Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Quantile Optimization via Multiple-Timescale Local Search for Black-Box Functions

Jiaqiao Hu, Meichen Song, Michael C. Fu

To cite this article:
Jiaqiao Hu, Meichen Song, Michael C. Fu (2024) Quantile Optimization via Multiple-Timescale Local Search for Black-Box Functions. Operations Research

Published online in Articles in Advance 12 Mar 2024

. https://doi.org/10.1287/opre.2022.0534

https://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Articles in Advance, pp. 1–23
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Methods

Quantile Optimization via Multiple-Timescale Local Search for Black-Box Functions

Jiaqiao Hu,[a,*] **Meichen Song,**[a] **Michael C. Fu**[b,c]

[a] Department of Applied Mathematics and Statistics, State University of New York at Stony Brook, Stony Brook, New York 11794; [b] Robert H. Smith School of Business, University of Maryland, College Park, Maryland 20742; [c] Institute for Systems Research, University of Maryland, College Park, Maryland 20742

*Corresponding author

Contact: jiaqiao.hu.1@stonybrook.edu, https://orcid.org/0000-0002-9999-672X (JH); meichen.song@stonybrook.edu, https://orcid.org/0000-0001-8054-0602 (MS); mfu@umd.edu, https://orcid.org/0000-0003-2105-4932 (MCF)

Received: October 6, 2022
Revised: June 15, 2023; October 16, 2023
Accepted: January 30, 2024
Published Online in Articles in Advance: March 12, 2024

Area of Review: Simulation

https://doi.org/10.1287/opre.2022.0534

Abstract. We consider quantile optimization of black-box functions that are estimated with noise. We propose two new iterative three-timescale local search algorithms. The first algorithm uses an appropriately modified finite-difference-based gradient estimator that requires $2d + 1$ samples of the black-box function per iteration of the algorithm, where d is the number of decision variables (dimension of the input vector). For higher-dimensional problems, this algorithm may not be practical if the black-box function estimates are expensive. The second algorithm employs a simultaneous-perturbation-based gradient estimator that uses only three samples for each iteration regardless of problem dimension. Under appropriate conditions, we show the almost sure convergence of both algorithms. In addition, for the class of strongly convex functions, we further establish their (finite-time) convergence rate through a novel fixed-point argument. Simulation experiments indicate that the algorithms work well on a variety of test problems and compare well with recently proposed alternative methods.

Funding: This work was supported by the Air Force Office of Scientific Research [Grant FA95502010211] and the National Science Foundation [Grants CMMI-2027527, IIS-2123684].
Supplemental Material: The online appendix is available at https://doi.org/10.1287/opre.2022.0534.

Keywords: black-box optimization • quantile • local search • stochastic approximation • finite differences • simultaneous perturbation

1. Introduction

In black-box settings, only estimates of an output function are available; that is, there is minimal knowledge of the underlying system generating the output. Furthermore, the output estimates might also contain observation noise. For such problems, there is an extensive literature of algorithms addressing the case where the performance measure is an expectation, most commonly the mean; for example, see Fu (2015) and references therein in the context of simulation optimization. However, in many situations such as many risk management problems, one is interested in tail behavior of the output function or the median rather than the mean, in which case the performance measure of interest is a quantile, and the objective is quantile optimization.

Black-box optimization (BBO), defined by Audet and Hare (2017, p. 6) as "the study of design and analysis of algorithms that assume the objective and/or constraint functions are given by blackboxes" is a well-developed field in the deterministic (noiseless) setting. Although the focus of the algorithms described and analyzed in their book is derivative-free approaches, Audet and Hare (2017, p. 6) begin by strongly recommending that "if gradient information is available, reliable, and obtainable at reasonable cost, then gradient-based methods should be used."

In this paper, we consider the stochastic BBO setting where the goal is to optimize the quantile of a black-box output random variable. Our main assumption is that the quantile function is smooth enough so that gradient-based search will yield locally optimal solutions. Such a smoothness condition is guaranteed when the output distribution is differentiable (see Equation (4)), which is common in many engineering applications, ranging from queueing network optimization (e.g., Fu and Hill 1997) to traffic simulation (Spall and Chin 1997, Li et al. 2017) to neural network (NN) training (e.g., Spall and Cristion 1997, Hong et al. 2010). For instance, the steady-state waiting time distribution in a queueing network is usually differentiable with respect to the service rates of the nodes, whereas in traffic simulation, the distribution of vehicle travel time on a road

network is typically a smooth function of traffic signal timings (e.g., Cao et al. 2014). Nevertheless, because of the complexity of such problems and/or the lack of model details (e.g., when a simulation program is coded using an off-the-shelf commercial package, or knowledge transfer from the builder to the user of a model is not properly assured), directly estimating gradients based on exploiting model structure is sometimes difficult or practically infeasible. Moreover, in certain applications such as NN controller design for systems with unknown dynamics (Spall and Chin 1997, Spall and Cristion 1997), it is not even possible to determine the gradient of a loss function through direct gradient techniques. In these circumstances, the underlying system model is essentially treated as a black box, and finite-difference (FD)-based estimates of the gradient are often used to carry out local search, dating back to the Kiefer-Wolfowitz stochastic approximation (SA) algorithm (Kiefer and Wolfowitz 1952, Kushner and Clark 1978, Kushner and Yin 1997). However, because the usual quantile estimator itself is not unbiased, constructing an effective/efficient FD estimator of the quantile gradient requires far more care than the straightforward mean-based case. To our knowledge, there are very few algorithms in general for what could be called quantile BBO, in contrast to the abundance of local search algorithms for mean performance.

We will be relying on (approximate) quantile gradient estimates based only on output function samples. To be specific, let $Y(\theta)$ denote the output random variable and $\theta \in \Theta \subseteq \Re^d$ the set of input decision variables, which we will refer to henceforth as the (input) parameter vector, which can include both distributional and structural parameters, meaning that the elements of θ may affect the black-box function both directly and via the input distributions. The usual optimization problem is of the form $\min_{\theta\in\Theta} E[Y(\theta)]$, whereas we consider the optimization problem

$$\min_{\theta\in\Theta} q_\varphi(\theta), \tag{1}$$

where the quantile function $q_\varphi(\theta)$ is defined by

$$P(Y(\theta) \le q_\varphi(\theta)) = \varphi, \quad \varphi \in (0,1).$$

Because φ is fixed throughout this paper, its dependence will be dropped henceforth to simplify notation; that is, the quantile will simply be denoted by $q(\theta)$ or sometimes just q.

Under appropriate smoothness conditions on $q(\theta)$, solving (1) essentially becomes equivalent to finding the zero of the gradient $\nabla q(\theta)$, so a gradient-based iterative local search algorithm would take the general form

$$\theta_{k+1} = \theta_k - \alpha_k \hat{\nabla}\, q(\theta_k), \tag{2}$$

where $\hat{\nabla} q(\cdot)$ denotes an estimator of the quantile gradient, which is the key element in defining the algorithm. A straightforward symmetric finite-difference (SD) estimator for the quantile gradient would take the following form:

$$\hat{\nabla}_i\, q(\theta) = \frac{\hat{q}(\theta + ce_i) - \hat{q}(\theta - ce_i)}{2c}, \; i = 1, \ldots, d, \tag{3}$$

where $\hat{\nabla}_i$ denotes the ith component of the gradient estimator, e_i denotes the unit vector in the ith direction, and $c > 0$. This is essentially the approach taken by Kibzun and Matveev (2012). Note that one could also consider one-sided (forward or backward) FD estimators. However, one challenge that is apparent in the estimator (3) is that unlike in the mean case, the two difference terms in the numerator of the quantile finite-difference gradient are not themselves unbiased, only consistent, which means that the iteration sample size would eventually have to increase to infinity to guarantee convergence. Furthermore, each iteration would require calculation of $\hat{q}$ using order statistics, which may be computationally impractical. Our alternative approach is to use two additional iterative updates for $q(\theta)$ and $\nabla q(\theta)$ based on the following result.

Assuming that the output function $Y(\theta)$ is a continuous random variable with cumulative distribution function (c.d.f.) F and probability density function (p.d.f.) f, we make use of the following relationship (Fu et al. 2009):

$$\nabla_\theta q(\theta) = -\frac{\nabla_\theta F(y;\theta)|_{y=q}}{f(q;\theta)}. \tag{4}$$

The simplest way to use this would be to solve for the zero of the numerator and use the SD estimator analogous to (3) for the c.d.f. gradient (e.g., Song et al. 2023):

$$\hat{\nabla}_i F(q;\theta) = \frac{I\{Y(\theta + ce_i) \le q\} - I\{Y(\theta - ce_i) \le q\}}{2c}, \quad i = 1, \ldots, d, \tag{5}$$

where $I\{\cdot\}$ denotes the indicator function, in which case (2) becomes a two-timescale SA algorithm:

$$\theta_{k+1} = \theta_k + \alpha_k \hat{\nabla}_\theta F(q_k;\theta_k), \tag{6}$$

$$q_{k+1} = q_k + \gamma_k(\varphi - I\{Y(\theta_k) \le q_k\}), \tag{7}$$

where (7) is a recursive quantile estimator replacing the classical sample quantile based on order statistics (see Section 2), and to make the algorithmic convergent, the perturbation c in (5) would also need to go to zero as $k \to \infty$. However, empirical results indicate that this approach may not work well in practice, especially when the quantile level φ is close to one (or zero), in which case the two indicator terms in (5) will simultaneously take the value one (or zero) with high probability. As a result, a large number of iterations need to be

performed in order to obtain a meaningful (nonzero) estimate of the c.d.f. gradient. Moreover, because (5) provides an estimate for only the direction of the quantile gradient, not its magnitude, the approach may not be useful in other related applications such as robustness assessment and quantile sensitivity analysis.

The algorithms we propose in this work follow the general structure of (6) and (7); however, our goal is to estimate the "real" quantile gradient (4) rather than just $\nabla_\theta F(y;\theta)|_{y=q}$ in the numerator. As in (5), each step of our proposed SD-based estimator requires $2d+1$ function evaluations. When the number of decision variables is large, the SD estimator may become computationally impractical, so we introduce a second algorithm based on using a simultaneous-perturbation (SP) gradient estimator along the lines of Spall (1992), which uses only three function evaluations at each iteration, independent of input parameter dimension. Both the SD and SP estimators require significant adjustments to handle the quantile setting.

As alluded to earlier, the literature on quantile optimization in the stochastic BBO setting is very sparse, and we now review the most closely related work. The work most closely related to ours is Kibzun and Matveev (2012), cited earlier, which proposes a stochastic quasi-gradient (QG) algorithm for convex quantile objectives by estimating quantile gradients via a "traditional" symmetric difference approximation. Also relevant to the BBO setting are the derivative-free methods using the Bayesian optimization approach, for example, Wang et al. (2021) and Sabater et al. (2021), which employ a surrogate model to approximate the response surface of the unknown quantile function. Lastly, the multitimescale SA procedure developed in Hu et al. (2022) has the same structure proposed in our work, but the algorithm cannot be applied in the BBO setting, because it relies on the availability of direct gradients (e.g., through techniques such as perturbation analysis or the likelihood ratio method) that are not available in a black-box setting, as knowledge of the underlying system is needed to derive the gradient estimators, whereas our algorithms use only the black-box function outputs.

Some other related work, albeit much less relevant to our BBO setting, is algorithms that rely on knowledge of the output distribution. These include the mathematical programming approaches presented in Kibzun and Kurbakovskiy (1991), Kibzun et al. (2013), and Vasiléva and Kan (2015), and the scenario optimization method of Zamar et al. (2017). For differentiable problems, there are also approaches that use gradient information, such as Kim and Powell (2011), who propose a recursive gradient algorithm for a special class of heavy-tailed distributions that admits the interchange of the derivative and quantile function.

Under appropriate conditions, we analyze the bias and variance of the proposed quantile gradient estimators and establish the almost sure local convergence of the two FD-based algorithms—SD quantile optimization (SDQO) and SP quantile optimization (SPQO)—for general multimodal problems. Most importantly, for the class of problems with strongly convex objective functions, we are able to analyze the (finite-time) convergence *rate* of the algorithms by introducing a novel fixed-point argument. The key idea is to bound the algorithm's estimation errors through the composition of a sequence of suitably constructed contraction mappings, so that the convergence rates of quantile/gradient estimates can be characterized in detail by inspecting the solutions to a collection of fixed-point equations. As far as we are aware, these are the first quantile BBO algorithms with both guaranteed convergence and a known rate of convergence. Although the convergence rate of single-timescale SA is well understood in the literature (cf., e.g., Fabian 1968, Spall 1992, Kushner and Yin 1997, Borkar 2008), the rate analysis for multitimescale SA algorithms has been a long-standing open research challenge. The only existing results seem to be Konda and Tsitsiklis (2004) and Mokkadem and Pelletier (2006) for two-timescale SA algorithms. Our algorithms operate on three timescales, and the convergence rate study of such SA algorithms has not been addressed. Moreover, the fixed-point argument presented in this work is by no means limited to the analysis of these algorithms, but provides a new general approach that can potentially be applied to address the convergence rate issues of other multitimescale SA algorithms.

In sum, we view our work as making the following research contributions:

- We introduce new FD-based local search algorithms, SDQO and SPQO, for optimizing a black-box quantile function, prove their convergence, and characterize their convergence rate.
- In terms of theory, the convergence rate analysis is the first such result for a three-timescale SA algorithm, and the fixed-point argument used in the analysis is a new general approach that can be applied to other multitimescale SA algorithms.
- In terms of practice, SPQO is particularly well suited to high-dimensional problems, because the number of black-box evaluations per iteration is independent of the number of decision variables (optimization input parameters).
- Lastly, the new algorithms provide a practical complement to existing global optimization algorithms that primarily use metamodeling/surrogate functions for BBO.

The rest of this paper is organized as follows. Section 2 begins with an intuitive motivation for the two FD-based black-box quantile gradient estimators and then presents the SD/SP estimators, along with their corresponding optimization algorithms SDPO and SPQO, with a detailed discussion of the proposed simultaneous-

perturbation estimator. The convergence and convergence rate analyses of the algorithms are provided in Sections 3 and 4, respectively. In Section 5, simulation experiments are used to illustrate and test the performance of the algorithms, and Section 6 provides some conclusions and future research.

2. New FD-Based Quantile BBO Algorithms

We begin with an intuitive informal derivation of the general form of an FD gradient estimator, which will then be specialized to the SD and SP gradient estimators, to be analyzed more rigorously. To simplify the discussion (and notation) here, we consider the case where θ is a scalar ($d=1$), so we seek to estimate the derivative $q'(\theta)$, in which (4) can be viewed as the ratio of two derivatives:

$$q'(\theta) = -\frac{\partial_2 F(q;\theta)}{\partial_1 F(q;\theta)}, \text{ defining } \partial_i F(q;\theta) \equiv \partial_i F(x;\theta)|_{x=q},$$

where ∂_i denotes the derivative w.r.t. the ith argument and the latter definition is for notational convenience. When enough is known about the system to develop direct derivative estimators for ∂_1 and ∂_2, that is, the setting considered in Hu et al. (2022), a natural approach to estimate $q'(\theta)$ would be to solve the equivalent root-finding problem:

$$q'(\theta)\partial_1 F(q;\theta) + \partial_2 F(q;\theta) = 0. \tag{8}$$

Assuming that direct derivative estimators are not available, it turns out that a straightforward extension to using FD estimates of each of these derivatives not only would be computationally burdensome but could also lead to numerical difficulties. In particular, as mentioned in Section 1, an FD estimator such as (5) for either ∂_1 or ∂_2 would frequently yield a value of zero, in which case the root-finding Equation (8) is not even well posed. Therefore, we instead motivate an alternative estimator for approximating the entire left-hand side of (8) by considering a simple first-order Taylor series expansion of F in the two arguments:

$$\begin{aligned} F^{\pm} &\equiv F(q \pm \Delta q; \theta \pm \Delta\theta) \\ &= F(q;\theta) \pm \partial_1 F(q;\theta)\Delta q \pm \partial_2 F(q;\theta)\Delta\theta, \end{aligned}$$

where we are ignoring higher-order terms for now, but these arguments will be made more formal shortly. Taking the difference,

$$\begin{aligned} F^+ - F^- &= 2\partial_1 F(q;\theta)\Delta q + 2\partial_2 F(q;\theta)\Delta\theta \\ &= 2\Delta\theta[\partial_1 F(q;\theta)q'(\theta) + \partial_2 F(q;\theta)], \end{aligned}$$

where we have taken $\Delta q = q'(\theta)\Delta\theta$, so solving $(F^+ - F^-)/2\Delta\theta = 0$ is equivalent to the root-finding problem using direct gradients given above by (8). Noting that $F^{\pm} = E[I\{Y^{\pm} \le q + q'(\theta)\Delta\theta\}]$, where $Y^{\pm} \sim F(\cdot;\theta \pm \Delta\theta)$, it thus motivates the two coupled root-finding equations that must be solved:

$$E\left[\frac{-I\{Y^+ \le q + q'(\theta)\Delta\theta\} + I\{Y^- \le q - q'(\theta)\Delta\theta\}}{2\Delta\theta}\right] = 0, \tag{9}$$

$$E[I\{Y(\theta) \le q\}] = \varphi, \tag{10}$$

where the second equation is solved via the SA iteration (7), and the first equation will be incorporated into a new SA iteration to serve as the gradient estimator in the SA iteration (6).

