla f(x_k), s_k\rangle|^2}{2L} + \tfrac{Lt^2}{8}. \tag{5.5}$$

Taking mathematical expectation with respect to all randomness from the previous inequality, we get

$$\begin{aligned} \underbrace{\mathbf{E}[f(x_{k+1})] - f_*}_{r_{k+1}} &\overset{①}{\leq} \underbrace{\mathbf{E}[f(x_k)] - f_*}_{r_k} - \tfrac{\mu_{\mathcal{D}}^2}{2L}\mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}^2] + \tfrac{Lt^2}{8} \\ &\overset{②}{\leq} r_k - \tfrac{\mu_{\mathcal{D}}^2}{2LR_0^2} r_k^2 + \tfrac{Lt^2}{8}, \end{aligned} \tag{5.6}$$

where ① is due to the tower property of mathematical expectation and (3.2):

$$\begin{aligned} \mathbf{E}\left[|\langle \nabla f(x_k), s_k\rangle|^2\right] &= \mathbf{E}\left[\mathbf{E}\left[|\langle \nabla f(x_k), s_k\rangle|^2 \mid x_k\right]\right] \geq \mathbf{E}\left[\left(\mathbf{E}\left[|\langle \nabla f(x_k), s_k\rangle| \mid x_k\right]\right)^2\right] \\ &\overset{(3.2)}{\geq} \mu_{\mathcal{D}}^2 \mathbf{E}\left[\|\nabla f(x_k)\|_{\mathcal{D}}^2\right]; \end{aligned} \tag{5.7}$$

② follows from Assumption 5.1:

$$\mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}^2] \geq \frac{\mathbf{E}\left[\left(f(x_k) - f_*\right)^2\right]}{R_0^2} \geq \frac{\left(\mathbf{E}\left[f(x_k) - f_*\right]\right)^2}{R_0^2} = \frac{r_k^2}{R_0^2}.$$

From this and monotonicity of $\{f(x_k)\}_{k\geq 0}$ we have

$$\frac{1}{r_{k+1}} - \frac{1}{r_k} \geq \frac{r_k - r_{k+1}}{r_k r_{k+1}} \geq \frac{\frac{\mu_{\mathcal{D}}^2}{2LR_0^2} r_k^2 - \frac{Lt^2}{8}}{r_k^2} \geq \frac{\mu_{\mathcal{D}}^2}{2LR_0^2} - \frac{L}{8}\left(\frac{t}{r_k}\right)^2. \tag{5.8}$$

If $k \leq K-1$ and $0 < t \leq \frac{\sqrt{2}\mu_{\mathcal{D}} r_{K-1}}{LR_0}$, then we can write

$$\frac{1}{r_{k+1}} - \frac{1}{r_k} \geq \frac{\mu_{\mathcal{D}}^2}{4LR_0^2} = a,$$

since $r_k \leq r_{K-1}$. Finally, we have $\frac{1}{r_k} \geq \frac{1}{r_0} + ka$ and hence $r_k \leq \frac{1}{\frac{1}{r_0}+ka}$ for all $k \leq K$. Thus, if $K \geq \frac{1}{a}(\frac{1}{\varepsilon} - \frac{1}{r_0})$, then $r_K \leq \frac{1}{\frac{1}{r_0}+Ka} \leq \varepsilon$. ∎

Notice that Assumption 5.4 can be relaxed in the following way: one can assume that $\|s\|_2 \leq 1$ almost surely for $s \sim \mathcal{D}$, which is closely related to the assumptions used in section 3.3.1 from [13].

Actually, the requirement $t \leq \frac{\sqrt{2}\mu_{\mathcal{D}}\mathbf{E}[f(x_{K-1}) - f_*]}{LR_0}$ could be replaced by $t \leq \frac{\sqrt{2}\mu_{\mathcal{D}}\varepsilon}{LR_0}$ if we additionally enforce that for all $k \leq K$ we have $r_k \geq \varepsilon$ since otherwise it means that we get the desired result after a smaller number of iterations. We notice that the valid range of t depends on the parameters $\mu_{\mathcal{D}}$, L, and R_0, where the last parameter depends on the solution x_*. However, in practice, one can substitute L and R_0 by

some upper bounds $\hat{L}$ and $\hat{R}_0$ for them, while $\mu_{\mathcal{D}}$ can often be computed explicitly (see Lemma 3.4). After that, it remains to specify the desired accuracy ε and choose $t \leq \frac{\sqrt{2}\mu_{\mathcal{D}}\varepsilon}{\hat{L}\hat{R}_0}$.

We want also to remark that in the ideal situation when the method has an access to the exact values of f and machine precision can be made small enough, one can choose t much smaller than the upper bound we provide, and it will help to reach smaller accuracy ε. However, in real world applications it is prohibited to choose t as small as possible either because of presence of some noise in functional values or because of machine precision (see [8] and references therein).

6. Strongly convex problems. In this section we derive the complexity of the STP method in the case of strongly convex f.

Assumption 6.1. f is λ-strongly convex with respect to the norm $\|\cdot\|_{\mathcal{D}}^*$.

Through this section, we denote by x_* the unique minimizer of f.

6.1. Variable stepsize. In our first theorem we prove linear convergence of STP using a variable stepsize.

THEOREM 6.2. *Let Assumptions* 3.2, 3.3, *and* 6.1 *be satisfied. Let stepsize* $\alpha_k = \frac{\theta_k \mu_{\mathcal{D}}}{L}\sqrt{2\lambda(f(x_k)-f(x_*))}$ *for some* $\theta_k \in (0,2)$ *such that* $\theta \overset{def}{=} \inf_k 2\theta_k - \theta_k^2 > 0$. *Assume that inequality* $\frac{\mu_{\mathcal{D}}^2\theta\lambda}{L} \leq 1$ *holds. If*

$$K \geq \frac{L}{\lambda\mu_{\mathcal{D}}^2\theta}\log\left(\frac{f(x_0)-f(x_*)}{\varepsilon}\right), \tag{6.1}$$

then $\mathbf{E}[f(x_K)] - f(x_*) \leq \varepsilon$.

