POLITECNICO DI TORINO
Repository ISTITUZIONALE

An adaptive importance sampling algorithm for risk-averse optimization

Original
An adaptive importance sampling algorithm for risk-averse optimization / Pieraccini, S., Vanzan, T.. - In: JOURNAL OF COMPUTATIONAL PHYSICS. - ISSN 0021-9991. - 547:(2026), pp. 1-16. [10.1016/j.jcp.2025.114548]

Availability:
This version is available at: 11583/3005603 since: 2025-12-03T07:36:44Z

Publisher:
Elsevier

Published
DOI:10.1016/j.jcp.2025.114548

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

18 November 2026

Journal of Computational Physics 547 (2026) 114548

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An adaptive importance sampling algorithm for risk-averse optimization

Sandra Pieraccini, Tommaso Vanzan *

Dipartimento di Scienze Matematiche, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy

ARTICLE INFO

Keywords:
Stochastic optimization
Adaptive sampling
Risk-averse minimization
PDE-constrained optimization under uncertainty
Reduced order models

ABSTRACT

Adaptive sampling algorithms are modern and efficient methods that dynamically adjust the sample size throughout the optimization process. However, they may encounter difficulties in risk-averse settings, particularly due to the challenge of accurately sampling from the tails of the underlying distribution of random inputs. This often leads to a much faster growth of the sample size compared to risk-neutral problems. In this work, we propose a novel adaptive sampling algorithm that adapts both the sample size and the sampling distribution at each iteration. The biasing distributions are constructed on the fly, leveraging a reduced-order model of the objective function to be minimized, and are designed to oversample a so-called risk region. As a result, a reduction of the variance of the gradients is achieved, which permits to use fewer samples per iteration compared to a standard algorithm, while still preserving the asymptotic convergence rate. Our focus is on the minimization of the Conditional Value-at-Risk (CVaR), and we establish the convergence of the proposed computational framework. Numerical experiments confirm the substantial computational savings achieved by our approach.

1. Introduction

In this manuscript, we consider the minimization of

$$\min_{z\in\mathcal{C}} \left\{ \widehat{F}(z) := \mathrm{CVaR}_\beta[f(z,\boldsymbol{\xi})] \right\}, \tag{1}$$

where z is a decision variable, $\mathcal{C}$ is a convex and closed subset of a Hilbert space $(\mathcal{Z},(\cdot,\cdot))$, $\boldsymbol{\xi}$ is a random variable over a probability space $(\Omega,\mathcal{F},\mathbb{P})$, with values on a subset $\Gamma\subset\mathbb{R}^N$, $N\in\mathbb{N}^+$, and with probability density ρ. The quantity of interest is $f:\mathcal{Z}\times\Gamma\to\mathbb{R}$ which is supposed to be smooth and whose gradient with respect to z is denoted by $\nabla f(z,\boldsymbol{\xi})$. CVaR_β is the so-called Conditional Value-at-Risk (CVaR), namely the expectation of $f(z,\cdot)$ conditioned on being above its β quantile. If CVaR_β is replaced by the expectation operator $\mathbb{E}_\rho$, (1) has found numerous applications in different fields such as, e.g., machine learning [1,2], portfolio optimization [3,4], stochastic programming [5,6] and PDE-constrained optimization with random inputs [7–9]. More recently, minimization of the CVaR_β has drawn interest since it is often desirable to be robust against risk-averse events; see, e.g., Rockafellar et al. [10], Rockafellar and Uryasev [11], Kouri and Surowiec [12], Curi et al. [13].

The numerical solution of (1) with $\mathbb{E}_\rho$ has traditionally been dealt with two main classes of algorithms. The first family, called Stochastic Approximation (SA) methods [6, Chapter 5.9], includes iterative methods that at each iteration draw new realizations of $\boldsymbol{\xi}$ independent from the previous ones and perform one step of a descent algorithm. Examples are the stochastic gradient method

* Corresponding author.
E-mail addresses: sandra.pieraccini@polito.it (S. Pieraccini), tommaso.vanzan@polito.it (T. Vanzan).