2.1. SD/SP Quantile Gradient Estimators

We now return to the multidimensional ($d>1$) setting and provide two versions of the FD scheme just described. Both versions can be viewed as different implementations of the SA method for numerically solving the two coupled stochastic root-finding Equations (9) and (10).

Denote by θ^* an optimal solution to (1) and let $\hat{\theta}_k$ be an estimate of θ^*. Let $\hat{\theta}_k$ be fixed, and let $\hat{q}_k$ and $\hat{D}_k$ be the current estimates of $q(\hat{\theta}_k)$ and $\nabla_\theta q(\theta)|_{\theta=\hat{\theta}_k}$. The SD estimator we propose simultaneously computes new estimates of the quantile and its gradient as follows:

$$\hat{D}_{k+1} = \hat{D}_k + \frac{\beta_k}{2c_k}\begin{pmatrix} -I\{Y(\hat{\theta}_k + c_k e_1) \le \hat{q}_k + c_k\hat{D}_k^T e_1\} \\ \quad + I\{Y(\hat{\theta}_k - c_k e_1) \le \hat{q}_k - c_k\hat{D}_k^T e_1\} \\ \vdots \\ -I\{Y(\hat{\theta}_k + c_k e_d) \le \hat{q}_k + c_k\hat{D}_k^T e_d\} \\ \quad + I\{Y(\hat{\theta}_k - c_k e_d) \le \hat{q}_k - c_k\hat{D}_k^T e_d\} \end{pmatrix} \tag{11}$$

$$\hat{q}_{k+1} = \hat{q}_k + \gamma_k(\varphi - I\{Y(\hat{\theta}_k) \le \hat{q}_k\}), \tag{12}$$

where β_k, $\gamma_k > 0$ are step-sizes, $c_k > 0$ is the perturbation size, and $Y(\hat{\theta}_k \pm c_k e_i)$ (for $i = 1,\ldots,d$) are output random variables obtained by perturbing the ith element of $\hat{\theta}_k$ while holding all other components unchanged. Clearly, each step of (11) requires $2d$ function evaluations. This, together with $Y(\hat{\theta}_k)$ needed in (12) for quantile estimation, results in a total of $2d+1$ function evaluations per iteration of the procedure.

The SP estimator, on the other hand, simultaneously varies all components of the underlying parameter vector in random directions, so that the same effect of the SD scheme can be achieved with only three function evaluations. Compared with the $2d+1$ per-iteration complexity of the SD estimator, this has the potential to lead to significant savings in computational cost, especially when the problem dimension is high and/or black-box function evaluations are expensive. Let θ_k, q_k,

and D_k denote the respective SP estimates for θ^*, $q(\theta_k)$, and $\nabla_\theta q(\theta)|_{\theta=\theta_k}$. The estimator can be compactly expressed in the following recursive form:

$$D_{k+1} = D_k + \beta_k \left[\frac{\begin{array}{c} -I\{Y(\theta_k + c_k\Delta_k) \le q_k + c_k D_k^T \Delta_k\} \\ +I\{Y(\theta_k - c_k\Delta_k) \le q_k - c_k D_k^T \Delta_k\} \end{array}}{2c_k\Delta_k} \right] \tag{13}$$

$$q_{k+1} = q_k + \gamma_k(\varphi - I\{Y(\theta_k) \le q_k\}), \tag{14}$$

where $\Delta_k = (\Delta_{k,1}, \ldots, \Delta_{k,d})^T$ is a zero-mean random direction with i.i.d. components, and the division by the vector Δ_k is understood to be element-wise.

The above estimators depart significantly from the usual SD/SP formulations in several different aspects: (i) unlike (3), they involve the difference quotient of an indicator function rather than that of the quantile function whose gradient is sought; (ii) both are iterated, rather than one-shot (as in (3) and (5)), procedures in which the gradient estimation is coupled with another iterative process for estimating the quantile; (iii) in contrast to conventional SD/SP, where only the parameter vector θ is varied (see, e.g., (3) and (5)), the quantile estimate $\hat{q}_k$ (respectively (resp.), q_k) is randomly perturbed in (11) (resp. (13)) at the same time, with the magnitude of the perturbation being directly affected by the gradient estimate $\hat{D}_k$ (resp. D_k) itself. As we will see shortly, this last difference further leads to other differences in algorithm design and analysis.

We now provide additional validation for why these estimators work, formalizing the intuitive derivation outlined in the beginning of the section. We focus on the SP estimator and consider (13) and (14) in their deterministic forms. The SD estimator works in a completely analogous way, so most of the arguments for the SP case also carry over to the SD estimator. Note that conditional on θ_k, q_k, D_k, and Δ_k, the expectations of the two indicator terms in (13) are given by $F(q_k + c_k D_k^T \Delta_k; \theta_k + c_k\Delta_k)$ and $F(q_k - c_k D_k^T \Delta_k; \theta_k - c_k\Delta_k)$. A two-variable third-order Taylor series expansion of these two functions around (q_k, θ_k) then yields

$$\begin{aligned} &\frac{-F(q_k + c_k D_k^T \Delta_k; \theta_k + c_k\Delta_k) + F(q_k - c_k D_k^T \Delta_k; \theta_k - c_k\Delta_k)}{2c_k\Delta_k} \\ &= \frac{-2f(q_k, \theta_k) c_k \Delta_k^T D_k - 2\nabla_\theta^T F(q_k; \theta)|_{\theta=\theta_k} c_k\Delta_k}{2c_k\Delta_k} + O(c_k^2), \end{aligned} \tag{15}$$

where the big-O notation signifies the order of a term, which is formally defined in Section 3. Thus, by applying the key argument of SP theory (i.e., $E[\Delta_{k,i}/\Delta_{k,j}] = 0$ for all $i \ne j$) and ignoring the higher-order bias term $O(c_k^2)$ in (15), it is not difficult to observe that the expected-value version of (13) (with the difference quotient there replaced by its conditional expectation given θ_k, q_k, and D_k, but excluding Δ_k) can be written as

$$D_{k+1} = D_k + \beta_k(-f(q_k; \theta_k)D_k - \nabla_\theta F(q_k; \theta)|_{\theta=\theta_k}). \tag{16}$$

Equation (16) is a fixed-point iteration for solving $-f(q_k; \theta_k)D - \nabla_\theta F(q_k; \theta)|_{\theta=\theta_k} = 0$ for D, which has solution $-\nabla_\theta F(q_k; \theta)|_{\theta=\theta_k}/f(q_k; \theta_k)$ in exactly the same form as (4) with the true quantile $q(\theta_k)$ replaced by its estimate q_k. A similar interpretation also applies to (14), and by noting that $E[I\{Y(\theta_k) \le q_k\}|\theta_k, q_k] = F(q_k; \theta_k)$, the sequence $\{q_k\}$ can be seen to track the unique solution $q(\theta_k)$ to the root-finding problem $\varphi - F(q; \theta_k) = 0$. Consequently, as q_k tends to $q(\theta_k)$, it is reasonable to expect that D_k will provide a close approximation to the true gradient $\nabla_\theta q(\theta)|_{\theta=\theta_k}$.

The preceding developments ignored the fact that because of the extra perturbations $\pm c_k D_k^T \Delta_k$ introduced in (13), the iterate D_k itself is contained in the higher-order term in (15) (see the proof of Lemma 2); thus, the sequence $\{D_k\}$ could in fact increase in magnitude to negate the claimed $O(c_k^2)$ order of the term. This is a technical issue that does not occur in the usual mean-based setting, where the perturbation size is solely determined by c_k, so that the order of the bias can be bounded uniformly even without explicitly requiring the boundedness of the iterates (cf. proof of lemma 1 in Spall 1992). To address this issue, we instead consider a slight variant of (13) that replaces c_k in the difference quotient by a perturbation size that adapts to the magnitude of D_k. Specifically, let $M_k = \max\{1, \|D_k\|/\sqrt{d}\}$ and define $\bar{c}_k = c_k/M_k$. We suggest the following modification of (13):

$$D_{k+1} = D_k + \beta_k \left[\frac{\begin{array}{c} -I\{Y(\theta_k + \bar{c}_k\Delta_k) \le q_k + \bar{c}_k D_k^T \Delta_k\} \\ +I\{Y(\theta_k - \bar{c}_k\Delta_k) \le q_k - \bar{c}_k D_k^T \Delta_k\} \end{array}}{2\bar{c}_k\Delta_k} \right]. \tag{17}$$

It can be easily seen that (17) serves the same estimation purpose as (13) in the sense that its "mean flow" (i.e., deterministic counterpart), modulo the higher-order error terms, is identical to (16). Nevertheless, because $|\bar{c}_k D_k^T \Delta_k| \le c_k\sqrt{d}\|\Delta_k\|$, the use of $\bar{c}_k$ in (17) prevents the perturbations in q_k from becoming excessively large, and thus reduces the influence of D_k on the resulting estimation bias. We show in Section 3.1 that under reasonable conditions, the sequence $\{D_k\}$ generated by (17) remains bounded, both almost surely and in second-order moments, which in effect justifies the $O(c_k^2)$ bias of the proposed estimator.

The constructions of our SD/SP estimators are based on a symmetric difference scheme. It is possible to

consider alternative estimators relying on one-sided difference that use $d+1$ (resp. two) simulation evaluations per iteration. For example, in the SP case, the difference quotient in (17) could instead be replaced by either

$$\frac{-I\{Y(\theta_k) \le q_k + \overline{c}_k D_k^T \Delta_k\} + I\{Y(\theta_k - \overline{c}_k \Delta_k) \le q_k\}}{\overline{c}_k \Delta_k} \text{ or}$$

$$\frac{-I\{Y(\theta_k + \overline{c}_k \Delta_k) \le q_k + \overline{c}_k D_k^T \Delta_k\} + I\{Y(\theta_k) \le q_k\}}{\overline{c}_k \Delta_k}.$$

Both approaches would lead to the same desired effect for estimating the quantile gradient but come at the cost of introducing large biases of order $O(c_k)$ compared with the $O(c_k^2)$ bias in (17).

2.2. SA Local Search Algorithms

The quantile optimization algorithms we propose integrate $\hat{D}_k$ and D_k into the standard gradient descent method, as presented below, where $\Pi_\Theta(\cdot)$ denotes a projection operator that brings an iterate back onto the parameter space Θ whenever it becomes infeasible.

Algorithm 1 (SDQO)

Input: initial estimates $\hat{q}_0$, $\hat{D}_0$, $\hat{\theta}_0$; sequences $\{\alpha_k\}, \{\beta_k\}, \{\gamma_k\}, \{c_k\}$;

Initialize: iteration counter $k \leftarrow 0$;

Loop until a stopping criterion is met:

$$\tilde{c}_k = c_k / \max\{1, \|\hat{D}_k\|/\sqrt{d}\}, \tag{18}$$

$$\hat{q}_{k+1} = \hat{q}_k + \gamma_k(\varphi - I\{Y(\hat{\theta}_k) \le \hat{q}_k\}),$$

$$\hat{D}_{k+1} = \hat{D}_k + \frac{\beta_k}{2\tilde{c}_k} \begin{pmatrix} -I\{Y(\hat{\theta}_k + \tilde{c}_k e_1) \le \hat{q}_k + \tilde{c}_k D_k^T e_1\} \\ + I\{Y(\hat{\theta}_k - \tilde{c}_k e_1) \le \hat{q}_k - \tilde{c}_k D_k^T e_1\} \\ \vdots \\ -I\{Y(\hat{\theta}_k + \tilde{c}_k e_d) \le \hat{q}_k + \tilde{c}_k D_k^T e_d\} \\ + I\{Y(\hat{\theta}_k - \tilde{c}_k e_d) \le \hat{q}_k - \tilde{c}_k D_k^T e_d\} \end{pmatrix},$$

$$\hat{\theta}_{k+1} = \Pi_\Theta(\hat{\theta}_k - \alpha_k \hat{D}_k),$$

$$k \leftarrow k+1; \tag{19}$$

Note that (19) is a variant of (11), in which the same substitution as in (17) has been made, that is, with $\tilde{c}_k$ defined by (18) replacing c_k in (11).

Algorithm 2 (SPQO)

Input: initial estimates q_0, D_0, θ_0; sequences $\{\alpha_k\}$, $\{\beta_k\}$, $\{\gamma_k\}$, $\{c_k\}$.

Initialize: iteration counter $k \leftarrow 0$;

Loop until a stopping criterion is met:

$$\overline{c}_k = c_k / \max\{1, \|D_k\|/\sqrt{d}\},$$

$$q_{k+1} = q_k + \gamma_k(\varphi - I\{Y(\theta_k) \le q_k\}), \tag{20}$$

$$D_{k+1} = D_k + \beta_k \begin{bmatrix} -I\{Y(\theta_k + \overline{c}_k \Delta_k) \le q_k + \overline{c}_k D_k^T \Delta_k\} \\ + I\{Y(\theta_k - \overline{c}_k \Delta_k) \le q_k - \overline{c}_k D_k^T \Delta_k\} \\ \hline 2\overline{c}_k \Delta_k \end{bmatrix}, \tag{21}$$

$$\theta_{k+1} = \Pi_\Theta(\theta_k - \alpha_k D_k), \tag{22}$$

$$k \leftarrow k+1;$$

Both SDQO and SPQO have a multitimescale structure, as reflected by the use of distinct step-sizes γ_k, β_k, and α_k in the recursions. Intuitively speaking, because our discussion on the convergence behavior of the SP estimator has assumed a fixed value for θ_k, the step-size α_k in SPQO should be chosen very small relative to β_k and γ_k. As a result, when viewed from (21) and (20), the increment in θ_k at each step of (22) is almost negligible as if the parameter vector θ_k were held at a constant value. On the other hand, because (21) is designed to iteratively approximate $-\nabla_\theta F(q_k;\theta)|_{\theta=\theta_k}/f(q_k;\theta_k)$, the step-size β_k should be taken to be the largest to warrant proper tracking of the ratio as both q_k and θ_k vary over time. In the same manner, the three recursions in SDQO should also be carried out at different speeds, with their step-sizes satisfying $\alpha_k = o(\gamma_k)$ and $\gamma_k = o(\beta_k)$.

When the black-box function is given by a computer simulation program, it is natural to exploit the use of common random numbers (CRN) (e.g., Law 2013)—in the same spirit as the use of CRN in, for example, simultaneous-perturbation stochastic approximation (SPSA) (Kleinman et al. 1999)—for reducing the variance in the difference estimates. With a slight abuse of notation, let $Y(U_k;\hat{\theta}_k \pm \tilde{c}_k e_i)$ be the output random variables simulated using the same input random number stream U_k under the perturbed vectors $\hat{\theta}_k \pm \tilde{c}_k e_i$ ($i = 1, \ldots, d$). The CRN version of SDQO simply works by replacing $Y(\hat{\theta}_k \pm \tilde{c}_k e_i)$ in (19) with $Y(U_k;\hat{\theta}_k \pm \tilde{c}_k e_i)$. Likewise, a CRN version of SPQO can be easily implemented by substituting $Y(U_k;\theta_k \pm \overline{c}_k \Delta_k)$ for $Y(\theta_k \pm \overline{c}_k \Delta_k)$ in (21). In Section 3.1, we give conditions under which we show that this CRN approach helps to induce a positive correlation between the indicator terms and reduces the estimation variance at each step.