Proof. By injecting α_k into (3.7) of Lemma 3.5 and then subtracting $f(x_*)$ from both sides, we get

$$\mathbf{E}[f(x_{k+1}) \mid x_k] - f(x_*) \leq f(x_k) - f(x_*) - \frac{\mu_{\mathcal{D}}^2\theta_k\sqrt{2\lambda(f(x_k)-f(x_*))}\|\nabla f(x_k)\|_{\mathcal{D}}}{L} + \frac{\mu_{\mathcal{D}}^2\theta_k^2\lambda(f(x_k)-f(x_*))}{L}.$$

From strong convexity of f we have $\|\nabla f(x_k)\|_{\mathcal{D}}^2 \geq 2\lambda(f(x_k)-f(x_*))$; therefore,

$$\begin{aligned}
&\mathbf{E}[f(x_{k+1})|x_k] - f(x_*) \\
&\quad\leq f(x_k) - f(x_*) - \frac{2\mu_{\mathcal{D}}^2\theta_k\lambda(f(x_k)-f(x_*))}{L} + \frac{\mu_{\mathcal{D}}^2\theta_k^2\lambda(f(x_k)-f(x_*))}{L} \\
&\quad\leq f(x_k) - f(x_*) - \frac{\mu_{\mathcal{D}}^2\lambda(f(x_k)-f(x_*))}{L}(2\theta_k - \theta_k^2) \\
&\quad\leq f(x_k) - f(x_*) - \frac{\mu_{\mathcal{D}}^2\theta\lambda(f(x_k)-f(x_*))}{L},
\end{aligned}$$

where we used the definition of θ. Let $r_k = \mathbf{E}\left[f(x_k)\right] - f(x_*)$. By taking the expectation of the last inequality, we get $r_{k+1} \leq (1 - \frac{\mu_{\mathcal{D}}^2\theta\lambda}{L})r_k$, and therefore

$$r_k \leq \left(1 - \frac{\mu_{\mathcal{D}}^2\theta\lambda}{L}\right)^k r_0.$$

Hence if K satisfies (6.1), we get $r_K \leq \varepsilon$. ∎

From this theorem we conclude that if there exist $0 < \theta_1 \le \theta_2 < 2$ such that

$$\frac{\theta_1 \mu_{\mathcal{D}}}{L}\sqrt{2\lambda(f(x_k) - f(x_*))} \le \alpha_k \le \frac{\theta_2 \mu_{\mathcal{D}}}{L}\sqrt{2\lambda(f(x_k) - f(x_*))},$$

then the sequence $(r_k)_k$ converges linearly to zero. We notice also that the condition $\frac{\mu_{\mathcal{D}}^2 \theta \lambda}{L} \le 1$ holds in most typical cases. Indeed, inequalities $\theta \le 1$ and $L \ge \lambda$ always hold, and $\mu_{\mathcal{D}}$ is typically no greater than 1 (see Lemma 3.4).

6.2. Practical stepsize. The stepsizes from the previous theorem depend on $f(x_*)$. In practice, we cannot always use these stepsizes as we usually do not know $f(x_*)$. Our next theorem gives the similar result for `STP` with stepsizes independent from $f(x_*)$ under the additional assumption that for all $s \sim \mathcal{D}$ we have $\|s\|_2 = 1$ with probability 1.

THEOREM 6.3 (practical stepsize). *Let Assumptions* 3.2, 3.3, 5.4, *and* 6.1 *be satisfied. Let* $\alpha_k = \frac{|f(x_k + ts_k) - f(x_k)|}{Lt}$ *for* $0 < t \le \frac{2\mu_{\mathcal{D}}\sqrt{\lambda\varepsilon}}{L}$. *Assume that inequality* $\frac{\mu_{\mathcal{D}}^2 \lambda}{L} \le 1$ *holds. If*

$$K \ge \frac{L}{\lambda \mu_{\mathcal{D}}^2} \log\left(\frac{2(f(x_0) - f(x_*))}{\varepsilon}\right), \tag{6.2}$$

then $\mathbf{E}[f(x_K)] - f(x_*) \le \varepsilon$.

Proof. From (5.5) we have $f(x_{k+1}) \le f(x_k) - \frac{|\langle \nabla f(x_k), s_k\rangle|^2}{2L} + \frac{Lt^2}{8}$. Taking mathematical expectation w.r.t. all randomness from the previous inequality, we get

$$\begin{aligned} \underbrace{\mathbf{E}[f(x_{k+1})] - f_*}_{r_{k+1}} &\overset{①}{\le} \underbrace{\mathbf{E}[f(x_k)] - f_*}_{r_k} - \frac{\mu_{\mathcal{D}}^2}{2L}\mathbf{E}[\|\nabla f(x_k)\|_{\mathcal{D}}^2] + \frac{Lt^2}{8} \\ &\overset{②}{\le} \left(1 - \frac{\mu_{\mathcal{D}}^2 \lambda}{L}\right) r_k + \frac{Lt^2}{8}, \end{aligned} \tag{6.3}$$

where ① is due to the tower property of mathematical expectation and is detailed in (5.7); ② follows from λ-strong convexity of f: $\|\nabla f(x_k)\|_{\mathcal{D}}^2 \ge 2\lambda(f(x_k) - f_*)$. From (6.3) we have

$$\begin{aligned} r_{k+1} &\le \left(1 - \frac{\mu_{\mathcal{D}}^2 \lambda}{L}\right)^{k+1} r_0 + \frac{Lt^2}{8}\sum_{i=0}^{k}\left(1 - \frac{\mu_{\mathcal{D}}^2 \lambda}{L}\right)^i \\ &\le \left(1 - \frac{\mu_{\mathcal{D}}^2 \lambda}{L}\right)^{k+1} r_0 + \frac{L^2 t^2}{8\mu_{\mathcal{D}}^2 \lambda}. \end{aligned} \tag{6.4}$$