https://doi.org/10.1016/j.jcp.2025.114548
Received 14 February 2025; Received in revised form 18 September 2025; Accepted 24 November 2025
Available online 27 November 2025
0021-9991/© 2025 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

and its variants; see, e.g., Bottou et al. [1] and references therein. These algorithms show a low cost per iteration and, under suitable assumption, converge to the minimizer $z^\star$ of (1), but at a sublinear rate. The second family, called Sample Average Approximation (SAA) in stochastic optimization communities or simply batch methods in machine learning, consists in replacing $\mathbb{P}$, once and for all, with a discrete empirical probability distribution $\mathbb{P}_M$, obtained by sampling M realizations of $\boldsymbol{\xi}$. The approximated problem can then be solved with standard nonlinear optimization algorithms that often converge very rapidly, but only to a minimizer $z^\star_M \neq z^\star$. Between these two classes, there are attractive modern strategies such as adaptive sampling methods (see, e.g., Byrd et al. [14], Friedlander and Schmidt [15], Bollapragada et al. [16], Beiser et al. [17], Xie et al. [18]) which aim to improve the efficiency of SA methods by dynamically adjusting the sample size along the optimization. The driving idea is to use cheap, but noisy, gradient evaluations far from the optimum, while more accurate and expensive estimates are computed close to the minimizer. Earlier works showed that increasing the sample size geometrically is sufficient to preserve linear convergence of a gradient descent algorithm, while a few implementable rules have been now derived to *adapt* the sample size based on current optimization progresses, both in the unconstrained and constrained settings.

Compared to $\mathbb{E}_\rho$, both the evaluation and optimization of CVaR_β raise additional challenges. First, evaluating $\mathrm{CVaR}_\beta[f(z,\boldsymbol{\xi})]$ requires to sample from the tail of the unknown distribution of $f(z,\cdot)$, which implies that a large number of samples are needed to have sufficiently accurate estimates. Second, $\nabla f(z,\boldsymbol{\xi})$ actually contributes to the approximation of $\nabla \widehat{F}(z)$ only for samples $\boldsymbol{\xi}$ belonging to a so-called *risk-region*, defined as the set of $\boldsymbol{\xi}$ such that $f(z,\boldsymbol{\xi}) \geq \mathrm{VaR}_\beta[f(z,\boldsymbol{\xi})]$, $\mathrm{VaR}_\beta[f(z,\boldsymbol{\xi})]$ being the Value-at-Risk of level β, that is, its β quantile. Thus a significant computational effort is wasted if evaluating $f(z,\boldsymbol{\xi})$ is expensive, as one may compute $f(z,\boldsymbol{\xi})$ and then realize that the computation of $\nabla f(z,\boldsymbol{\xi})$ is not needed. This is typically the case in PDE-constrained optimization under uncertainty, where evaluating $f(z,\boldsymbol{\xi})$ consists in solving a PDE state equation, while $\nabla f(z,\boldsymbol{\xi})$ entails the solution of an additional PDE adjoint equation. Third, adaptive sampling algorithms commonly adjust the sample size along the optimization to control the variance of $\boldsymbol{\xi} \to \nabla f(z,\boldsymbol{\xi})$. In risk-averse problems, the variance of the stochastic gradients typically grows as β increases, leading to larger sample sizes, hence making risk-averse optimization significantly more expensive compared to the risk-neutral case.

1.1. Contributions and related works

The primary contribution of this manuscript is the derivation of a new adaptive sampling algorithm that adjusts the sampling distribution along the optimization. The algorithm generates a sequence of biasing densities $\{\widetilde{\rho}_k\}_{k\in\mathbb{N}}$ that are adapted to the sequence of iterates $\{z_k\}_{k\in\mathbb{N}}$, so that each $\widetilde{\rho}_k$ oversamples the current risk-region, and thus most of the sample do contain valuable gradient information. This leads to an importance sampling estimation of the gradient $\nabla \widehat{F}(z)$, and to a variance reduction which permits then to take smaller sample sizes compared to standard adaptive sampling algorithms, eventually reducing the overall computational cost. Our algorithm *learns* the biasing distributions on the fly during the optimization, relying on the availability of a reduced order model (ROM) of the map $\boldsymbol{\xi} \to f(z_k,\boldsymbol{\xi})$, denoted by $f_r(z_k,\cdot)$, at every iterate z_k. The ROM is then used both to estimate the current $\mathrm{VaR}_\beta\left[f(z_k,\cdot)\right]$ and to generate the new biased samples using an acceptance-rejection algorithm. The construction and availability of such a ROM is of course strongly problem dependent. We here present numerical experiments based on an application to PDE-constrained optimization under uncertainty, and discuss how these reduced models can be efficiently built online, during the optimization, at a reduced computational cost. A secondary contribution of this manuscript lies in a convergence analysis of an alternating optimization algorithm tailored for (1). Indeed, while [17] presents an alternating procedure without discussing its convergence, Ganesh and Nobile [19] presents a rigorous convergence analysis under however some quite restrictive hypothesis. We here propose an alternative path to prove convergence under some relaxed assumptions.