3. Convergence Results

To fix ideas and to avoid unnecessary repetition, we perform detailed analysis and present results mainly for the SPQO algorithm. All results obtained for SPQO can be shown (with appropriate/slight modifications) to hold for SDQO. We begin by defining $(\Omega, \mathcal{F}, P)$ as the probability space induced by SPQO, where Ω is the set of all sample trajectories that could possibly be observed by executing the algorithm, $\mathcal{F}$ is the σ-field of

subsets of Ω, and P is a probability measure on $\mathcal{F}$. We also define $\mathcal{F}_k = \sigma\{D_0, q_0, \theta_0, \ldots, D_k, q_k, \theta_k\}$ as an increasing σ-field representing the information available at iteration $k = 0, 1, \ldots$. For a given vector v, let $\|v\|$ be the Euclidean norm of v, whereas for a matrix A, let $\|A\|$ be the matrix norm induced by the Euclidean norm. For any two real-valued functions $u(k)$ and $v(k)$, we write $u(k) = O(v(k))$ if $\limsup_{k\to\infty} u(k)/v(k) < \infty$ and $u(k) = o(v(k))$ if $\lim_{k\to\infty} u(k)/v(k) = 0$. We assume that the parameter space Θ is a compact convex set described by functional constraints and takes the form $\Theta = \{\theta \in \Re^d : h_j(\theta) \le 0,\ j = 1, \ldots, m\}$, where $h_j(\cdot)$, $j = 1, \ldots, m$ are continuously differentiable functions with their gradients satisfying $\nabla_\theta h_j(\theta) \neq 0$ whenever $h_j(\theta) = 0$ (cf., e.g., Kushner and Yin 1997). Such a characterization is satisfied by many common choices for Θ, including hyperballs, hyperrectangles, and more general convex polytopes. For notational convenience, throughout our analysis, we denote $I_k^+ := I\{Y(\theta_k + \overline{c}_k\Delta_k) \le q_k + \overline{c}_k D_k^T\Delta_k\}$, $I_k^- := I\{Y(\theta_k - \overline{c}_k\Delta_k) \le q_k - \overline{c}_k D_k^T\Delta_k\}$, $F_k^+ := F(q_k + \overline{c}_k D_k^T\Delta_k; \theta_k + \overline{c}_k\Delta_k)$, and $F_k^- := F(q_k - \overline{c}_k D_k^T\Delta_k; \theta_k - \overline{c}_k\Delta_k)$.

3.1. Strong Convergence

The projection $\Pi_\Theta(\cdot)$ in (22) ensures the boundedness of θ_k by projecting an iterate onto the compact region Θ. This operation can be defined through adding an extra correction term Z_k to the recursion (see, e.g., Kushner and Yin 1997, chapter 5), leading to

$$\theta_{k+1} = \theta_k - \alpha_k D_k + \alpha_k Z_k, \tag{23}$$

where $\alpha_k Z_k := \theta_{k+1} - \theta_k + \alpha_k D_k$ is the vector with the shortest Euclidean length needed to bring $\theta_k - \alpha_k D_k$ back onto Θ. In our setting, because Θ is a convex set, Z_k takes values in the convex cone generated by the inward normals to the surface of Θ at the point θ_{k+1}, that is, $Z_k \in -C(\theta_{k+1})$, where

$$C(\theta) := \{\nu \in \Re^d : \nu^T(\tilde{\theta} - \theta) \le 0,\ \ \forall \tilde{\theta} \in \Theta\} \tag{24}$$

is the normal cone to Θ at θ. Note that $C(\theta) = \{0\}$ whenever θ lies in the interior of Θ.

The convergence of SPQO is investigated by following an ordinary differential equation (ODE) argument (e.g., Kushner and Yin 1997, Borkar 2008, Hu et al. 2022). The general idea is to construct interpolations of the iterates $\{D_k, q_k, \theta_k\}_{k=0}^\infty$ by "stretching" them continuously in time and then capture the long-run behavior of these interpolations using a set of coupled ODEs. In particular, our main result is to show that the sequence $\{\theta_k\}$ generated by (23) asymptotically approaches the limiting solution to a projected ODE of the form

$$\dot{\theta}(t) = -\nabla_\theta q_\varphi(\theta)|_{\theta=\theta(t)} + z(t), \quad t \ge 0, \tag{25}$$

where $z(t) \in -C(\theta(t))$ is the minimum force (the real vector with the smallest Euclidean norm) needed to keep the trajectory $\theta(t)$ within the constraint set Θ.

We introduce the list of assumptions that will be used in our analysis.

Assumptions:

A1. *For almost all (q_k, θ_k) pairs, there exists an open neighborhood of (q_k, θ_k), independent of k and $\omega \in \Omega$, such that*

(a) *The partial derivatives $\partial^2 f(y;\theta)/\partial y^2$, $\partial^2 f(y;\theta)/\partial y \partial\theta^T$, $\partial^2 f(y;\theta)/\partial\theta^T\partial\theta^T$, and $\partial^3 F(y;\theta)/\partial\theta^T\partial\theta^T\partial\theta^T$ all exist and are continuous on the neighborhood with their elements uniformly bounded in k and ω.*

(b) *The density function $f(y,\theta) \ge \epsilon$ for all (y,θ) pairs in the neighborhood for some constant $\epsilon > 0$.*

A2. *The random directions $\{\Delta_k\}$ are i.i.d., independent of $\mathcal{F}_k$. Each Δ_k has mutually independent components with the Bernoulli distribution $P(\Delta_{k,i} = 1) = P(\Delta_{k,i} = -1) = 1/2$ for all $i = 1, \ldots, d$.*

A3. *The sequences $\{\alpha_k\}$, $\{\beta_k\}$, $\{\gamma_k\}$, and $\{c_k\}$ satisfy the following conditions:*

(a) $\beta_k, c_k > 0$, $c_k \to 0$, $\sum_{k=0}^\infty \beta_k = \infty$, $\sum_{k=0}^\infty \beta_k^2/c_k^2 < \infty$;
(b) $\gamma_k > 0$, $\sum_{k=0}^\infty \gamma_k = \infty$;
(c) $\alpha_k > 0$, $\sum_{k=0}^\infty \alpha_k = \infty$;
(d) $\alpha_k = o(\gamma_k)$, $\gamma_k = o(\beta_k)$.

Assumption A1(a) is consistent with the condition used in lemma 1 of Spall (1992) but is stated within a quantile optimization context. It ensures the $O(c_k^2)$ order of the estimation bias in (21) (see Lemma 2) and is satisfied when F is three times continuously differentiable (in both arguments) with bounded derivatives. Note that the condition can be weakened to twice differentiability, in which case the order of the estimation bias would become $O(c_k)$. From the discussion at the end of Section 2, because (21) iteratively approximates a gradient of the form $-\nabla_\theta F(q_k;\theta)|_{\theta=\theta_k}/f(q_k;\theta_k)$, Assumption A1(b) ensures that the denominator of the ratio is bounded away from zero, so that the limit of the $\{D_k\}$ sequence (assuming its existence) does not get arbitrarily large; see Wang et al. (2021) for a similar assumption. The suitability of Assumption A1(b) has been discussed in Hu et al. (2022). Specifically, because $f(\cdot;\cdot)$ is continuous and Θ is compact, the condition holds trivially when $\{q_k\}$ lies in a compact set. In practice, this can be guaranteed by truncating the sequence to a large closed interval containing the true quantiles $q(\theta)$ for all $\theta \in \Theta$. It has been shown in Hu et al. (2022) that such a truncation will not have an influence on the convergence behavior of $\{q_k\}$. Both Assumptions A2 and A3 are conditions on the algorithm input parameters. The Bernoulli random direction is perhaps the most commonly used choice when implementing SP estimators. Assumption A3 is also standard in the SA literature (e.g., Kushner and Clark 1978, Spall 1992, Kushner and Yin 1997). Assumption A3(d) is needed in multitimescale SA methods (cf., e.g., Bhatnagar 2005, Borkar 2008, Zhang and Hu 2019, Hu et al. 2022); it guarantees the three recursions to be

performed at timescales that are noticeably distinct from each another (see the discussion at the end of Section 2). The condition, when combined with $\sum_{k=0}^{\infty}\beta_k^2/c_k^2 < \infty$ in Assumption A3(a), implies that $\sum_{k=0}^{\infty}\gamma_k^2 < \infty$ and $\sum_{k=0}^{\infty}\alpha_k^2 < \infty$.

We begin by stating a result that is essential for characterizing the convergence behavior of the algorithm. It shows that the gradient estimators constructed through (21) have finite second-order moments and that the sequence $\{D_k\}$ itself remains bounded almost surely for all k.

Lemma 1. *Assume that Assumptions* A1, A2, *and* A3(a) *hold; then we have* (i) $\sup_k E[\|D_k\|^2] < \infty$; (ii) $\sup_k \|D_k\| < \infty$ *w.p.1.*

Proof. See Section A of the online appendix. □

Lemma 2 below gives an explicit bound on the (conditional) bias introduced by the symmetric SP scheme used in (21). As a result of Lemma 1, the estimation bias goes to zero at the rate $O(c_k^2)$, both almost surely and in expectation.

Lemma 2. *Let Assumptions* A1, A2, *and* A3(a) *hold, and define the bias*

$$b_k(q_k, D_k, \theta_k) := E\left[\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_k}\middle|\mathcal{F}_k\right] + f(q_k,\theta_k)D_k + \nabla_\theta F(q_k;\theta_k).$$

Then we have that (i) $b_k(q_k, D_k, \theta_k) = O(c_k^2)$ *w.p.1;* (ii) $E[\|b_k(q_k, D_k, \theta_k)\|] = O(c_k^2)$.

Proof. From Assumption A2, $1/\Delta_k = \Delta_k$. It follows that

$$E\left[\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_k}\middle|\mathcal{F}_k\right] = \frac{M_k}{2c_k}E[(-I_k^+ + I_k^-)\Delta_k|\mathcal{F}_k] = \frac{M_k}{2c_k}E[(-F_k^+ + F_k^-)\Delta_k|\mathcal{F}_k].$$

Let $\nabla_\theta^3 F(y;\theta) := \partial^3 F(y;\theta)/\partial\theta^T\partial\theta^T\partial\theta^T$ be the tensor of F. Note that the tensor when evaluated at a vector v of appropriate dimension, denoted by $\nabla_\theta^3 F(y;\theta)[v]$, gives a matrix. Thus, by a third-order Taylor series expansion of F_k^+ and F_k^- around (q_k,θ_k) and then using $\Delta_{k,i}^2 = 1$, $E[\Delta_{k,i}\Delta_{k,j}|\mathcal{F}_k] = 0$ for all $i \neq j$, we obtain

$$\begin{aligned}E\left[\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_k}\middle|\mathcal{F}_k\right] &= E[(-f(q_k;\theta_k)\Delta_k^T D_k - \nabla_\theta^T F(q_k;\theta_k)\Delta_k)\\ &\quad \Delta_k|\mathcal{F}_k] + E[R_3(\overline{q}_k^+,\overline{q}_k^-,\overline{\theta}_k^+,\overline{\theta}_k^-)\Delta_k|\mathcal{F}_k]\\ &= -f(q_k;\theta_k)D_k - \nabla_\theta F(q_k;\theta_k)\\ &\quad + E[R_3(\overline{q}_k^+,\overline{q}_k^-,\overline{\theta}_k^+,\overline{\theta}_k^-)\Delta_k|\mathcal{F}_k],\end{aligned}$$

where $\overline{q}_k^+$, $\overline{q}_k^-$ are on the line segments between q_k and $q_k \pm \overline{c}_k D_k^T\Delta_k$, $\overline{\theta}_k^+$, $\overline{\theta}_k^-$ are on the line segments connecting θ_k and $\theta_k \pm \overline{c}_k\Delta_k$, and R_3 is a remainder term whose absolute value is bounded by

$$\begin{aligned}|R_3| \le{}& \frac{c_k^2}{12M_k^2}(|f_{yy}(\overline{q}_k^+;\overline{\theta}_k^+)| + |f_{yy}(\overline{q}_k^-;\overline{\theta}_k^-)|)|D_k^T\Delta_k|^3\\ &+ \frac{c_k^2}{12M_k^2}(|\Delta_k^T\nabla_\theta^3 F(\overline{q}_k^+;\overline{\theta}_k^+)[\Delta_k]\Delta_k|\\ &+ |\Delta_k^T\nabla_\theta^3 F(\overline{q}_k^-;\overline{\theta}_k^-)[\Delta_k]\Delta_k|)\\ &+ \frac{c_k^2}{4M_k^2}(|\Delta_k^T\nabla_\theta^2 f(\overline{q}_k^+;\overline{\theta}_k^+)D_k^T\Delta_k\Delta_k|\\ &+ |\Delta_k^T\nabla_\theta^2 f(\overline{q}_k^-;\overline{\theta}_k^-)D_k^T\Delta_k\Delta_k|)\\ &+ \frac{c_k^2}{4M_k^2}(|\nabla_\theta^T f_y(\overline{q}_k^+;\overline{\theta}_k^+)(D_k^T\Delta_k)^2\Delta_k|\\ &+ |\nabla_\theta^T f_y(\overline{q}_k^-;\overline{\theta}_k^-)(D_k^T\Delta_k)^2\Delta_k|).\end{aligned}$$

Note that the facts $M_k \ge 1$, $\|\overline{c}_k D_k\| = c_k\|D_k\|/M_k \le c_k\sqrt{d}$, and $\|\overline{c}_k\Delta_k\| \le c_k\|\Delta_k\| = c_k\sqrt{d}$ ensure that the pairs $(\overline{q}_k^+, \overline{\theta}_k^+)$ and $(\overline{q}_k^-, \overline{\theta}_k^-)$ will all lie within the neighborhood of (q_k,θ_k) stated in Assumption A1 as $c_k \to 0$. Thus, invoking Assumption A1 and part (*ii*) of Lemma 1, we arrive at the conclusion that $E[\|R_3(\overline{q}_k^+,\overline{q}_k^-,\overline{\theta}_k^+,\overline{\theta}_k^-)\Delta_k\| |\mathcal{F}_k] = O(c_k^2)$ w.p.1. This completes the proof of part (*i*) of the lemma. Part (*ii*) follows from a simple application of Hölder's inequality and the fact that $\sup_k E[\|D_k\|^2] < \infty$ (Lemma 1, part (*i*)). □

We now present the main convergence theorem. In its most general form, the result implies that the local/global convergence of the algorithm can be determined by examining the asymptotic behavior of the ODE (25). For example, if the ODE has multiple isolated stable equilibrium points, then the sequence $\{\theta_k\}$ will converge to one of them. A strengthened version of the result is obtained when the quantile function is strictly convex, in which case the unique optimal solution θ^* to (1) turns out to be a globally asymptotically stable equilibrium of (25) (see corollary 1 of Hu et al. 2022), so the sequence $\{\theta_k\}$ will converge to θ^* w.p.1. Because the proof is similar to the convergence analysis in Hu et al. (2022), it is included in the online appendix.

Theorem 1. *Assume that Conditions* A1, A2, *and* A3 *hold. Then the sequence* $\{\theta_k\}$ *generated by SPQO converges to some limit set of the ODE* (25) *w.p.1. In addition, if the objective function* $q_\varphi(\theta)$ *is strictly convex on* Θ, *then the sequence* $\{\theta_k\}$ *converges to the unique optimal solution* θ^* *to* (1) *w.p.1.*

Note that as with Lemma 2, it is readily seen that the SD estimator in (19) also has $O(c_k^2)$ bias. Using this

observation, it can be shown, along the same lines as in the proof of Theorem 1, that the following (same) convergence result holds for SDQO.

Theorem 2. *Assume that Condition* A1 *holds with* (q_k, θ_k) *replaced by* $(\hat{q}_k, \hat{\theta}_k)$. *Then under Assumption* A3, *the sequence* $\{\hat{\theta}_k\}$ *generated by SDQO converges to some limit set of the ODE* (25) *w.p.1. In addition, if the objective function* $q_\varphi(\theta)$ *is strictly convex on* Θ, *then the sequence* $\{\hat{\theta}_k\}$ *converges to the unique optimal solution* θ^* *to* (1) *w.p.1.*

As mentioned in Section 3, under the special setting of simulation optimization, the CRN versions of (19) and (21) could be used in some cases to improve the algorithm efficiency. We close this section by providing conditions that guarantee the variance reduction property of this approach. For simplicity, we only state and prove the result for (21). The variance reduction property of (19) is ensured under the same set of conditions given in Proposition 1 below.