Hence, if $t \le \frac{2\mu_{\mathcal{D}}\sqrt{\lambda\varepsilon}}{L}$ and K satisfies (6.2), we get $r_K \le \varepsilon$. ∎

Notice again that the inequality $\frac{\mu_{\mathcal{D}}^2 \lambda}{L} \le 1$ holds in typical cases, since L is always no smaller than λ and $\mu_{\mathcal{D}}$ is typically not greater than 1 (see Lemma 3.4). Moreover, to choose t properly it is enough to know some lower bound $\hat{\lambda}$ for λ and some $\hat{L}$ upper bound L such that $2\mu_{\mathcal{D}}\sqrt{\hat{\lambda}\varepsilon}/\hat{L}$ is not too small to cause problems with machine precision (see also our discussion of this question in the end of section 5). Then, one can choose $t \le 2\mu_{\mathcal{D}}\sqrt{\hat{\lambda}\varepsilon}/\hat{L}$.

7. Numerical results. In this section, we report the results of some preliminary experiments performed in order to assess the efficiency and the robustness of the proposed algorithms compared to the coordinate search method (this method will

be called DDS) and the algorithm proposed in [18]. In the latter approach, at each iteration k, a random vector s_k following the uniform distribution on the unit sphere is generated; then the next iterate is computed as follows:

$$x_{k+1} = x_k - \alpha_k \frac{f(x_k + \mu_k s_k) - f(x_k)}{\mu_k} s_k, \tag{7.1}$$

where $\mu_k \in (0,1)$ is the finite differences parameter and α_k is the stepsize. This method generates a trial step similar to one of the trial steps in our method ($x_- = x_k - \alpha_k s_k$) when the probability distribution $\mathcal{D}$ is chosen to be the uniform distribution on the unit sphere up to the multiplication of the step by $\frac{f(x_k+\mu_k s_k)-f(x_k)}{\mu_k}$. This method will be called the random gradient-free (RGF) method. In the nonconvex case we will add to the comparison the direct search based on the probabilistic descent method proposed in [9] (this method will be called DSRM).

To compare the performance of the algorithms we use performance profiles proposed by Dolan and Moré [6] over a variety of problems. Given a set of problems $\mathcal{P}$ (of cardinality $|\mathcal{P}|$) and a set of algorithms (solvers) $\mathcal{S}$, the performance profile $\rho_s(\tau)$ of an algorithm s is defined as the fraction of problems where the performance ratio $r_{p,s}$ is at most τ:

$$\rho_s(\tau) \;=\; \frac{1}{|\mathcal{P}|} \text{size}\{p \in \mathcal{P} : r_{p,s} \le \tau\}.$$

The performance ratio $r_{p,s}$ is in turn defined by

$$r_{p,s} \;=\; \frac{t_{p,s}}{\min\{t_{p,s} : s \in \mathcal{S}\}},$$

where $t_{p,s} > 0$ measures the performance of the algorithm s when solving problem p, seen here as the number of function evaluations. Better performance of the algorithm s, relative to the other algorithms on the set of problems, is indicated by higher values of $\rho_s(\tau)$. In particular, efficiency is measured by $\rho_s(1)$ (the fraction of problems for which algorithm s performs the best), and robustness is measured by $\rho_s(\tau)$ for τ sufficiently large (the fraction of problems solved by s). Following what is suggested in [6] for a better visualization, we will plot the performance profiles in a $\log_2$-scale (for which $\tau = 1$ will correspond to $\tau = 0$). All the results presented here are averaged over 10 runs of the algorithms. In fact, in our performance profiles, we used the average over 10 runs of $t_{p,s}$. We did all our experiments in sections 7.1–7.3 using MATLAB, and we use Python for experiments presented in section 7.4.

The distribution $\mathcal{D}$ used here for our random direction generation is the uniform distribution on the unit sphere. We performed other experiments (not reported here) with different choices for distributions $\mathcal{D}$, for instance, the distributions listed in Lemma 3.4. We found similar performance as those reported here. The parameters defining the implemented algorithms are set as follows: The stepsize in DSRM is initialized by $\alpha_0 = 1$; then it is updated dynamically with the iterations by multiplying it by 2 when the step is successful and dividing it by 2 otherwise. The forcing function chosen in the sufficient decrease condition is α^2. For RGF we choose $\mu_k = 10^{-4}$, and $\alpha_k = \frac{1}{4(n+4)}$ where n is the problem dimension. For this method the authors proposed to use the stepsize $\alpha_k = \frac{1}{4L(n+4)}$, where L is the Lipschitz constant of the gradient of the objective function. Since for our test problems we do not know this constant, we ran the RGF method with different values for L, for instance, 0.1, 1, 10, and 100. The