Besides the above mentioned references on adaptive sampling algorithms, we wish to emphasize a couple of manuscripts that are more closely related to this work. In machine learning, Curi et al. [13] precisely addresses the challenges of the CVaR_β optimization using adaptive sampling algorithm, and similarly proposes to adjust the sampling distribution along the optimization. However, the construction of the biasing distribution differs and it is based on a dual reformulation of (1) as a distributionally robust problem, leading to a min-max game between two players. Closely related to this work are also [20,21], which propose two approaches to use ROMs for the evaluation of the CVaR_β. The first approach consists in simply replacing the full order model with a ROM, while the second suggests to use a ROM to derive a biasing distribution from which then sampling the full order model. The first approach has been applied to the CVaR_β minimization within SAA algorithms in [22,23]. In contrast, as the importance sampling approach cannot be embedded into SAA algorithms (since the sample set is fixed in these algorithms), the second approach has been less investigated. The present manuscript, which combines importance sampling with adaptive sampling algorithms, can be viewed as an extension of the second approach of [20] to the optimization context.

The manuscript is structured as follows. Section 2 introduces the model and a classical reformulation of (1) as a double minimization problem. Then, an alternating optimization procedure is recalled and Section 2.3 discusses in details the challenges of adaptive sampling algorithms for risk-averse minimization. Section 3 recalls the importance sampling framework, discusses the ROM-based construction of the biasing densities, and present the adaptive importance sampling algorithm. Section 3.4 proposes a convergence analysis of the alternating procedure for the interested readers. Section 4 presents numerical experiments aimed at showing the better complexity of the new approach and details the dynamic update of the ROM during the optimization.

2. Problem setting

In this section, we recall an equivalent formulation of (1) that is the starting point of several approaches to optimize CVaR_β [10,24,25]. We then describe an adaptive sampling algorithm to minimize (1), and highlight a few challenges that will be addressed by our contributions in Section 3.

2.1. Conditional value-at-risk

For a given random variable $X \in L^1_\rho(\Gamma;\mathbb{R})$, the Value-at-Risk of confidence level $0 < \beta < 1$ (also known as β quantile) is defined as

$$\mathrm{VaR}_\beta[X] := \inf_{t\in\mathbb{R}} \{\mathbb{P}(X \le t) \ge \beta\},$$

and its existence is guaranteed by the right continuity of the cumulative distribution function. The Conditional Value-at-Risk (also called expected shortfall or average Value-at-Risk) of confidence level β is the expectation of X conditioned on being larger than $\mathrm{VaR}_\beta[X]$,

$$\mathrm{CVaR}_\beta[X] := \mathbb{E}_\rho\big[X | X \ge \mathrm{VaR}_\beta[X]\big].$$

The Conditional Value-at-Risk has gained popularity as risk measure since, in contrast to the Value-at-Risk, it embeds information on the average loss above the critical β quantile. In [10,26], Rockafellar and Uryasev proved that $\mathrm{CVaR}_\beta[X]$ admits the equivalent formulation

$$\mathrm{CVaR}_\beta[X] = \inf_{t\in\mathbb{R}} \left\{ t + \frac{1}{1-\beta}\mathbb{E}_\rho\big[(X-t)^+\big] \right\}, \tag{2}$$

where $(\cdot)^+ := \max(0,\cdot)$, and that the infimum (2) is attained, although possibly on a whole bounded interval of the real line $\mathcal{A}_\beta[X]$ [1], with $\mathrm{VaR}_\beta[X] = \min_t \{t \in \mathcal{A}_\beta[X]\}$.

This result allows us to reformulate (1) as the double minimization

$$\min_{z\in C,\, t\in\mathbb{R}} \left\{ \widetilde{F}(z,t) := t + \frac{1}{1-\beta}\mathbb{E}_\rho\big[(f(z,\boldsymbol{\xi})-t)^+\big] \right\}, \tag{3}$$

which enjoys attractive features: on the one hand, solving (3) permits to both optimize $\mathrm{CVaR}_\beta[f]$ and simultaneously get the associated $\mathrm{VaR}_\beta\big[f(z^\star,\cdot)\big]$, see, e.g., Rockafellar and Uryasev [26, Corollary 15]. On the other hand, (3) involves the minimization of an expectation, and thus either SAA or SA methods could be readily applied.