Proposition 1. *Denote by* $\hat{I}_k^{\pm} := I\{Y(U_k; \theta_k \pm \overline{c}_k\Delta_k) \le q_k \pm \overline{c}_k D_k^T \Delta_k\}$. *Let Conditions* A1, A2, *and* A3(a) *hold. Suppose that for any given parameter vector* θ, *the output random variable* $Y(U_k; \theta)$, *when viewed as a function of input random numbers, is monotone in each argument; that is,* $U_{k,i} \le U'_{k,i}$ *implies* $Y(\ldots, U_{k,i-1}, U_{k,i}, U_{k,i+1}, \ldots; \theta) \le Y(\ldots, U_{k,i-1}, U'_{k,i}, U_{k,i+1}, \ldots; \theta)$ *for all* i *or* $Y(\ldots, U_{k,i-1}, U_{k,i}, U_{k,i+1}, \ldots; \theta) \ge Y(\ldots, U_{k,i-1}, U'_{k,i}, U_{k,i+1}, \ldots; \theta)$ *for all* i. *Then we have*

$$Var\left(\frac{-\hat{I}_k^+ + \hat{I}_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\mathcal{F}_k\right) \le Var\left(\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\mathcal{F}_k\right), \quad i = 1, \ldots, d,$$

for all k *w.p.1.*

Proof. The proof relies on an inequality given on pp. 187–188 of Givens and Hoeting (2013), which can be stated as follows: Let $X_1, \ldots, X_k$ be a sequence of i.i.d. random variables, and $g_1(x_1, \ldots, x_k)$ and $g_2(x_1, \ldots, x_k)$ be two functions that are both monotonically nondecreasing (or nonincreasing) in each argument. Then

$$\begin{aligned} &E[g_1(X_1, \ldots, X_k)g_2(X_1, \ldots, X_k)] \\ &\ge E[g_1(X_1, \ldots, X_k)]E[g_2(X_1, \ldots, X_k)]. \end{aligned} \tag{26}$$

Now fix an $i = 1, \ldots, d$; it is straightforward to show that

$$\begin{aligned} &Var\left(\frac{-\hat{I}_k^+ + \hat{I}_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\Delta_k, \mathcal{F}_k\right) - Var\left(\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\Delta_k, \mathcal{F}_k\right) \\ &= \frac{1}{2\overline{c}_k^2\Delta_{k,i}^2}[E[I_k^+|\Delta_k, \mathcal{F}_k]E[I_k^-|\Delta_k, \mathcal{F}_k] - E[\hat{I}_k^+\hat{I}_k^-|\Delta_k, \mathcal{F}_k]]. \end{aligned}$$

Because for fixed q_k, D_k, Δ_k, and θ_k, the indicator functions $I\{\cdot \le q_k \pm \overline{c}_k D_k^T\Delta_k\}$ are nonincreasing and by our assumption $Y(\cdot; \theta_k \pm \overline{c}_k\Delta_k)$ are monotone in each argument, the compositions $\hat{I}_k^{\pm} = I\{Y(\cdot; \theta_k \pm \overline{c}_k\Delta_k) \le q_k \pm \overline{c}_k D_k^T\Delta_k\}$ are also monotone in each argument. Hence, we have from (26) that $E[\hat{I}_k^+\hat{I}_k^-|\Delta_k, \mathcal{F}_k] \ge E[\hat{I}_k^+|\Delta_k, \mathcal{F}_k]E[\hat{I}_k^-|\Delta_k, \mathcal{F}_k] = E[I_k^+|\Delta_k, \mathcal{F}_k]E[I_k^-|\Delta_k, \mathcal{F}_k]$. This shows that $Var\left(\frac{-\hat{I}_k^+ + \hat{I}_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\Delta_k, \mathcal{F}_k\right) \le Var\left(\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\Delta_k, \mathcal{F}_k\right)$. Finally, by noticing that

$$\begin{aligned} &Var\left(E\left[\frac{-\hat{I}_k^+ + \hat{I}_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\Delta_k, \mathcal{F}_k\right]\middle|\mathcal{F}_k\right) \\ &= Var\left(E\left[\frac{-I_k^+ + I_k^-}{2\overline{c}_k\Delta_{k,i}}\middle|\Delta_k, \mathcal{F}_k\right]\middle|\mathcal{F}_k\right), \end{aligned}$$

the proof is completed by unconditioning on Δ_k using the law of total variance. □

Proposition 1 shows that when the simulation output random variables react monotonically with respect to the input random numbers, the conditional variance of the gradient estimator D_{k+1} constructed using CRN is always no greater than that obtained under independent sampling. This monotonicity requirement can be expected in certain applications such as in the simulation of regenerative processes and queueing systems (see, e.g., Law 2013). Note that Proposition 1 is a finite-time result that holds almost surely for every k. This is different from the work of Kleinman et al. (1999), in which the same CRN approach has been used in SPSA and shown to lead to a faster asymptotic convergence rate than the original SPSA without CRN. Kleinman et al. (1999), however, consider mean-based simulation optimization, and a key condition used in deriving their result, when put into our current context, requires the two indicator functions $\hat{I}_k^{\pm}$ to be differentiable with bounded derivatives. So, their result does not directly carry over to the quantile setting.

4. Convergence Rate Analysis

Again, because SPQO differs from SDQO only in the perturbation scheme used in constructing gradient estimates (i.e., simultaneous versus element-wise perturbation), we use SPQO as a representative algorithm and investigate its rate of convergence in detail. A completely analogous argument, which we omit, yields essentially the same rate result for SDQO (see Theorem 4 at the end of this section). Our analysis assumes that the ODE (25) has a unique globally asymptotically stable equilibrium θ^* that lies in the interior of Θ. Clearly, because $C(\theta^*) = \{0\}$, θ^* satisfies $\nabla_\theta q_\varphi(\theta)|_{\theta=\theta^*} = 0$, and according to Theorem 1, the sequence $\{\theta_k\}$ generated by (22) converges to θ^* w.p.1. The main result for SPQO is obtained through the repeated application of a fixed-point argument for characterizing the mean absolute errors (MAEs) of SA estimates. The novelty of

the approach resides in the use of a sequence of suitably designed contraction mappings to quantify the estimation errors accumulated over the iterations. This, in essence, translates a difficult rate analysis problem into the simple task of examining the fixed points of a sequence of contraction mappings. Notice that the three recursions in SPQO each represent a major category of SA algorithms, that is, a stochastic rooting finding procedure (20), a Kiefer-Wolfowitz/SPSA-type algorithm (21), and a Robbins-Monro-type gradient iteration (22). We show that these recursions, even when coupled through different timescales, can all be analyzed using the proposed fixed-point argument.

The analysis proceeds in three steps. First, we consider the two coupled iterations (20) and (22) and derive the convergence speed of the quantile recursion (20) while taking into account the variations in the value of θ_k. Then, we characterize the rate at which the gradient recursion (21) converges as both q_k and θ_k vary over time. Finally, we present the main convergence rate result for (22) and discuss the selection of algorithm parameters that optimizes the performance of SPQO. Throughout this section, we focus on standard step- and perturbation-sizes of the forms $\alpha_k = a/k^{\alpha}$, $\beta_k = b/k^{\beta}$, $\gamma_k = r/k^{\gamma}$, and $c_k = c/k^{\tau}$, where $\alpha, \beta, \gamma, \tau \in (0,1)$ and $a, b, r, c > 0$. Let $q(\theta)$ be twice continuously differentiable with Hessian matrix $H(\theta) := \nabla^2_{\theta} q(\theta)$. In addition to the assumptions used in Section 3.1, we also impose the following regularity conditions:

Assumptions:

B1. *For almost all (q_k, θ_k) pairs, there exist constants $\varepsilon, C_f > 0$ such that $\varepsilon \le f(y;\theta_k) \le C_f$ for all y in the interval between q_k and $q(\theta_k)$.*

B2. *The output density $f(y;\theta)$ is jointly continuous in both y and θ. There are constants $L_f, L_F > 0$ such that $|f(y_1;\theta) - f(y_2;\theta)| \le L_f \|y_1 - y_2\|$ and $\|\nabla_{\theta} F(y_1;\theta) - \nabla_{\theta} F(y_2;\theta)\| \le L_F \|y_1 - y_2\|$ for all $\theta \in \Theta$.*

B3. *Let $\lambda(\theta)$ be the smallest eigenvalue of $H(\theta)$. There is a constant $\varrho > 0$ such that $\lambda(\theta) \ge \varrho$ for all θ on the line segment between θ_k and θ^*.*

Note that because no knowledge of the bounding constants ε and C_f is required, Condition B1 is acceptable in many practical situations. For example, when the simulation outputs themselves are bounded or truncated to a large interval, it is easy to see from (20) that $\{q_k\}$ will stay bounded. Thus, the assumption holds if $f(\cdot;\cdot)$ is continuous on Θ and a compact interval that contains $\{q_k\}$ and $\{q(\theta_k)\}$ (see also the remarks on Assumption A1(b) in Section 3.1). Assumption B2 is roughly a globalized version of Assumption A1(a) but without requirements on the higher-order derivatives of F. A sufficient condition for Assumption B2 to hold is that the output distribution F is twice differentiable in both arguments and has bounded derivatives. Assumption B3, in a sense, is a stronger version of the strict convexity assumption on $q(\theta)$ used in Theorem 1 and is satisfied when $q(\theta)$ is strongly convex on Θ, which is a condition frequently adopted in the literature for analyzing the convergence rates of gradient descent algorithms (e.g., Ghadimi and Lan 2012, Bottou et al. 2018).

The following lemma provides a good estimate for the weighted sum of a sequence of decreasing functions of order $O(1/k^s)$, $s \in (0,1)$. The result will be repeatedly used in the subsequent analysis.

Lemma 3. *Let $u(i) = a/i^p$ and $w(i) = O(1/i^s)$, where $a > 0$, $p, s \in (0,1)$, and $w(i) > 0$ for all $i = 1, 2, \ldots$. Then*

$$\sum_{i=1}^{k} \left[\prod_{j=i+1}^{k} (1 - u(j)) \right] u(i) w(i) = O(k^{-s}).$$

Proof. See Section C of the online appendix. □

A characterization of the convergence rate for the mean squared errors of the quantile estimates is given below; see Section D of the online appendix for a proof.

Lemma 4. *Assume Assumptions* A1–A3 *and* B1 *hold. Then the sequence $\{q_k\}$ generated by* (20) *satisfies*

$$\sqrt{E[(q_k - q(\theta_k))^2]} = O\left(\frac{\alpha_k}{\gamma_k}\right) + O\left(\gamma_k^{\frac{1}{2}}\right).$$

From the discussion in Section 2, the q_k iteration (20) is an SA method for solving a sequence of time-varying root-finding problems. The $O(\alpha_k/\gamma_k)$ term above reflects the influence of the slowest component θ_k on the tracking ability of the $\{q_k\}$ sequence. In particular, a large α_k value implies that the underlying input parameter θ_k will change quickly over the iterations, in which case the step-size γ_k should decay sufficiently slowly to ensure that the $\{q_k\}$ sequence could properly follow the true quantiles $\{q(\theta_k)\}$. When θ_k is fixed, that is, $\alpha_k = 0$, it can be seen from the proof of Lemma 4 that the rate of convergence of $\{q_k\}$ to the true quantile in MAE is of order $O(\gamma_k^{1/2})$. Therefore, if (20) is used as a stand-alone procedure for estimating distribution quantiles, its best rate of convergence is $O(1/\sqrt{k})$ (e.g., when $\gamma_k = r/k$). This is consistent but stronger than the classical (weak) convergence rate result for root-finding SA algorithms.

Based on Lemma 4, we further obtain the following convergence rate result for the gradient estimates $\{D_k\}$.

Lemma 5. *Assume Assumptions* A1–A3, B1, *and* B2 *hold. Then the sequence $\{D_k\}$ satisfies*

$$\sqrt{E[\|D_k - \nabla_{\theta} q(\theta)|_{\theta=\theta_k}\|^2]} = O\left(\frac{\alpha_k}{\gamma_k}\right) + O(c_k^2) + O\left(\frac{\beta_k^{\frac{1}{2}}}{c_k}\right).$$

Proof. See Section E of the online appendix. □

The result also has an intuitive explanation. A comparison of the results of Lemmas 4 and 5 indicates that the $O(\alpha_k/\gamma_k)$ term is attributed to the approximation error of the quantile estimator. Lemma 5 shows that this error sets a limit on the convergence speed of $\{D_k\}$, suggesting that the performance of the two coupled SA recursions is primarily governed by the rate of the slower component. In the special case when the input parameter vector is fixed, the $O(\alpha_k/\gamma_k)$ term vanishes. Consequently, if (20) and (21) are jointly used as a means for quantile sensitivity analysis, then the lemma implies that the convergence rate of the gradient estimates is $O(c_k^2) + O(\beta_k^{1/2}/c_k)$ provided that $\gamma_k = o(\beta_k)$ (because of Assumption A3(d)). If in addition the quantile is also fixed, then (21) alone is in the form of the standard SPSA algorithm, and the rate result is simply given by $O(c_k^2) + O(\beta_k^{1/2}/c_k)$. When $\beta \le 6\tau$, this further reduces to $O(\beta_k^{1/2}/c_k)$, which becomes identical to the (asymptotic) rate result previously obtained in Spall (1992), except in the mode of convergence.

Finally, we arrive at the following main convergence rate result for SPQO.

Theorem 3. *If Conditions* A1–A3 *and* B1–B3 *hold, then the sequence* $\{\theta_k\}$ *generated by SPQO satisfies*

$$E[\|\theta_k - \theta^*\|] = O\left(\frac{\alpha_k}{\gamma_k}\right) + O(c_k^2) + O\left(\frac{\beta_k^{\frac{1}{2}}}{c_k}\right). \tag{27}$$

Proof. Define $\psi_k := \theta_k - \theta^*$. We have from (23) that

$$\begin{aligned}\psi_{k+1} &= \psi_k - \alpha_k(D_k - \nabla_\theta q(\theta)|_{\theta=\theta_k}) - \alpha_k \nabla_\theta q(\theta)|_{\theta=\theta_k} + \alpha_k Z_k \\ &= \psi_k - \alpha_k \eta_k - \alpha_k \nabla_\theta q(\theta)|_{\theta=\theta_k} + \alpha_k Z_k,\end{aligned}$$

where $\eta_k = D_k - \nabla_\theta q(\theta)|_{\theta=\theta_k}$. Because $\nabla_\theta q(\theta)|_{\theta=\theta^*} = 0$, a Taylor series expansion of $\nabla_\theta q(\theta)|_{\theta=\theta_k}$ around θ^* shows that

$$\begin{aligned}\psi_{k+1} &= \psi_k - \alpha_k \eta_k - \alpha_k H(\overline{\theta}_k)\psi_k + \alpha_k Z_k \\ &= (I - \alpha_k H(\overline{\theta}_k))\psi_k - \alpha_k \eta_k + \alpha_k Z_k,\end{aligned}$$

where $\overline{\theta}_k$ is on the line segment between θ_k and θ^*. Taking norms on both sides, using the Rayleigh-Ritz inequality (cf., e.g., Rugh 1996) and Condition B3, we obtain that for all k sufficiently large such that $\alpha_k \varrho < 1$,

$$\begin{aligned}\|\psi_{k+1}\| &\le \|(I - \alpha_k H(\overline{\theta}_k))\psi_k\| + \alpha_k\|\eta_k\| + \alpha_k\|Z_k\| \\ &\le (1 - \alpha_k\varrho)\|\psi_k\| + \alpha_k\|\eta_k\| + \alpha_k\|Z_k\|.\end{aligned}$$

It follows that

$$E[\|\psi_{k+1}\|] \le (1 - \alpha_k\varrho)E[\|\psi_k\|] + \alpha_k E[\|\eta_k\|] + \alpha_k E[\|Z_k\|]. \tag{28}$$

We now derive a bound for $E[\|Z_k\|]$. Because θ^* is in the interior of Θ, there is a constant $\varsigma > 0$ such that the 2ς-neighborhood of θ^* is contained in Θ. Let $\mathcal{E}_k = \{\|\theta_{k+1} - \theta^*\| \ge 2\varsigma\}$. Note that by (24), the occurrence of $\mathcal{E}_k^c$ implies that $Z_k = 0$. Using this observation, we obtain that

$$\begin{aligned}E[\|Z_k\|] &= E[\|Z_k\||\mathcal{E}_k]P(\mathcal{E}_k) + E[\|Z_k\||\mathcal{E}_k^c]P(\mathcal{E}_k^c) \\ &\le E[\|D_k\|]P(\|\theta_{k+1} - \theta^*\| \ge 2\varsigma) \\ &\le E[\|D_k\|]P(\|\theta_{k+1} - \theta_k\| \ge \varsigma \cup \|\theta_k - \theta^*\| \ge \varsigma) \\ &\le E[\|D_k\|]\frac{E[\|\theta_{k+1} - \theta_k\|]}{\varsigma} + E[\|D_k\|]\frac{E[\|\psi_k\|]}{\varsigma} \\ &\quad \text{by Markov's inequality} \\ &\le \frac{2\alpha_k E^2[\|D_k\|]}{\varsigma} + E[\|D_k\|]\frac{E[\|\psi_k\|]}{\varsigma},\end{aligned} \tag{29}$$

where the first inequality is due to the fact that $\|Z_k\| \le \|D_k\|$ (see the proof of lemma 5 in Hu et al. 2022), and the last step follows from $\|\theta_{k+1} - \theta_k\| \le \alpha_k\|D_k - Z_k\| \le 2\alpha_k\|D_k\|$.