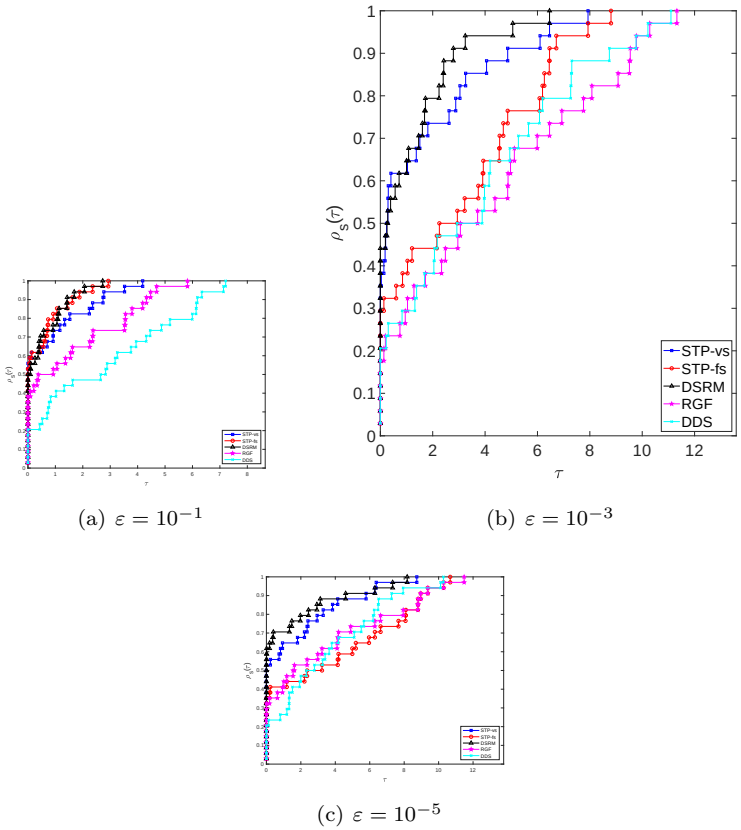
best performance was found for $L = 1$. The stepsize in DDS is initialized by $\alpha_0 = 1$; then it is updated dynamically with the iterations by multiplying it by 2 when the step is successful and dividing it by 2 otherwise.

For all algorithms, we counted the number of function evaluations taken (i) to drive the function value below $f^* + \varepsilon(f(x_0) - f^*)$, where f^* is a local minimal value of the objective function f, and ε is a tolerance (in our experiments $\varepsilon = 10^{-1}$, 10^{-3}, and 10^{-5}) or (ii) when the maximum number of iterations attains $100,000$.

7.1. Nonconvex case. In this section, we report the results of comparison of our approach STP for nonconvex problems with DSRM, DDS, and RGF. We will call our STP method when using the variable stepsize STP-vs, and STP-fs when we use a fixed stepsize. For STP-vs we choose $\alpha_k = \frac{1}{\sqrt{k+1}}$. For STP-fs we choose $\alpha_k = \alpha = 0.1\varepsilon$.

We use the 34 test problems by Moré, Garbow, and Hillstrom [17] which are implemented in MATLAB. All the test problems are smooth. The dimension n of the problems changes between $n = 2$ to $n = 100$, typically $n = 2, 10, 50$, and 100. We use the starting points and the values f^* suggested in [17] for all the problems.

Figure 1 depicts the performance profiles of the algorithms. It shows that our approach (the methods STP-vs and STP-fs) has the same performance (sometimes slightly worse) as that of DSRM. Note that this latter method uses dynamical stepsize, which we believe is a better strategy than a prefixed one. In the future we may extend STP to handle this kind of strategy. We note also that our approach



(a) $\varepsilon = 10^{-1}$ (b) $\varepsilon = 10^{-3}$

(c) $\varepsilon = 10^{-5}$

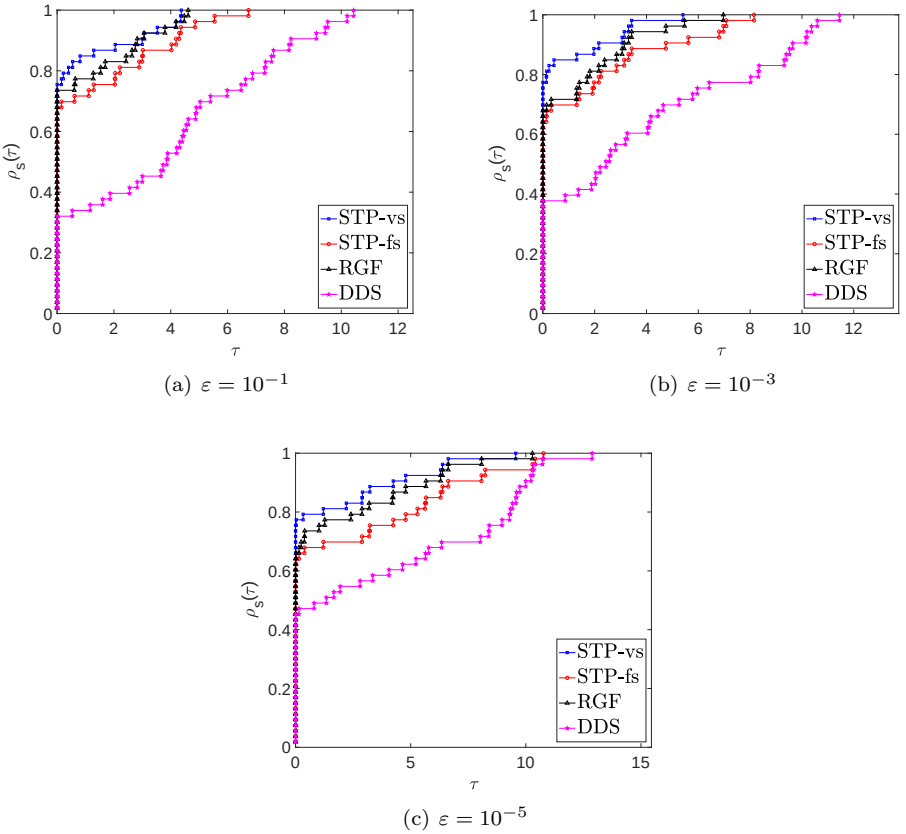
FIG. 1. *Performance profiles on* 34 *optimization problems (nonconvex case).*

(and DSRM) improves the efficiency of the DDS and RGF algorithms on the tested problems. In fact, the number of the function evaluation performance profiles shows that the use of the random directions leads to a significant improvement on terms of the efficiency (for $\tau = 0$, on about 40% of the tested problems our approach performs the best, and less than 5% for RGF and DDS). From Figure 1(a) and (b), we see that the use of the random directions leads to a better robustness when a small precision is targeted (i.e., $\varepsilon = 10^{-1}$ and $\varepsilon = 10^{-3}$). However, when a big precision ($\varepsilon = 10^{-5}$) is targeted DDS becomes competitive. In fact, as shown in Figure 1(c), DDS is more robust than the RGF approach and than our method using a fixed stepsize. Our method STP-vs is still more robust than DDS.