Nevertheless (3) involves the function $(\cdot)^+$ which is only Lipschitz continuous. Despite $\widetilde{F}$ may be jointly Fréchet differentiable with respect to z and t due to the smoothing effect of $\mathbb{E}_\rho$ [2] [19, Theorem 2.1], $\widetilde{F}$ is non-differentiable whenever the continuous expectation is replaced by an empirical approximation (as in SAA methods), while $(f(z,\boldsymbol{\xi})-t)^+$ does not admit a classical gradient (which prevents gradient-based SA methods). In order to use optimization algorithms for smooth functions, Kouri and Surowiec [24] suggested to replace $(\cdot)^+$ with a smooth approximation g_ε. In this work, we choose the C^2 differentiable function

$$g_\varepsilon = \begin{cases} 0 & \text{if } x \le -\frac{\varepsilon}{2}, \\ \frac{(x-\frac{\varepsilon}{2})^3}{\varepsilon^2} - \frac{(x-\frac{\varepsilon}{2})^4}{2\varepsilon^3} & \text{if } x \in (-\frac{\varepsilon}{2}, \frac{\varepsilon}{2}), \\ x & \text{if } x \ge \frac{\varepsilon}{2}, \end{cases}$$

and remark two properties that will be used in the following, namely that $\sup_{t\in\mathbb{R}} |g'_\varepsilon(t)| \le 1$ and $g''_\varepsilon(t) \ge 0$, for all $t \in \mathbb{R}$. From now on, we consider the smoothed version of (3),

$$\min_{z\in C,\, t\in\mathbb{R}} \left\{ \widetilde{F}^\varepsilon(x,t) := t + \frac{1}{1-\beta}\mathbb{E}_\rho\big[g_\varepsilon(f(z,\boldsymbol{\xi})-t)\big] \right\}. \tag{4}$$

A useful guide to choose a suitable value of ε is the error estimate [24, Theorem 4.13],

$$(|t^\star - t^\star_\varepsilon|^2 + \|z^\star - z^\star_\varepsilon\|^2)^{\frac{1}{2}} \le C\sqrt{\varepsilon},$$

where $(z^\star, t^\star)$ and $(z^\star_\varepsilon, t^\star_\varepsilon)$ are minimizers of (3) and (4), respectively, and C is a positive constant independent of $(z^\star, t^\star)$. Concretely, ε should be carefully tuned via a tradeoff between a desired relative error, and the conditioning of (4) which deteriorates as $\varepsilon \to 0$.

The double minimization formulation paves the way to a family of algorithms that alternate between exact/inexact minimization with respect to x and t, see [17,19,23,25]. A particular instance is the procedure

$$t_k \in \mathrm{argmin}_{t\in\mathbb{R}} \widetilde{F}^\varepsilon(z_k,t), \tag{5}$$

$$z_{k+1} = z_k + \alpha_k d_k, \tag{6}$$

[1] $\mathcal{A}_\beta[X]$ is not a singleton when the cumulative distribution function is flat in a right neighborhood of $\mathrm{VaR}_\beta[X]$, see [26, Section 2].

[2] Sufficient conditions are the differentiability of f with respect to z and that the distribution of $f(z,\cdot)$ does not have atoms.

where d_k is an appropriate direction, hopefully descent for $\widetilde{F}^\epsilon$ at (z_k, t_k). Notice that (5) consists in a simple one-dimensional optimization problem which can be solved either using some statistical library to compute quantiles or through, e.g., a bisection algorithm applied to $\nabla_t \widetilde{F}^\epsilon$. In our numerical implementation, we followed the latter approach. Instead, for a fixed t, the direction d_k can be computed with any optimization algorithm mentioned in the introduction, see, e.g., Beiser et al. [17], Ganesh and Nobile [19] for adaptive sampling approaches which we recall in the next subsection.