Next, substitute the bound (29) into (28) and combine like terms,

$$\begin{aligned}E[\|\psi_{k+1}\|] \le &\left(1 - \alpha_k\left(\varrho - \frac{E[\|D_k\|]}{\varsigma}\right)\right)E[\|\psi_k\|] \\ &+ \alpha_k E[\|\eta_k\|] + \frac{2\alpha_k^2 E^2[\|D_k\|]}{\varsigma}.\end{aligned}$$

Note that because $\theta_k \to \theta^*$ w.p.1 and $\nabla_\theta q(\theta)|_{\theta=\theta^*} = 0$, the continuity of $\nabla_\theta q(\theta)$ indicates that $\|\nabla_\theta q(\theta)|_{\theta=\theta_k}\| \to 0$ w.p.1. This, together with the boundedness of $\nabla_\theta q(\theta)$ (because of the compactness of Θ), shows that $E[\|\nabla_\theta q(\theta)|_{\theta=\theta_k}\|] \to 0$ by the dominated convergence theorem. Thus, we have from Lemma 5 that $E[\|D_k\|] \le E[\|D_k - \nabla_\theta q(\theta)|_{\theta=\theta_k}\|] + E[\|\nabla_\theta q(\theta)|_{\theta=\theta_k}\|] \to 0$. This then implies the existence of an integer $\mathcal{N} > 0$ such that $\varrho - E[\|D_k\|]/\varsigma \ge \varrho/2 := \overline{\varrho}$ for all $k \ge \mathcal{N}$. Consequently, we get that for all $k \ge \mathcal{N}$

$$\begin{aligned}E[\|\psi_{k+1}\|] \le &(1 - \alpha_k\overline{\varrho})E[\|\psi_k\|] + \alpha_k E[\|\eta_k\|] \\ &+ \frac{2\alpha_k^2 E^2[\|D_k\|]}{\varsigma}.\end{aligned}$$

Directly expanding the above inequality from term $\mathcal{N}$ onwards,

$$\begin{aligned}E[\|\psi_k\|] \le &\prod_{i=\mathcal{N}}^{k}(1 - \alpha_i\overline{\varrho})E[\|\psi_{\mathcal{N}}\|] \\ &+ \sum_{i=\mathcal{N}}^{k}\left[\prod_{j=i+1}^{k}(1 - \alpha_j\overline{\varrho})\right]\alpha_i E[\|\eta_i\|] \\ &+ \sum_{i=\mathcal{N}}^{k}\left[\prod_{j=i+1}^{k}(1 - \alpha_j\overline{\varrho})\right]\alpha_i\frac{2\alpha_i E^2[\|D_i\|]}{\varsigma}.\end{aligned}$$

Finally, because $\prod_{i=\mathcal{N}}^{k}(1-\alpha_i\overline{\varrho}) = o(k^{-1})$, $E[\|\eta_k\|] = O(\alpha_k/\gamma_k) + O(c_k^2) + O(\beta_k^{1/2}/c_k)$ (Lemma 5), and $\alpha_k E^2[\|D_k\|] = O(\alpha_k)$ (Lemma 1(i)), a direct application of Lemma 3 leads to the conclusion $E[\|\psi_k\|] = O(\alpha_k/\gamma_k) + O(c_k^2) + O(\beta_k^{1/2}/c_k)$. □

In view of Lemma 5, regardless of the choice of α_k, the MAE of D_k converges at a rate that is always slower than α_k itself, which results in the long-run behavior of (22) being dominated by the errors in gradient estimation. Therefore, in contrast to single-timescale SA, whose rate is determined by its step-sizes, an interesting observation from Theorem 3 is that the step-size α_k in (22) does not have a direct effect on the convergence rate of $\{\theta_k\}$, but only does so indirectly through the expression for the convergence rate of the faster component D_k in Lemma 5.

From Assumption A3(a), both β_k and c_k should be chosen to satisfy $\tau + 1/2 < \beta \le 1$. In addition, to improve the rate in (27), it is clear that α should be taken large, which gives the obvious choice $\alpha \approx 1$ (note that $\alpha \in (0,1)$). On the other hand, because $\beta_k^{1/2}/c_k = O(k^{-(\beta/2-\tau)})$ and $c_k^2 = O(k^{-2\tau})$, the $O(c_k^2) + O(\beta_k^{1/2}/c_k)$ term is optimized when $\beta = 6\tau$, yielding $O(c_k^2) = O(\beta_k^{1/2}/c_k) = O(k^{-2\tau})$. Consequently, by equating the terms in (27), we find that under the above choice of α, β and the constraint $\alpha > \gamma > \beta$ (Assumption A3(d)), an upper bound on the MAEs of $\{\theta_k\}$ diminishes at an optimal rate that can be made arbitrarily close to $O(k^{-1/4})$, which is approximately attained when $\alpha \approx 1$, $\gamma = 3/4$, $\beta \approx 3/4$, and $\tau = 1/8$.

Finally, for the sake of completeness, we conclude by stating the following rate result we have for SDQO, implying that the algorithm essentially shares the same $O(k^{-1/4})$ best convergence rate bound as SPQO, except possibly in the constant contained in the big-O notation.

Theorem 4. *Assume that the conditions of Theorem 2 and Assumptions* B1–B3 *hold but with* $(\hat{q}_k, \hat{\theta}_k)$ *replacing* (q_k, θ_k) *in Assumption* B1. *Then the MAEs of the sequence* $\{\hat{\theta}_k\}$ *generated by SDQO satisfy*

$$E[\|\hat{\theta}_k - \theta^*\|] = O\left(\frac{\alpha_k}{\gamma_k}\right) + O(c_k^2) + O\left(\frac{\beta_k^{\frac{1}{2}}}{c_k}\right).$$

5. Simulation Experiments

We begin with two simple examples in Section 5.1 to illustrate the smoothness requirement on the output distribution. Then, in Sections 5.2 and 5.3, we test the algorithms by performing some computational experiments on a set of artificially created black-box functions and a queueing example. In all cases, the performance of SPQO and SDQO is compared with that of the QG algorithm and the surrogate-based gTSSO-QML algorithm proposed in Wang et al. (2021). We describe the latter two algorithms in more detail now. As discussed in Section 1, QG is a single-timescale SA algorithm that employs the conventional SD (3) to approximate quantile gradients, where the true quantiles are estimated by order statistics. Denote by $\upsilon_k > 0$ the perturbation size, and let $\hat{q}(\theta)$ stand for the $\lceil n_k\varphi \rceil$th order statistic of an output sample $Y_1, \ldots, Y_{n_k} \sim F(\cdot;\theta)$ of size n_k. The QG algorithm uses the following update:

$$\theta_{k+1} = \Pi_\Theta(\theta_k - \rho_k \tilde{D}_k), \tag{30}$$

where $\rho_k > 0$ is the step-size and $\tilde{D}_k$ is the gradient estimate whose ith element is given by

$$\tilde{D}_{k,i} = \frac{\hat{q}((\tilde{\theta}_{k,1}, \ldots, \theta_{k,i} + \upsilon_k, \ldots, \tilde{\theta}_{k,d})^T) - \hat{q}((\tilde{\theta}_{k,1}, \ldots, \theta_{k,i} - \upsilon_k, \ldots, \tilde{\theta}_{k,d})^T)}{2\upsilon_k}, \; i = 1, \ldots, d.$$

Here, $\theta_{k,i}$ is the ith element of θ_k and $\tilde{\theta}_{k,j}$'s are random variables uniformly distributed over $[\theta_{k,j} - \upsilon_k, \theta_{k,j} + \upsilon_k]$ for all $j = 1, \ldots, d$, $j \ne i$. Note that this construction differs from our proposed algorithms in that $2dn_k$ function evaluations (as opposed to three in SPQO and $2d+1$ in SDQO) are needed at each iteration. Our implementation of QG is based on the parameter values $\rho_k = 1/k$, $\upsilon_k = 1/k^{0.501}$, and $n_k = \lceil k^{2.003} \rceil$, which are the minimum required to satisfy the conditions for the convergence of the algorithm (see theorem 8 of Kibzun and Matveev 2012). The gTSSO-QML algorithm uses a stochastic cokriging model to approximate the response surfaces of a set of quantile functions with progressively increasing quantile levels and selects new design points by optimizing an expected improvement criterion. Following Wang et al. (2021), the algorithm parameter values are determined in our implementation based on a cross-validation test, and at each step, simulation samples are adaptively allocated to the selected design points by using the optimal computing budget allocation method.

Similar to many other Bayesian optimization approaches, gTSSO-QML is very computationally demanding on high-dimensional problems, so we have implemented the algorithm on a parallel computing platform with 164 nodes. Each node has two Intel Xeon E5-2683v3 processors (with 14 cores each running at 2.0 GHz) and 128 GB of memory. The computational experiments for all other algorithms are performed on a Window PC with an Intel Core i5 1.8-GHz processor and 8 GB of memory.

5.1. Necessity of Differentiability for Convergence

As discussed in Section 3.1, Assumption A1(a) is primarily used to guarantee the typical $O(c_k^2)$ bias order of the symmetric FD scheme employed in (19) and (21). The assumption, however, is stronger than necessary

for the convergence of the algorithms, which we illustrate through two examples.

(i) Consider the class of output random variables

$$Y(\theta) = |\theta|^{\ell} + X,\ \Theta = [-2, 2], \tag{31}$$

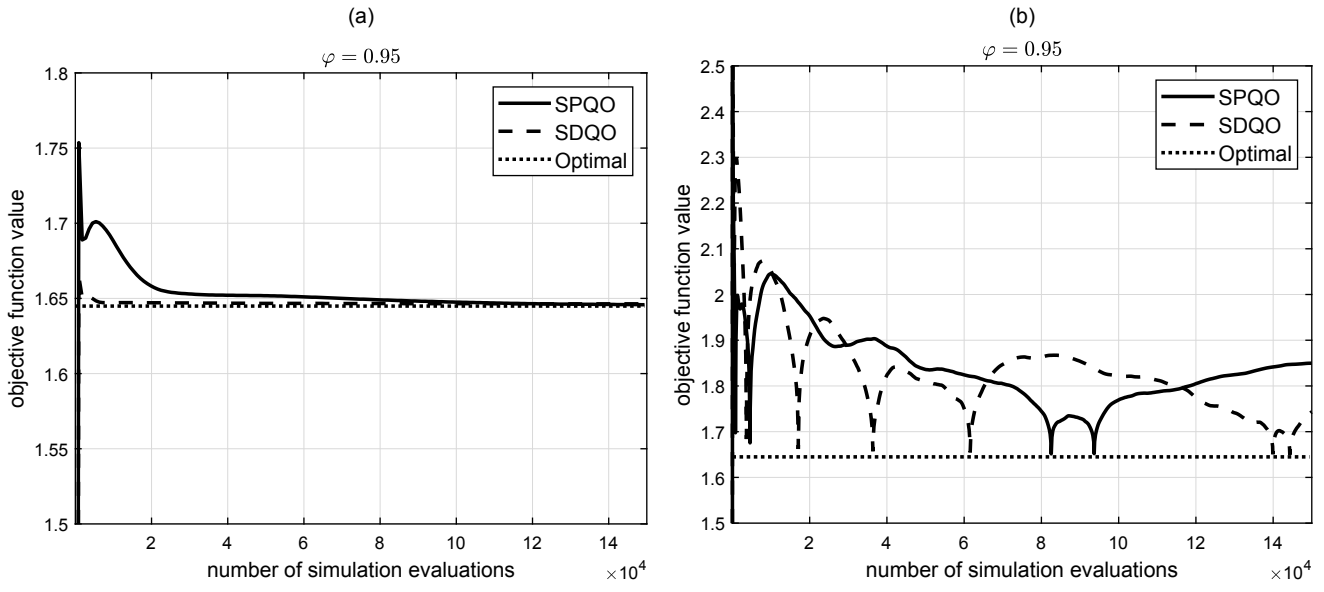
where $\ell \in (1,2)$, and $X \sim N(0,1)$ is the standard normal random variable. The φ-quantile of $Y(\theta)$ can be expressed in terms of the inverse of the error function $\mathrm{erf}(z) := (2/\sqrt{\pi})\int_0^z e^{-t^2}dt$ as $q_\varphi(\theta) = |\theta|^{\ell} + \sqrt{2}\mathrm{erf}^{-1}(2\varphi - 1)$, which is strictly convex with a unique minimizer $\theta^* = 0$. It is easy to see that the output distribution $F(y;\theta)$ is continuously differentiable, but its second-order derivative $\partial^2 F(y;\theta)/\partial^2\theta$ has an infinite discontinuity at $\theta = 0$. This implies that if the algorithms converge to the optimum, then $\{\theta_k\}$ would enter any neighborhood of θ^* infinitely often, within which $\partial^2 F(y;\theta)/\partial^2\theta$ may become arbitrarily large. In other words, if the sequence $\{\theta_k\}$ converges to θ^*, then Assumption A1(a) cannot be true. Nevertheless, by using a first-order (linear) Taylor approximation of $F_k^{\pm}$ in the proof of Lemma 2, it can be shown that the biases of (19) and (21) at $\theta = 0$ are in fact of order $O(c_k)$, which tends to zero as $k \to \infty$ under Assumption A3(a). Therefore, both SPQO and SDQO would converge to θ^* as a result of Theorems 1 and 2. Figure 1(a) empirically illustrates the convergence behavior of SPQO and SDQO on this example when $\ell = 5/3$ and $\varphi = 0.95$, where each curve plots the sequence of quantile values $\{q_\varphi(\theta_k)\}$ (averaged over 100 independent replication runs) versus the number of simulation evaluations.

(ii) On the other hand, consider (31) with $\ell \in (0,1)$. In this case, the first-order derivative $\partial F(y;\theta)/\partial\theta$ contains an essential discontinuity at $\theta = 0$, which results in an infinite quantile derivative at θ^* (see (4) or the analytical expression for $q_\varphi(\theta)$ given in case (i) above). Intuitively, this means that whenever a solution θ_k is obtained in the close vicinity of θ^*, the large quantile derivative will quickly steer the search away from the optimum, leading to possible nonconvergence of the algorithms. Such a phenomenon can be observed from Figure 1(b), which shows the typical oscillatory behavior (i.e., going back and forth between better and worse solutions) of SPQO and SDQO when $\ell = 1/3$. This suggests the necessity of continuous differentiability of the output distribution on the convergence of the algorithms.

5.2. Black-Box Test Functions

Six noisy black-box functions are tested, with dimensions varying from 2 to 20. Case 1 contains multiplicative noise but only two decision variables and is relatively easy to solve. In case 2, the noise also scales the function, but as the problem dimension increases, the distribution of the function may become extremely flat, making its extreme quantiles challenging to estimate. In case 3, on the other hand, the noise is additive, and under the optimal parameter values, the quantiles are very distant from the origin, so predicting their values could also be challenging, especially when the initial estimates are far from the true values. In case 4, the noise is both multiplicative and additive. Cases 5 and 6 are relatively low-dimensional multimodal problems, and each contains a large number of local optima.

Figure 1. Performance of SPQO and SDQO on Problem (31)



Notes. (a) $\ell = 5/3$; (b) $\ell = 1/3$. Both algorithms are implemented based on the parameter setting $\alpha_k = 1/k^{0.99}$, $\beta_k = 1/k^{0.74}$, $\gamma_k = 1/k^{0.75}$, and $c_k = 1/k^{0.125}$.

Table 1. Optimal Quantile Values in the 24 Test Scenarios

	Normal		Cauchy	
Case	$\varphi = 0.6$	$\varphi = 0.95$	$\varphi = 0.6$	$\varphi = 0.95$
1	10	10	10	10
2	0.25	1.64	0.32	6.31
3	−717.25	−715.86	−717.18	−711.19
4	−49.29	−45.32	−49.08	−34.62
5	0.25	1.64	0.32	6.31
6	0.25	1.64	0.32	6.31

Case 1. $Y(\theta) = (2.6(\theta_1^2 + \theta_2^2) - 4.8\theta_1\theta_2)X + 10$, where $\Theta = [-2,2]^2$.