7.2. Convex case. In this section, we report the results of comparison of two STP methods for convex problems with DDS and RGF. The first STP method is the one using the variable stepsize $\alpha_k = |\frac{1}{t}(f(x_k + ts_k) - f(x_k))|$, where $t = 10^{-4}$. We will call this method STP-vs. The second STP method is the one using the fixed stepsize $\alpha_k = \alpha = 0.1\varepsilon$. It will be called STP-fs.

We selected from the Moré–Garbow–Hillstrom problems those with a unique minimum. To have a large test bed, we create different instances for problems by varying the problem dimension n when it is possible. Our test bed in this section contains 40 problems.

In Figure 2, the performance profiles show that the random based methods (the RGF method and our two methods STP-fs and STP-vs) outperform by far the DDS



(a) $\varepsilon = 10^{-1}$ (b) $\varepsilon = 10^{-3}$

(c) $\varepsilon = 10^{-5}$

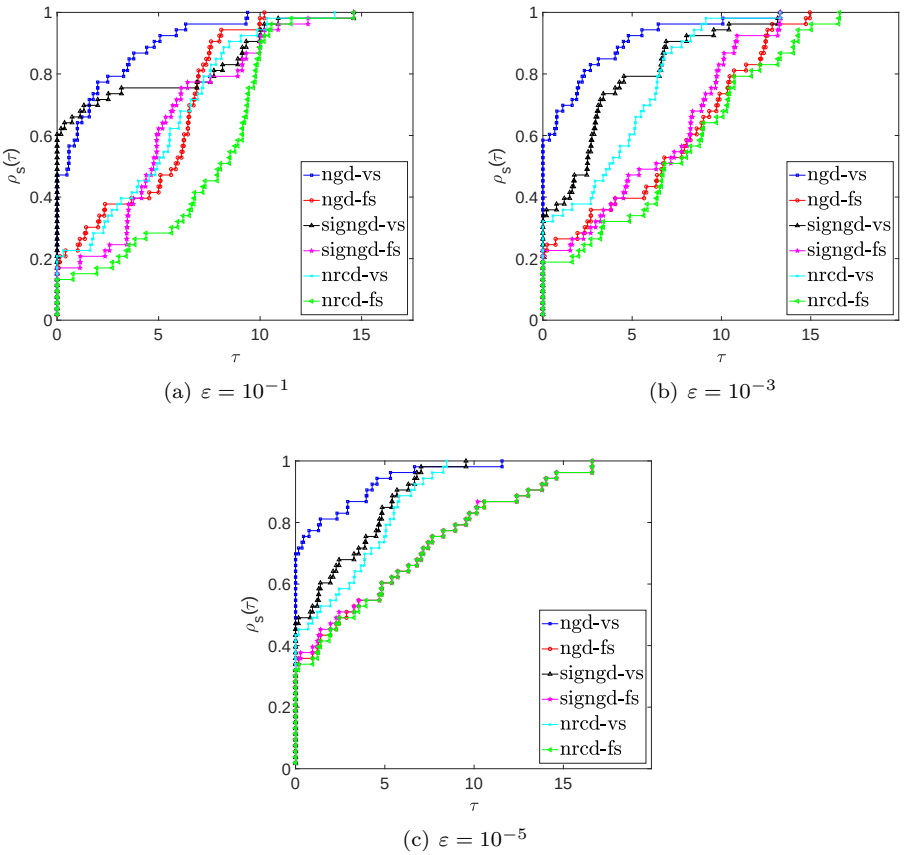
FIG. 2. *Performance profiles on* 40 *optimization problems (convex case).*

method. Our method `STP-vs` gives the best performances for small precision (see Figure 2(a) and (b)). For big precision ($\varepsilon = 1e-5$), it gives performances almost similar to the `RGF` method ((see Figure 2(c)). Our method `STP-fs` is outperformed by `RGF`.

7.3. First order methods. In this section, we report the results of comparison of gradient based methods that our approach covers, using the variable stepsize $\alpha_k = \frac{1}{\sqrt{k+1}}$ and the fixed stepsize $\alpha_k = 0.1\varepsilon$. In fact, to select these stepsizes, we ran many experiments with different values and found the best results for the chosen stepsizes. We denote with `ngd-vs` and `ngd-fs` the `NGD` methods using the variable stepsize and the fixed stepsize, respectively. With similar notation we denote by `signgd-vs` and `signgd-fs` the `SignGD` methods and by `nrcd-vs` and `nrcd-fs` `NRCD` methods using the variable stepsize and the fixed stepsize, respectively.

We use the 34 Moré–Garbow–Hillstrom test problems to which we add 20 problems by creating different instances for problems by varying the problem dimension n when it is possible. Our test bed in this section contains 54 problems.