Case 2. $Y(\theta) = (\sum_{i=1}^{d}(\theta_i - i)^2 + 1)X$, where $d = 10$ and $\theta_i \in [i-1, i+1]$ for $i = 1, \ldots, d$.

Case 3. $Y(\theta) = X + \sum_{i=1}^{d}(\theta_i - i)\theta_i$, where $d = 20$ and $\Theta = [-20,20]^d$.

Case 4. $Y(\theta) = \frac{1}{d}\sum_{i=1}^{d}(\theta_i - 1)^2 X + \frac{1}{d}\sum_{i=1}^{d}(\theta_i^4 - 16\theta_i^2 + 5\theta_i)$, where $d = 20$ and $\Theta = [1,4]^d$.

Case 5. $Y(\theta) = \left[-10\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d}\theta_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(\pi\theta_i)\right) + 11 + e\right]X$, where $d = 5$ and $\Theta = [-5,5]^d$.

Case 6. $Y(\theta) = \frac{1}{d}\sum_{i=1}^{d}[0.4\ \sin^2(0.2\pi(\theta_i - 0.9)) + 0.3\sin^2(0.4\pi(\theta_i - 0.9)) + 0.001(\theta_i - 0.9)^2] + X$, where $d = 5$ and $\Theta = [-10,10]^d$.

For each test problem, we consider two quantile levels, $\varphi = 0.6$, $\varphi = 0.95$, and two choices of the (unknown) noise distribution, $X \sim N(0,1)$ and $X \sim Cauchy(0,1)$, resulting in 24 total test scenarios. The optimal quantile values in all scenarios are listed in Table 1. Note that under the Cauchy noise, neither the mean nor the variance of the output distribution exists.

In the implementation of SPQO and SDQO, the decay rates of the parameters are determined from the result of Section 4, that is, $\alpha = 0.99$, $\gamma = 0.75$, $\beta = 0.74$, and $\tau = 0.125$ (see the discussion at the end of that section). Our experience indicates that their performance is not very sensitive to the choice of α_k, in that the standard stepsize $\alpha_k = 2/k^{\alpha}$ seems to work well across a variety of test cases. The parameters β_k and c_k resemble those of SPSA, and we choose them to be of the forms $\beta_k = b/(k+R)^{\beta}$ and $c_k = c/(k+R)^{\tau}$ as suggested in Spall (2003), where R is set to 10% of the maximum number of iterations allowed, $b = \kappa_1(2R)^{\beta}$ and $c = \kappa_2(2R)^{\tau}$. This choice maintains the respective values of β_k and c_k to be greater than κ_1 and κ_2 during the first R iterations. The constants κ_1 and κ_2 are then selected by trial and error, and we find that values satisfying $\kappa_2 \in [0.1, 0.9]$ and $0.005 \le \kappa_1/\kappa_2 \le 0.1$ all yield reasonable performance. Note that from (21) (resp. (19)), the increment (if there is any) in each component of D_k (resp. $\hat{D}_k$) is exactly $\beta_k/2\bar{c}_k$ (resp. $\beta_k/2\hat{c}_k$). So, the lower bound 0.005 on κ_1/κ_2 prevents the updates in gradient estimates from becoming too small. Our numerical results reported here are based on $\kappa_1 = 0.05$ and $\kappa_2 = 0.5$. The choice of γ_k, on the other hand, is most critical to the performance. This is mainly because of the recursive procedure used for estimating q_k (resp. $\hat{q}_k$). As can be observed from (20), each increment in quantile estimate is bounded in magnitude by γ_k. Thus, if a desired (true) quantile is far from the initial q_0, then a reasonable estimate of its value would take an enormously large number of iterations under the standard choice $\gamma_k = 1/k^{\gamma}$, leading to excessively slow (finite-time) convergence behavior. One way to address this issue would be to take γ_k to be of the same form as β_k, so that a nonnegligible gain could be maintained in tracking the true quantile values. In our study, however, we simply take $\gamma_k = R/k^{\gamma}$. The intuitive reason is that the large constant R will provide enough impetus in early iterations to help the iterates move quickly toward the "correct" quantile range, whose values can then be further fine-tuned as γ_k decreases rapidly with k, because of the large decay rate γ.

It can be verified that the monotonicity condition assumed in Proposition 1 is satisfied for all test functions. Therefore, in addition to SPQO and SDQO, we have also implemented their CRN versions, SPQO-CRN and SDQO-CRN. For each of the respective test cases, the six algorithms SPQO, SPQO-CRN, SDQO, SDQO-CRN, QG, and gTSSO-QML are run using the same computational budget, where the total number of evaluations is set to 3×10^4 for case 1, 3×10^5 for cases 2–4, and 10^6 for the multimodal test cases 5 and 6. In SPQO (as well as SDQO and their CRN versions), the initial estimates are taken to be $D_0 = (0, \ldots, 0)^T$ and $q_0 = 0$. The initial θ_0 is uniformly generated from Θ for all algorithms. Each algorithm is then independently repeated 40 times, and the numerical results (averaged over 40 runs) obtained in the respective test cases are presented in Tables 2 and 3, which show the means and standard errors of the true quantile function values at the final solutions found by the six comparison algorithms. In each row of the tables, the result that is closest to the true optimal value is shown in bold (in the case of a tie, the one with a smaller standard error is highlighted). The convergence behavior of SPQO, SDQO, QG, and gTSSO-QML is also illustrated in Figures 2–4, which plot the true quantile values at the current estimated solutions as functions of the numbers of function evaluations consumed.

Our comparison results indicate that SPQO (SPQO-CRN) has the most consistency on test cases 1–4. The final results obtained by SDQO are close to those of SPQO. However, as the figures clearly show, the convergence behavior of SDQO (in terms of the number of function evaluations) becomes slower as the problem dimension increases. For example, in cases 3 and 4, because SDQO uses 41 function evaluations per iteration, its total number of iterations is more than 13 times smaller than that of SPQO, resulting in inferior performance within the allowed budget. As expected, it can

Table 2. Performance on Test Functions for Normally Distributed Noise

Case	SPQO	SPQO-CRN	SDQO	SDQO-CRN	QG	gTSSO-QML
$\varphi = 0.6$						
1	10.06 (8.0e-3)	10.04 (7.8e-3)	10.06 (1.0e-2)	10.04 (7.0e-3)	10.11 (1.3e-2)	**10.01 (7.3e-4)**
2	0.30 (2.7e-3)	**0.28 (1.0e-3)**	0.55 (1.3e-2)	0.43 (1.6e-2)	0.39 (1.4e-2)	0.33 (1.7e-2)
3	−717.24 (3.3e-4)	−717.24 (7.2e-4)	−717.22 (1.4e-3)	**−717.25 (1.9e-4)**	−717.23 (6.7e-4)	−168.00 (2.71)
4	−49.22 (1.8e-3)	−49.19 (8.3e-4)	−49.16 (5.0e-3)	**−49.25 (3.7e-3)**	−49.17 (7.1e-3)	−48.60 (3.9e-2)
5	1.13 (4.4e-2)	1.05 (3.2e-2)	1.28 (3.8e-2)	1.18 (2.5e-2)	1.51 (3.7e-2)	**0.62 (4.1e-2)**
6	0.50 (1.4e-2)	0.52 (1.4e-2)	0.57 (1.6e-2)	0.54 (1.4e-2)	0.54 (1.2e-2)	**0.36 (5.1e-3)**
$\varphi = 0.95$						
1	10.07 (6.6e-3)	10.09 (1.0e-2)	10.04 (5.3e-3)	10.02 (4.0e-3)	10.22 (2.9e-2)	**10.00 (6.0e-6)**
2	1.65 (3.3e-4)	**1.64 (6.0e-6)**	1.68 (2.3e-3)	1.65 (4.6e-5)	1.66 (9.5e-4)	1.69 (3.0e-3)
3	−715.85 (5.6e-5)	**−715.86 (5.1e-6)**	−715.82 (1.6e-3)	−715.85 (1.7e-4)	−715.82 (1.8e-3)	−174.96 (3.59)
4	−45.22 (1.5e-3)	−45.21 (6.8e-4)	−45.13 (9.6e-3)	**−45.31 (9.5e-4)**	−45.05 (1.3e-2)	−44.41 (4.7e-2)
5	4.85 (4.0e-1)	5.31 (3.3e-1)	6.43 (3.0e-1)	5.17 (2.9e-1)	8.52 (1.9e-1)	**4.10 (2.6e-1)**
6	1.91 (1.4e-2)	1.91 (1.6e-2)	1.94 (1.4e-2)	1.93 (1.3e-2)	1.93 (1.5e-2)	**1.76 (6.3e-3)**

Note. Performance based on 40 independent runs (standard errors in parentheses).

be seen from the tables that SPQO-CRN and SDQO-CRN generally outperform their original versions and yield smaller standard errors in almost all test cases, indicating their consistent performance over repeated runs.

The performance of QG is comparable to SPQO under the normal noise setting. In particular, because QG is based on order statistics, the algorithm does not require the specification of an initial estimate and is less susceptible to the magnitude/location of the optimal quantile. This could be beneficial in extreme circumstances such as case 3, especially when the true quantile happens to be very far away from its initial guessed value in a recursive procedure like SPQO. We see from Figure 3 that in case 3, QG quickly identifies the correct quantile range in the first few iterations and shows a very fast initial improvement. However, because of the increasing sample size required at each step, the parameter update in QG is carried out at a frequency that becomes much lower as search progresses, which results in sluggish performance in the long run. From Table 2, we observe that the mean results found by SPQO are closer to the true optimal values than those obtained by QG.

Under the Cauchy noise, QG shows a significant performance degradation. We believe that this is owing to the heavy tail feature of the Cauchy distribution, which makes its high-level quantiles more difficult to estimate than the normal distribution. Therefore, for an order statistic-based estimator, a reliable quantile approximation can only be obtained after a large amount of simulation observations have been collected. This issue is especially manifested on cases 1, 2, and 4, where in each problem the standard Cauchy input distribution is further stretched by a large factor. We observe that at the $\varphi = 0.95$ quantile level, QG may fail to locate a near-optimal solution within the prescribed simulation budget. SPQO instead estimates quantiles and gradients by

Table 3. Performance on Test Functions for Cauchy Distributed Noise

Case	SPQO	SPQO-CRN	SDQO	SDQO-CRN	QG	gTSSO-QML
$\varphi = 0.6$						
1	10.06 (9.4e-3)	10.03 (6.6e-3)	10.07 (1.3e-2)	10.03 (6.3e-3)	10.11 (1.3e-2)	**10.00 (8.7e-4)**
2	0.37 (3.2e-3)	**0.33 (4.8e-4)**	0.55 (1.9e-2)	0.36 (5.0e-3)	0.47 (1.5e-2)	0.37 (4.5e-3)
3	−717.16 (8.5e-4)	**−717.17 (2.9e-5)**	−717.08 (5.5e-3)	−717.17 (1.3e-4)	−717.15 (1.1e-3)	−165.34 (2.52)
4	−48.99 (2.9e-3)	−48.98 (1.3e-3)	−48.75 (1.8e-2)	**−49.03 (4.6e-3)**	−48.87 (1.2e-2)	−48.20 (4.4e-2)
5	1.64 (4.6e-2)	1.34 (4.4e-2)	1.76 (5.7e-2)	1.31 (5.5e-2)	1.97 (5.0e-2)	**0.83 (5.8e-2)**
6	0.60 (1.3e-2)	0.56 (1.2e-2)	0.61 (1.6e-2)	0.60 (1.4e-2)	0.63 (1.3e-2)	**0.43 (5.6e-3)**
$\varphi = 0.95$						
1	10.03 (6.4e-3)	10.00 (5.3e-4)	10.04 (7.9e-3)	10.01 (1.6e-3)	120.03 (18.00)	**10.00 (3.6e-5)**
2	6.48 (8.6e-3)	**6.31 (9.1e-5)**	8.15 (3.5e-1)	6.32 (1.4e-3)	48.05 (1.25)	6.67 (2.2e-2)
3	−711.17 (9.2e-4)	**−711.19 (2.2e-5)**	−709.24 (1.9e-1)	−710.60 (7.0e-2)	−703.94 (3.8e-1)	−160.79 (3.06)
4	−33.80 (3.9e-2)	**−34.20 (1.0e-2)**	−30.16 (1.9e-1)	−33.92 (1.0e-1)	−19.54 (1.06)	−32.10 (1.1e-1)
5	23.96 (2.99)	**8.21 (9.0e-1)**	32.11 (2.99)	16.69 (1.30)	36.74 (1.30)	14.64 (1.23)
6	6.58 (1.6e-2)	6.58 (1.7e-2)	6.64 (1.4e-2)	6.61 (1.6e-2)	6.72 (1.2e-2)	**6.43 (6.5e-3)**

Note. Performance based on 40 independent runs (standard errors in parentheses).