Figure 3 depicts the performance profiles of the algorithms. It shows that the use of the variable stepsize gives better performances than the fixed stepsize. As one may expect, the NGD method `ngd-vs` exhibits performances better than the other methods, except for small precision ($\varepsilon = 1e-1$); it is less efficient than the sign



(a) $\varepsilon = 10^{-1}$ (b) $\varepsilon = 10^{-3}$

(c) $\varepsilon = 10^{-5}$

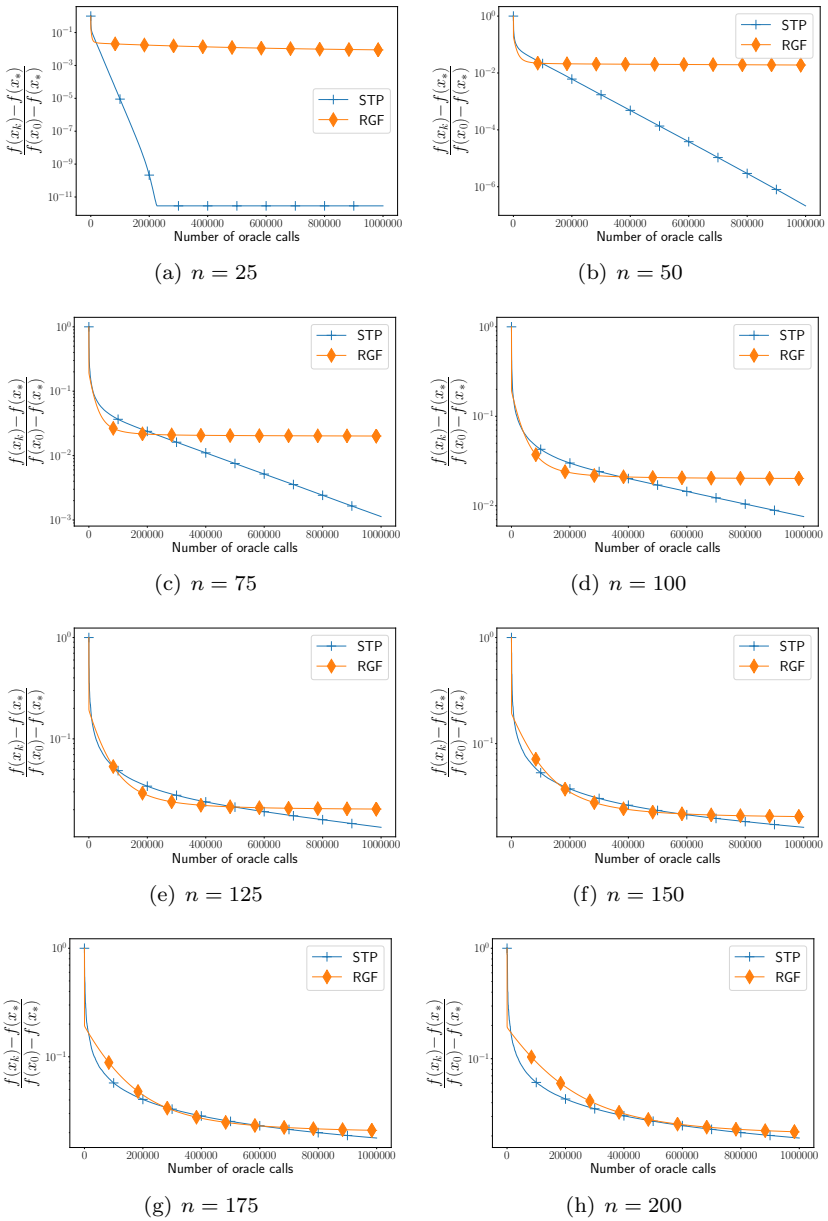
FIG. 3. *Performance profiles on* 54 *optimization problems.*

GD method `signgd-vs`. The latter method is more efficient and less robust than the NRCD method `nrcd-vs`.

7.4. `STP` vs. `RGF`. We considered the following function:

$$f(x) = \frac{1}{2}x_1^2 + \frac{1}{2}\sum_{i=1}^{n-1}(x_{i+1} - x_i)^2 + \frac{1}{2}x_n^2 - x_1 \tag{7.2}$$

which is convex and L-smooth with $L = 4$ and ran `STP` and `RGF` for different n (see Figure 4). For `STP` we use uniform distribution on basis vectors $\{e_1, \ldots, e_n\}$ as



(a) $n = 25$ (b) $n = 50$

(c) $n = 75$ (d) $n = 100$

(e) $n = 125$ (f) $n = 150$

(g) $n = 175$ (h) $n = 200$

FIG. 4. *Trajectories of `STP` with $\mathcal{D}$ being uniform distribution on unit sphere in $\mathbb{R}^n$ and `RGF` applied to minimize the function defined in* (7.2) *for different n.*

distribution $\mathcal{D}$; stepsizes α_k are as in Theorem 5.5 with $t = 10^{-6}$. We see that for $n \leq 50$, STP reaches good enough accuracy $\sim 10^{-6}$ faster than RGF. For bigger dimensions considered, the methods failed to reach good accuracy after a given number of iterations.

8. Conclusions. In this paper, we have proposed a very simple randomized algorithm—the STP method—for DFO. At each iteration, the proposed method tries to decrease the objective function along a random direction sampled from a certain fixed probability law. Under mild assumptions on this law, we have given the properties of this method for nonconvex, convex, and strongly convex problems. In fact, we have derived different practical rules for the stepsizes for which this method converges in expectation to a stationary point of the considered problem.

We have derived the worst case complexity of STP. In fact, in the nonconvex case, we have shown that STP needs $O(n\varepsilon^{-2})$ function evaluations to find a point at which the ℓ_2 norm of the objective function gradient is below ε in expectation. In the convex case, the number of iterations needed to find a point such that the distance between the objective function and its optimal value is below ε in expectation is $O(n\varepsilon^{-1})$. STP is shown to converge linearly for the strongly convex problems, i.e., the complexity is $O(n\log(1/\varepsilon))$. The complexity of STP depends linearly on the dimension of the considered problem, while this dependence is quadratic for DDS methods.