Figure 2. Performance of SPQO, SDQO, QG, and gTSSO-QML on Test Cases 1 and 2

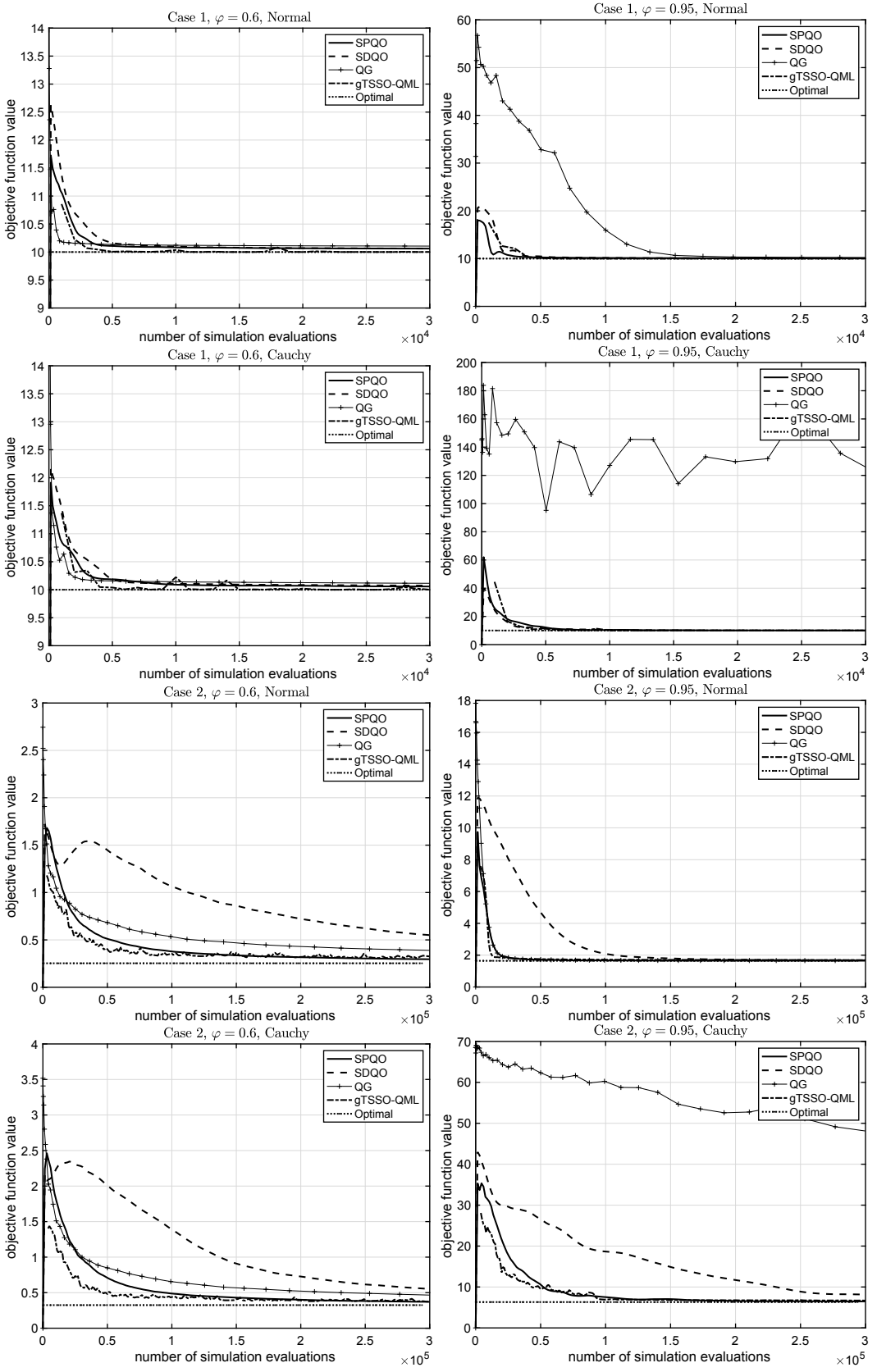


Figure 3. Performance of SPQO, SDQO, QG, and gTSSO-QML on Test Cases 3 and 4

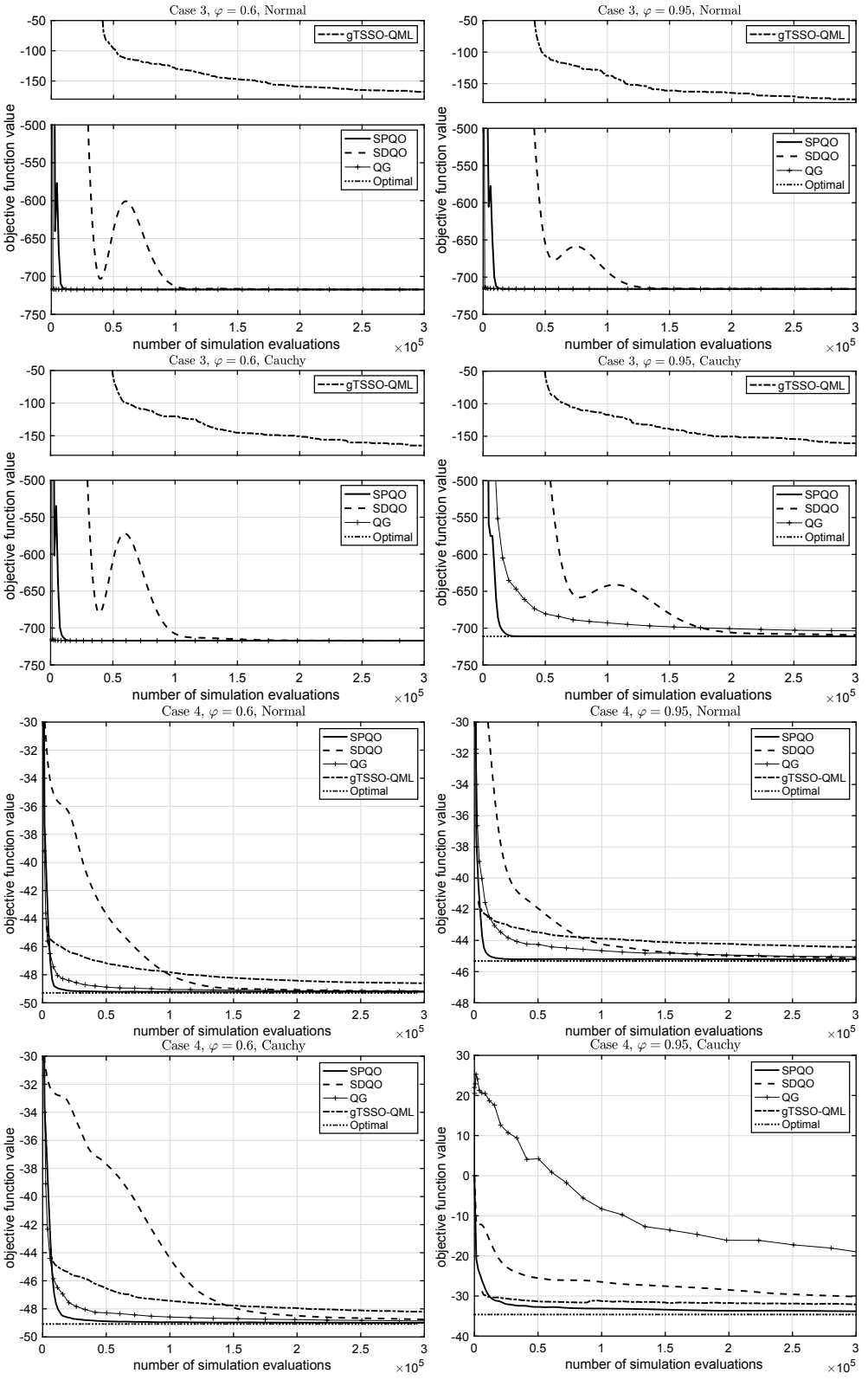
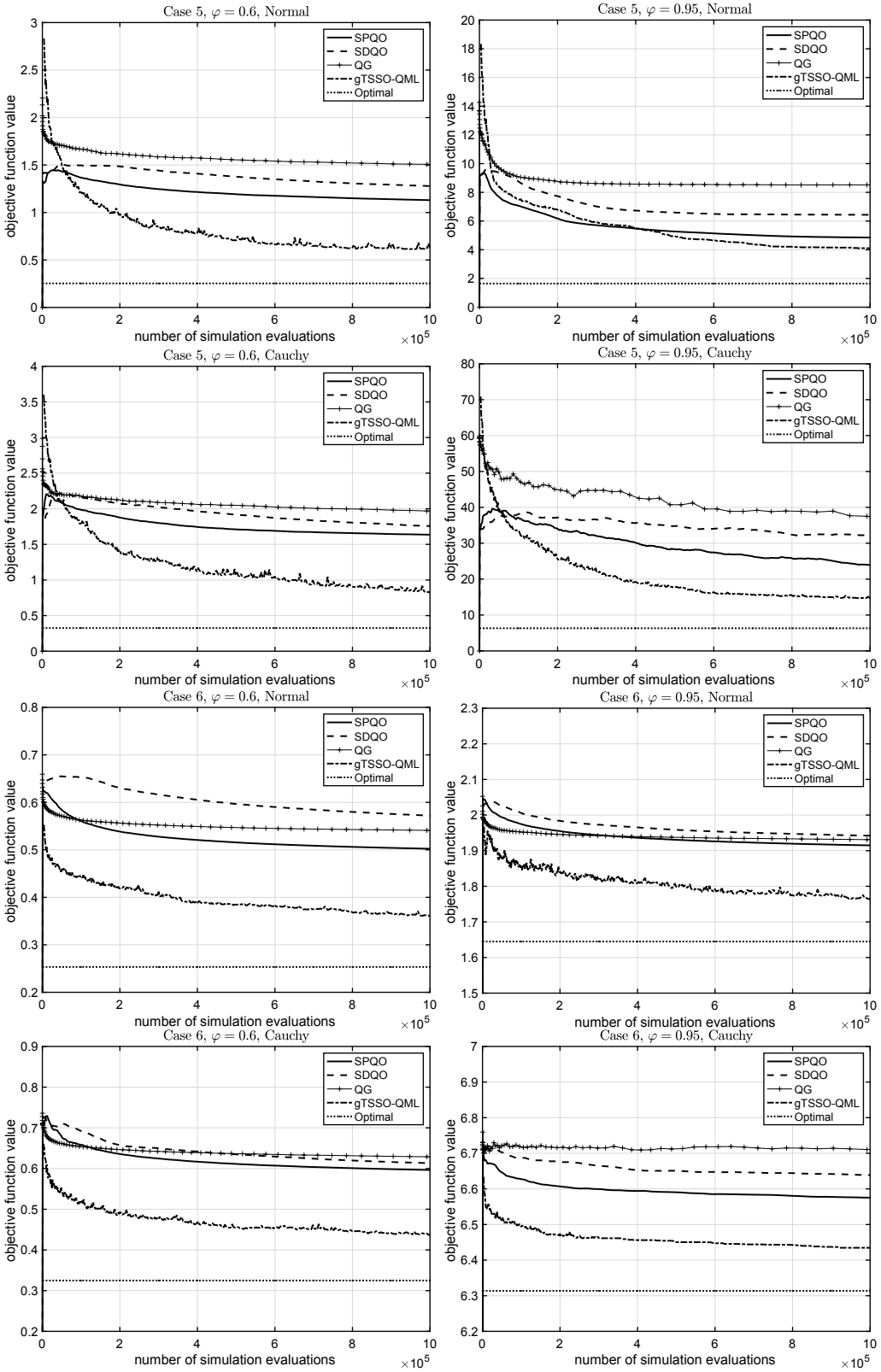


Figure 4. Performance of SPQO, SDQO, QG, and gTSSO-QML on Test Cases 5 and 6



averaging all simulation data collected in past iterations. Thus, it works equally well under the Cauchy noise setting and shows significantly faster convergence behavior than QG.

Functions 5 and 6 are highly multimodal, which makes it very difficult for a gradient-based algorithm to escape local optima. On these test functions, SPQO, SDQO, and QG may quickly get stuck at a local optimum and stop making improvement even during the early search phase (see Figure 4). The gTSSO-QML algorithm, instead, shows more robustness in dealing with local optima and yields much superior performance compared with other algorithms. However, note from Figure 3 that the algorithm is not as efficient on test cases 3 and 4. We conjecture that this is primarily caused by the high dimensionality of these functions so that a close approximation of the true response curve might require a large amount of data that exceeds the given budget. In addition, because each step of gTSSO-QML involves an expensive optimization procedure, the algorithm could be very time-consuming to run on high-dimensional problems. For example, the average running time of gTSSO-QML for solving case 3 on the parallel platform is more than eight hours. In contrast, the execution time of a single run of SPQO on a Windows PC is under five seconds. The above comparison suggests that our proposed algorithms, particularly SPQO, are best suited to high-dimensional differentiable problems that contain few local optimal solutions, whereas gTSSO-QML is better adapted to the optimization of complex multimodal objective functions with relatively small numbers of decision variables.

To provide an illustration of the convergence rate results given in Theorems 3 and 4, Figure 5 shows the log-log plots of the empirical MAEs of SPQO and SDQO estimates (averaged over 100 independent replication runs) versus the number of algorithm iterations for cases 1–4 when $\varphi = 0.6$, where in case 1, we have performed a more extensive experiment by setting the number of algorithm iterations to 10^5. Because of space limitation, the results for the $\varphi = 0.95$ case are reported in Section F of the online appendix. Note that because cases 5 and 6 are highly multimodal problems, whereas our rate results are established assuming a unique (global) minimizer, we have not provided the rate plots for cases 5 and 6. From the figure, we can see that the observed rates of convergence generally conform well to the theoretical results, but in some cases the rates are much faster than the theoretical rate $O(k^{-1/4})$. We conjecture that this is primarily due to the influence of the initial transience, because the algorithm parameters are tuned to yield good finite-time performance when k is small.

5.3. A Queueing Example

We consider a first-come, first-served single-server queue with parameterized service rate $\mu(\theta) = 1/v^T\theta + \lambda$, where $v \in \Re^d$ is a fixed positive vector and λ is the arrival rate. Denoting by $Y(\theta)$ the steady-state waiting time in the system and by $q_\varphi(\theta)$ the corresponding φ-quantile of Y, the objective is to determine an optimal parameter vector θ^* that minimizes the weighted cost of waiting and service given by

$$y(\theta) = c_1 q_\varphi(\theta) + c_2(\theta - \vartheta)^T A(\theta - \vartheta), \tag{32}$$

where $c_1, c_2 > 0$ are cost coefficients, $\vartheta \in \Re^d$ is a nominal vector, and $A \in \Re^{d\times d}$ is a positive definite matrix; these are all assumed known. The cost function (32) reflects the trade-off between decreasing θ to increase the service rate (and hence reduce the waiting time quantile) and choosing θ to make the quadratic penalty term small.

Because of the cost-of-service penalty term, optimizing (32) becomes finding the zeros of $\nabla y(\theta) = 0$ rather than $\nabla q_\varphi(\theta) = 0$, where

$$\nabla y(\theta) = c_1 \nabla q_\varphi(\theta) + 2c_2 A(\theta - \vartheta).$$

Consequently, the three algorithms SPQO, SDQO, and QG are adjusted accordingly to solve this slightly modified root-finding problem. All other steps of the algorithms remain intact. For the simulation experiments, we take i.i.d. exponentially distributed interarrival times and service times, that is, an M/M/1 queue, with $\Theta = [1, 20]^4$, $\lambda = 1$, $v = (0.1, 0.2, 0.3, 0.4)^T$, $c_1 = 0.1$, $c_2 = 0.02$, $\vartheta = (7, 8, 9, 10)^T$,

$$A = \begin{pmatrix} 10 & 2 & 1 & 2 \\ 2 & 9 & 2 & 4 \\ 1 & 2 & 8 & 0 \\ 2 & 4 & 0 & 7 \end{pmatrix},$$

and consider two cases: $\varphi = 0.5$ and $\varphi = 0.95$.

All algorithms except gTSSO-QML are implemented using the same parameter settings as for the black-box test functions in Section 5.2, and the total number of simulation evaluations is set to 1,800. The gTSSO-QML algorithm requires an initial set of design points and additional simulation samples to initialize the kriging model. In our implementation, we set the number of initial design points to 16 and allocate 20 observations to each point to compute its sample quantile. Thus, a total of 320 simulation evaluations are used during the initialization step.

The steady-state waiting time is approximated by the waiting time of the 1,000th customer. Each algorithm is independently repeated 40 times. The simulation results (means and standard errors based on 40 independent runs of each algorithm) obtained on the two test cases are presented in Table 4. For the purpose of comparing with the true optimum, we note that by basic queueing theory, the steady-state waiting time of an M/M/1 queue is exponentially distributed with parameter $\mu(\theta) - \lambda$, so

Figure 5. Convergence Rates of the Empirical MAEs of SPQO and SDQO on Cases 1–4, $\varphi = 0.6$

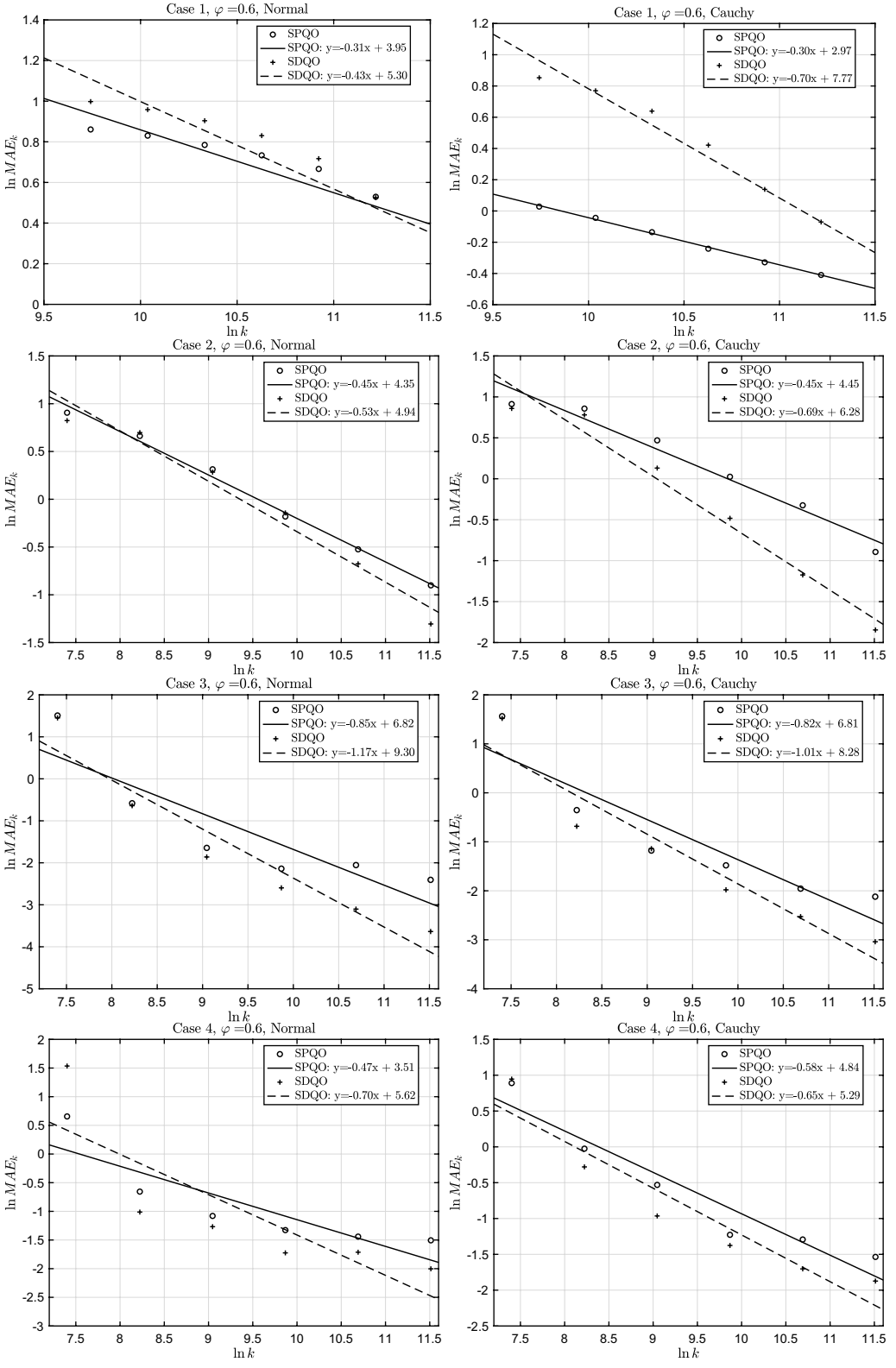


Table 4. Performance on the Queueing Example

	$\varphi = 0.50$	$\varphi = 0.95$
Optimal cost	0.62	2.66
SPQO	0.70 (1.2e-2)	2.78 (1.9e-2)
SPQO-CRN	**0.67 (8.5e-3)**	**2.75 (1.5e-2)**
SDQO	0.72 (1.6e-2)	2.80 (2.0e-2)
SDQO-CRN	0.73 (2.2e-2)	2.78 (1.7e-2)
QG	1.17 (6.6e-2)	3.57 (1.0e-1)
gTSSO-QML	0.87 (2.8e-2)	3.19 (3.7e-2)

Note. Performance based on 40 independent runs (standard errors in parentheses).

the cost function can be calculated in closed form in terms of θ as $y(\theta) = c_1 \ln(1-\varphi) v^T \theta + c_2(\theta - \vartheta)^T A(\theta - \vartheta)$, which attains its minimum at $\theta^* = \vartheta + \frac{c_1}{2c_2} \ln(1-\varphi) A^{-1} v$. In Figure 6, we also plot the true objective function values (averaged over 40 runs) obtained by SPQO, SDQO, QG, and gTSSO-QML as a function of the number of simulation evaluations. The convergence behavior of the empirical MAEs (averaged over 100 independent runs, in log-log scale) of SPQO and SDQO is shown in Figure 7.