Our numerical experiments showed encouraging performance of the proposed STP algorithm. A number of issues need further investigation, in particular the best choice of probability law for choosing the random directions, a potential parallel version of our method. Extending our results to the nonsmooth problems and/or the constrained problems remains also an interesting topic for the future research. It would also be interesting to confirm the potential of the proposed STP approach compared to the classical approaches in DFO using extensive numerical tests.

Appendix A. Proof of Lemma 3.4.

1. Let $A_n(1) = 2\pi^{\frac{n}{2}}/\Gamma(\frac{n}{2})$ be the surface area of the $n-1$ dimensional unit sphere, where Γ is the gamma function. Then

$$\gamma_{\mathcal{D}} = \mathbf{E}\|s\|_2^2 = \frac{1}{A_n(1)}\int_{\|s\|_2^2=1}\|s\|_2^2 ds = \frac{1}{A_n(1)}\int_{\|s\|_2^2=1} ds = 1.$$

Let $\varepsilon_1 = g/\|g\|_2$, and let $\varepsilon_2, \ldots, \varepsilon_n$ complete ε_1 to an orthonormal basis of $\mathbb{R}^n$. Then

$$\begin{aligned}\mathbf{E}|\langle g, s\rangle| &= \frac{1}{A_n(1)}\int_{\|s\|_2^2=1}|\langle g, s\rangle| ds = \|g\|_2\frac{1}{A_n(1)}\int_{\sum_{i=2}^n s_i^2 = 1-s_1^2}|s_1| ds\\ &= \|g\|_2\frac{1}{A_n(1)}\int_{-1}^{1}|s_1|\int_{\sum_{i=2}^n s_i^2=1-s_1^2} ds_{2:n} ds_1\\ &= \|g\|_2\frac{1}{A_n(1)}\int_{-1}^{1}|s_1| A_{n-1}\left(1-s_1^2\right) ds_1,\end{aligned}$$

where $A_{n-1}\left(1-s_1^2\right) = \frac{2\pi^{(n-1)/2}\left(1-s_1^2\right)^{n-2}}{\Gamma((n-1)/2)}$ is the volume of the $n-2$ dimensional sphere of radius $1-s_1^2$. Hence,

$$\begin{aligned}\mathbf{E}|\langle g, s\rangle| &= \|g\|_2\frac{1}{A_n(1)}\frac{2\pi^{(n-1)/2}}{\Gamma((n-1)/2)}\int_{-1}^{1}|s_1|\left(1-s_1^2\right)^{n-2} ds_1\\ &= \|g\|_2\frac{1}{A_n(1)}\frac{2\pi^{(n-1)/2}}{\Gamma((n-1)/2)(n-1)}.\end{aligned}$$

If $n-1=2p$, then according to the Stirling formula, $p! \sim p^p e^{-p}\sqrt{2\pi p}$, and hence

$$\mathbf{E}|\langle g,s\rangle| = \|g\|_2 \frac{2\pi^p \Gamma(p+1/2)}{2p\Gamma(p)2\pi^p\sqrt{\pi}} = \|g\|_2 \frac{(2p)!}{2^{2p+1}(p!)^2} \quad \sim \quad \frac{\|g\|_2}{2\sqrt{\pi p}}.$$

If $n-1=2p+1$, then

$$\mathbf{E}|\langle g,s\rangle| = \|g\|_2 \frac{2\pi^p\sqrt{\pi}\Gamma(p+1)}{2\pi^{p+1}(2p+1)\Gamma(p+1/2)} = \|g\|_2 \frac{(p!)^2 2^{2p}}{(2p+1)!\pi}$$
$$\sim \|g\|_2 \frac{\sqrt{p}}{\sqrt{\pi}(2p+1)} \sim \frac{\|g\|_2}{2\sqrt{\pi p}}.$$

In both cases, $\mathbf{E}|\langle g,s\rangle| \sim \frac{\|g\|_2}{2\sqrt{\pi p}} \sim \frac{\|g\|_2}{\sqrt{2\pi n}}$.

2. $\gamma_{\mathcal{D}} = \mathbf{E}\|s\|_2^2 = \frac{1}{n}\mathbf{E}\|x\|_2^2 = 1$, where $x \sim N(0,I)$. Note that $s \sim \frac{1}{\sqrt{n}}N(0,I)$ implies $\langle g,s\rangle \sim \frac{1}{\sqrt{n}}N(0,\|g\|_2^2)$, whence

$$\mathbf{E}|\langle g,s\rangle| = \frac{1}{\|g\|_2\sqrt{2n\pi}}\int_{-\infty}^{+\infty}|x|e^{-\frac{x^2}{2\|g\|_2^2}}dx = \frac{\sqrt{2}}{\sqrt{n\pi}}\|g\|_2.$$

3. $\gamma_{\mathcal{D}} = \sum_{i=1}^n \|e_i\|_2^2 P(s=e_i) = 1$ and $\mathbf{E}|\langle g,s\rangle| = \frac{1}{n}\sum_{i=1}^n |g_i| = \frac{1}{n}\|g\|_1$.
4. $\gamma_{\mathcal{D}} = \sum_{i=1}^n \|d_i\|_2^2 P(s=d_i) = \sum_{i=1}^n p_i = 1$ and $\mathbf{E}|\langle g,s\rangle| = \sum_{i=1}^n p_i|g_i d_i| = \|g\|_{\mathcal{D}}$.

Appendix B. Proof that our approach covers some first order methods.