The conclusions are generally consistent with the results for the black-box test functions 1–4 in Section 5.2, with SPQO (SPQO-CRN) showing the best performance. SPQO and SDQO outperform QG by a large margin in terms of both mean performance and consistency (standard error). In particular, the large sample size required by QG results in only eight iterations being carried out under the limited budget, whereas the numbers of iterations for SPQO (SPQO-CRN) and SDQO (SDQO-CRN) are 600 and 200, respectively. Moreover, because QG computes new quantile estimates at each iteration independently of past values, the observations collected in previous iterations are discarded, causing inefficient use of simulation data. In contrast, SPQO and SDQO allow the quantile/gradient estimates to be constructed incrementally based on all historical simulation data, and this in turn offers superior finite-sample performance under a limited simulation budget. Note that because this is a unimodal problem and gTSSO-QML does not exploit gradient information, it is not as efficient as SPQO/SDQO. However, the algorithm still outperforms QG, and its performance may be further improved through a more careful tuning of algorithm parameters.

6. Conclusions

For solving quantile optimization problems in the setting of noisy black-box functions, we have proposed two new three-timescale gradient-based SA algorithms. For this quantile BBO setting, there are very few existing algorithms, so these algorithms represent a methodological contribution to the BBO literature. These algorithms can also be applied to (stochastic) simulation optimization problems where direct gradient estimators based on techniques such as perturbation analysis or the likelihood ratio method, which rely on knowledge of the underlying model, are not readily available or are difficult to implement, so the algorithms also advance the state of the art in simulation optimization. The SPQO algorithm is especially promising for high-dimensional problems, requiring only three function evaluations per iterative update. Compared with methods relying on order statistics, the algorithms proposed here have the potential to achieve substantial computational savings. Variants of the algorithms using CRN offer the opportunity for further reductions in the variance of the gradient estimator and hence faster convergence of the algorithms in the simulation optimization setting.

Figure 6. Performance of SPQO, SDQO, QG, and gTSSO-QML on the Queueing Example

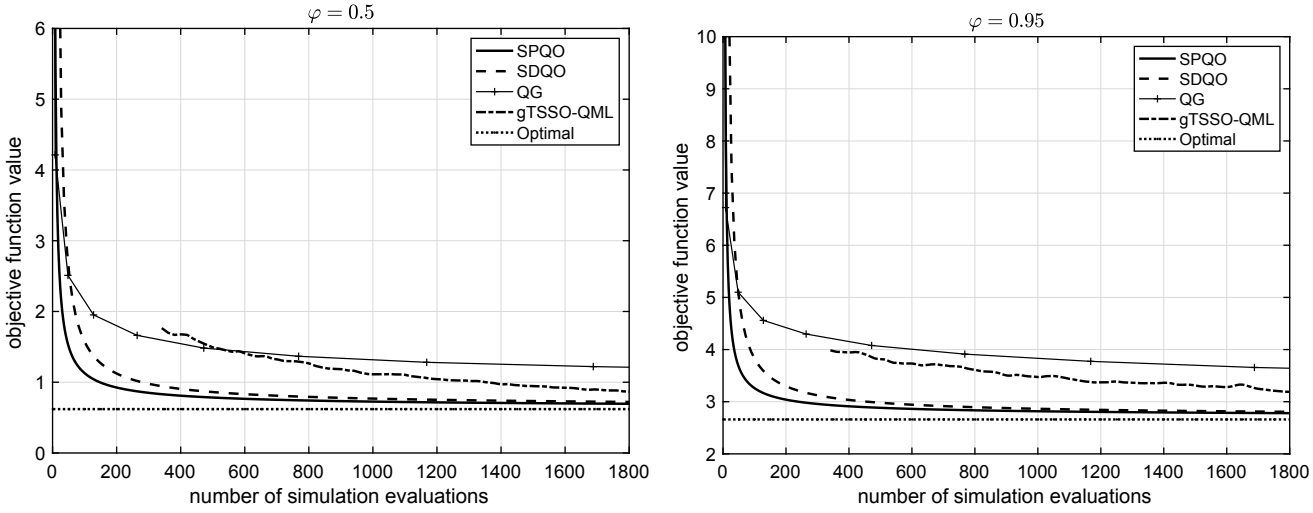
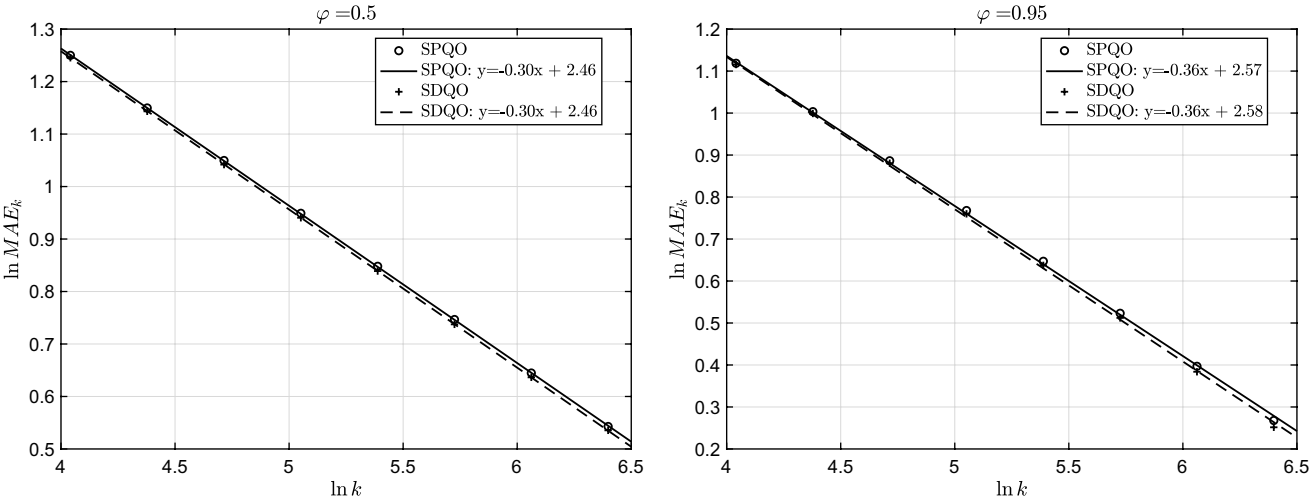


Figure 7. Convergence Rates of the Empirical MAEs of SPQO and SDQO on the Queueing Example



Under the assumption of differentiability of the quantile function and other appropriate conditions, we have analyzed the bias effect of the proposed gradient estimation scheme and established the local convergence of the resultant algorithms. More importantly, through a novel fixed-point argument, we have also provided detailed characterizations of the convergence rates of the quantile and quantile gradient estimators. These results extend existing work in the single-timescale setting and indicate that an upper bound on the MAEs of the algorithms diminishes at the optimal rate $O(k^{-1/4})$. Simulation experiments indicate that the algorithms perform well, and, in particular, SPQO is very promising for solving high-dimensional problems, in terms of the sample size required to achieve reasonable performance.

Future avenues of potential research building on the results here include (i) developing and analyzing other multitimescale algorithms for quantile BBO, such as the two-timescale version alluded to in Section 1, for comparison; (ii) applying the fixed-point approach to study the convergence rate of other multitimescale SA algorithms (or even single-timescale SA algorithms, as in Hu and Fu (2024)); (iii) investigating a more systematic way of tuning the step-size parameters needed to implement multitimescale SA algorithms; and (iv) designing a more comprehensive computational/experimental study to characterize when the multitimescale approach is most effective.

Acknowledgments

The authors thank the editors and three anonymous referees for their helpful comments and suggestions that have led to a substantially improved paper.

References

Audet C, Hare W (2017) *Derivative-Free and Blackbox Optimization*, 1st ed. (Springer, Berlin).

Bhatnagar S (2005) Adaptive multivariate three-timescale stochastic approximation algorithms for simulation based optimization. *ACM Trans. Model. Comput. Simul.* 15(1):74–107.

Borkar VS (2008) *Stochastic Approximation: A Dynamical Systems Viewpoint* (Cambridge University Press, Cambridge, UK).

Bottou L, Curtis FE, Nocedal J (2018) Optimization methods for large-scale machine learning. *SIAM Rev.* 60(2):223–311.

Cao P, Miwa T, Morikawa T (2014) Modeling distribution of travel time in signalized road section using truncated distribution. *Procedia Soc. Behav. Sci.* 138:137–147.

Fabian V (1968) On asymptotic normality in stochastic approximation. *Ann. Math. Statist.* 39(4):1327–1332.

Fu MC (2015) *Handbook of Simulation Optimization* (Springer Science + Business Media, New York).

Fu MC, Hill SD (1997) Optimization of discrete event systems via simultaneous perturbation stochastic approximation. *IIE Trans.* 29(3):233–243.

Fu MC, Hong LJ, Hu JQ (2009) Conditional Monte Carlo estimation of quantile sensitivities. *Management Sci.* 55(12):2019–2027.

Ghadimi S, Lan G (2012) Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization. I: A generic algorithmic framework. *SIAM J. Optim.* 22(4):1469–1492.

Givens GH, Hoeting JA (2013) *Computational Statistics*, 2nd ed. (John Wiley & Sons, Inc., Hoboken, NJ).

Hong YY, Chang HL, Chiu CS (2010) Hour-ahead wind power and speed forecasting using simultaneous perturbation stochastic approximation (SPSA) algorithm and neural network with fuzzy inputs. *Energy* 35(9):3870–3876.

Hu J, Fu MC (2024) Technical note—On the convergence rate of stochastic approximation for gradient-based stochastic optimization. *Oper. Res.*, ePub ahead of print March 8, https://doi.org/10.1287/opre.2023.0055.

Hu J, Peng Y, Zhang G, Zhang Q (2022) A stochastic approximation method for simulation-based quantile optimization. *INFORMS J. Comput.* 34(6):2889–2907.

Kibzun AI, Kurbakovskiy V (1991) Guaranteeing approach to solving quantile optimization problems. *Ann. Oper. Res.* 30(1):81–93.

Kibzun AI, Matveev E (2012) Optimization of the quantile criterion for the convex loss function by a stochastic quasi-gradient algorithm. *Ann. Oper. Res.* 200(1):183–198.

Kibzun AI, Naumov AV, Norkin VI (2013) On reducing a quantile optimization problem with discrete distribution to a mixed integer programming problem. *Autom. Remote Control* 74(6):951–967.

Kiefer J, Wolfowitz J (1952) Stochastic estimation of the maximum of a regression function. *Ann. Math. Statist.* 23(3):462–466.

Kim JH, Powell WB (2011) Quantile optimization for heavy-tailed distributions using asymmetric signum functions. Working paper, Princeton University, Princeton, NJ.

Kleinman NL, Spall JC, Naiman DQ (1999) Simulation-based optimization with stochastic approximation using common random numbers. *Management Sci.* 45(11):1570–1578.

Konda VR, Tsitsiklis JN (2004) Convergence rate of linear two-time-scale stochastic approximation. *Ann. Appl. Probab.* 14(2):796–819.

Kushner HJ, Clark DS (1978) *Stochastic Approximation Methods for Constrained and Unconstrained Systems* (Springer-Verlag, New York).

Kushner HJ, Yin GG (1997) *Stochastic Approximation and Recursive Algorithms and Applications* (Springer, New York).

Law AM (2013) *Simulation Modeling and Analysis*, 5th ed. (McGraw-Hill, New York).

Li MS, Xue HL, Shi F (2017) Optimization of traffic signal parameters based on distribution of link travel time. *J. Central South Univ.* 24(2):432–441.

Mokkadem A, Pelletier M (2006) Convergence rate and averaging of nonlinear two-time-scale stochastic approximation algorithms. *Ann. Appl. Probab.* 16(3):1671–1702.

Rugh WJ (1996) *Linear System Theory*, 2nd ed. (Prentice Hall, Upper Saddle River, NJ).

Sabater C, Le Maître O, Congedo PM, Görtz S (2021) A Bayesian approach for quantile optimization problems with high-dimensional uncertainty sources. *Comput. Methods Appl. Mech. Engrg.* 376:113632.

Song M, Hu J, Fu MC (2023) Simultaneous perturbation-based stochastic approximation for quantile optimization. Corlu CG, Hunter SR, Lam H, Onggo BS, Shortle J, Biller B, eds. *Proc. 2023 Winter Simulation Conf.* (IEEE Press, Piscataway, NJ), 3565–3576.

Spall JC (1992) Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Automat. Control* 37(3):332–341.

Spall JC (2003) *Introduction to Stochastic Search and Optimization* (John Wiley & Sons, Hoboken, NJ).

Spall JC, Chin DC (1997) Traffic-responsive signal timing for system-wide traffic control. *Transportation Res. Part C Emerging Tech.* 5(3–4):153–163.

Spall JC, Cristion JA (1997) A neural network controller for systems with unmodeled dynamics with applications to wastewater treatment. *IEEE Trans. Systems Man Cybernetics Part B Cybernetics* 27(3):369–375.

Vasiléva SN, Kan YS (2015) A method for solving quantile optimization problems with a bilinear loss function. *Autom. Remote Control* 76(9):1582–1597.

Wang S, Ng SH, Haskell WB (2021) A multilevel simulation optimization approach for quantile functions. *INFORMS J. Comput.* 34(1):569–585.

Zamar DS, Bhushan Gopaluni R, Sokhansanj S, Newlands NK (2017) A quantile-based scenario analysis approach to biomass supply chain optimization under uncertainty. *Comput. Chem. Engrg.* 97(2):114–123.

Zhang Q, Hu J (2019) Simulation optimization using multi-time-scale adaptive random search. *Asia-Pac. J. Oper. Res.* 36(6): 1940014.

Jiaqiao Hu is an associate professor in the Department of Applied Mathematics and Statistics at the State University of New York, Stony Brook. His research interests include Markov decision processes, simulation-based optimization, stochastic modeling and analysis, and computational learning theory.

Meichen Song is a PhD student in the Department of Applied Mathematics and Statistics at the State University of New York, Stony Brook. She received a BS degree in mathematics from Southeast University, Nanjing, China. Her research interests are simulation optimization, quantile optimization, and stochastic approximation.

Michael C. Fu holds the Smith Chair of Management Science in the Robert H. Smith School of Business, with a joint appointment in the Institute for Systems Research and affiliate faculty appointment in the Department of Electrical and Computer Engineering, all at the University of Maryland, College Park. His research interests include Markov decision processes, stochastic gradient estimation, simulation optimization, and applied probability. He is a Fellow of INFORMS and IEEE.