- *NGD method:* At iteration k, $s \sim \mathcal{D}_k$ means that $s = \frac{g_k}{\|g_k\|_2}$ with probability 1.

$$\gamma_{\mathcal{D}_k} = \mathbf{E}_{s\sim\mathcal{D}_k}\|s\|_2^2 = 1,$$

$$\mathbf{E}_{s\sim\mathcal{D}_k}|\langle g_k,s\rangle| = \|g_k\|_2.$$

- *SignGD method:* At iteration k, $s \sim \mathcal{D}_k$ means that $s = sign\,(g_k)$ with probability 1, where the *sign* operation is an elementwise sign.

$$\gamma_{\mathcal{D}_k} = \mathbf{E}_{s\sim\mathcal{D}_k}\|s\|_2^2 = \mathbf{E}_{s\sim\mathcal{D}_k}\|sign\,(g_k)\,\|_2^2 \le \sum_{i=1}^n 1 = n,$$

$$\mathbf{E}_{s\sim\mathcal{D}_k}|\langle g_k,s\rangle| = \mathbf{E}_{s\sim\mathcal{D}_k}|\langle g_k, sign\,(g_k)\rangle| = \|g_k\|_1.$$

- *NRCD method:*[4] At iteration k, $s \sim \mathcal{D}_k$ means that $s = \frac{g_k^i}{|g_k^i|}e_i$ with probability $\frac{1}{n}$, where g_k^i is the ith component of g_k.

$$\gamma_{\mathcal{D}_k} = \mathbf{E}_{s\sim\mathcal{D}_k}\|s\|_2^2 = \frac{1}{n}\sum_{i=1}^n 1 = 1,$$

$$\mathbf{E}_{s\sim\mathcal{D}_k}|\langle g_k,s\rangle| = \mathbf{E}_{i\sim U[1,\dots,n]}\left|\left\langle g_k, \frac{g_k^i}{|g_k^i|}e_i\right\rangle\right| = \frac{1}{n}\sum_{i=1}^n |g_k^i| = \frac{1}{n}\|g_k\|_1.$$

[4] This method can alternatively be called randomized signed gradient descent.

- *NSGD method:* At iteration k, $s \sim \mathcal{D}_k$ means that $s = \hat{g}_k$, where $\hat{g}_k$ is the stochastic gradient satisfying $\mathbf{E}[\hat{g}_k] = \frac{g_k}{\|g_k\|_2}$, and $\mathbf{E}\left[\|\hat{g}_k\|_2^2\right] \leq \sigma < \infty$.

$$\mathbf{E}_{s\sim\mathcal{D}_k} |\langle g_k, s\rangle| = \mathbf{E}_{s\sim\mathcal{D}_k} |\langle g_k, \hat{g}_k\rangle| \geq \mathbf{E}_{s\sim\mathcal{D}_k} \langle g_k, \hat{g}_k\rangle = \|g_k\|_2.$$

REFERENCES

[1] G. Allaire, *Shape Optimization by the Homogenization Method*, Springer Science & Business Media, New York, 2002.

[2] N. Baba, *Convergence of a random optimization method for constrained optimization problems*, J. Optim. Theory Appl., 33 (1981), pp. 451–461.

[3] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*, SIAM, Philadelphia, PA, 2009.

[4] M. A. Diniz-Ehrhardt, J. M. Martínez, and M. Raydan, *A derivative-free nonmonotone line-search technique for unconstrained optimization*, J. Comput. Appl. Math., 219 (2008), pp. 383–397.

[5] M. Dodangeh and L. N. Vicente, *Worst case complexity of direct search under convexity*, Math. Program., 155 (2016), pp. 307–332.

[6] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), pp. 201–213.

[7] C. C. Y. Dorea, *Expected number of steps of a random optimization method*, J. Optim. Theory Appl., 39 (1983), pp. 165–171.

[8] P. Dvurechensky, E. Gorbunov, and A. Gasnikov, *An accelerated method for derivative-free smooth stochastic convex optimization*, European J. Oper. Res., in press.

[9] S. Gratton, C. W. Royer, L. N. Vicente, and Z. Zhang, *Direct search based on probabilistic descent*, SIAM J. Optim., 25 (2015), pp. 1515–1541.

[10] J. Haslinger and R. A. E. Mäkinen, *Introduction to Shape Optimization: Theory, Approximation, and Computation*, SIAM, Philadelphia, PA, 2003.

[11] V. G. Karmanov, *Convergence estimates for iterative minimization methods*, Comput. Math. Math. Phys., 14 (1974), pp. 1–13.

[12] V. G. Karmanov, *On convergence of a random search method in convex minimization problems*, Theory Probab. Appl., 19 (1975), pp. 788–794.

[13] T. G. Kolda, R. M. Lewis, and V. Torczon, *Optimization by direct search: New perspectives on some classical and modern methods*, SIAM Rev., 45 (2003), pp. 385–482.

[14] J. Konečný and P. Richtárik, *Simple Complexity Analysis of Simplified Direct Search*, arXiv preprint arXiv:1410.0390, 2014.

[15] J. Matyas, *Random optimization*, Autom. Remote Control, 26 (1965), pp. 246–253.

[16] B. Mohammadi and O. Pironneau, *Applied Shape Optimization for Fluids*, Oxford University Press, Oxford, UK, 2010.

[17] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, *Testing unconstrained optimization software*, ACM Trans. Math. Softw., 7 (1981), pp. 17–41.

[18] Yurii Nesterov and Vladimir Spokoiny, *Random gradient-free minimization of convex functions*, Found. Comput. Math., 17 (2017), pp. 527–566.

[19] B. T. Polyak, *Introduction to Optimization*, Optimization Software, New York, 1987.

[20] M. S. Sarma, *On the convergence of the Baba and Dorea random optimization methods*, J. Optim. Theory Appl., 66 (1990), pp. 337–343.

[21] S. U. Stich, C. L. Müller, and B. Gärtner, *Optimization of convex functions with random pursuit*, SIAM J. Optim., 23 (2013), pp. 1284–1309.

[22] L. N. Vicente, *Worst case complexity of direct search*, EURO J. Comput. Optim., 1 (2013), pp. 143–153.