2.2. An adaptive sampling algorithm for risk-neutral optimization

A traditional mini-batch stochastic gradient algorithm applied to the unconstrained minimization of $F(z) := \mathbb{E}[f(z,\boldsymbol{\xi})]$ computes at iteration $k = 0, \dots,$

$$z_{k+1} = z_k - \alpha_k \frac{1}{M} \sum_{j=1}^{M} \nabla f(z_k, \boldsymbol{\xi}_j^k),$$

where $\left\{\boldsymbol{\xi}_j^k\right\}_{j=1}^{M}$, $M \in \mathbb{R}$, are independent realizations sampled according to ρ. This generally leads to noisy gradient estimates, which are then mitigated by a decreasing step size α_k, ultimately responsible for the slow convergence. In contrast, adaptive sampling methods reduce the variance of the stochastic gradients by dynamically increase the sample size along the optimization while using a fixed step size. An iteration reads

$$z_{k+1} = z_k - \alpha \nabla F_{S_k}(z_k) = z_k - \alpha \frac{1}{M_k} \sum_{j=1}^{M_k} \nabla f(z_k, \boldsymbol{\xi}_j^k),$$

where $F_{S_k}(z_k) := \frac{1}{M_k} \sum_{j=1}^{M_k} f(z_k, \boldsymbol{\xi}_j^k)$, $S_k = \left\{\boldsymbol{\xi}_j^k\right\}_{j=1}^{M_k}$ denotes the set of i.i.d. realizations of $\boldsymbol{\xi}$ drawn at iteration k, and α is now a fixed-constant suitably tuned.

Starting from [15, Theorem 3.1], which showed that growing the sample size geometrically, i.e. $M_k = \tau^k$ for $\tau > 1$, permits to recover the linear convergence of gradient descent with fixed step length for the unconstrained minimization of strongly convex functions, several contributions have then developed criteria to adapt the sample size based on the *actual* progress of the unconstrained optimization (and thus not on a preset rule), see [14,16,27,28]. In particular, let $\mathbb{E}[\cdot]$ denote the full expectation with respect to the distribution of all sample sets $\left\{S_k\right\}_{k\geq 1}$, introduce the filtration $\mathcal{F}_k = \sigma(S_i,\ i < k)$, that is the sigma algebra generated by all samples $\left\{\boldsymbol{\xi}_j^i\right\}_{j=1}^{M_i}$ used up to iteration $k-1$, and the shorthand notation $\mathbb{E}_k[\cdot]$ for the conditional expectation $\mathbb{E}\left[\cdot|\mathcal{F}_k\right]$. Hence, for instance $\mathbb{E}_k\left[F_{S_k}(z_k)\right]$ is well defined and corresponds to the expectation of $F_{S_k}(z_k)$ with respect only to the distribution of the samples $\left\{\boldsymbol{\xi}_j^k\right\}_{j=1}^{M_k}$ drawn at iteration k. The authors of [14] showed that imposing at each iteration that

$$\mathbb{E}_k\left[\|\nabla F_{S_k}(z_k) - \nabla F(z_k)\|^2\right] = \frac{\mathbb{E}_\rho\left[\|\nabla f(z_k,\cdot) - \nabla F(z_k)\|^2\right]}{M_k} = \frac{\mathrm{Var}_\rho\left[\nabla f(z_k,\cdot)\right]}{M_k} \leq \theta^2 \|\nabla F(z_k)\|^2, \tag{7}$$

is sufficient to preserve the linear convergence of the descent algorithm. In words, (7) requires that the variance of the estimated gradient $\nabla F_{S_k}(z_k)$ decreases proportionally to the norm square of the exact gradient evaluated at the current iteration z_k.

In practice, imposing (7) at each iteration may be inconvenient, as it would require to sequentially add new samples to S_k until the criterion is satisfied. Furthermore, the exact gradient is not an available quantity and thus it is approximated with the information available at iteration k, namely the empirical gradient. A common approach, often called *norm test*, is then to use (7) to estimate the *next* sample size by computing

$$\varrho = \frac{\sum_{\boldsymbol{\xi}_j^k \in S_k} \|\nabla f(z_k, \boldsymbol{\xi}_j^k) - \nabla F_{S_k}(z_k)\|^2}{(M_k - 1)(M_k)\theta^2 \|\nabla F_{S_k}(z_k)\|^2}, \tag{8}$$

and set $M_{k+1} = \varrho M_k$ if $\varrho > 1$. The factor $M_k - 1$ in (8) is due to the unbiased estimator of $\mathrm{Var}_\rho\left[\nabla f(z_k,\cdot)\right]$. For convex constrained optimization problems, (7) is not appropriate since ∇F may not vanish at the optimum. Nevertheless, [17,18] proved that the alternative condition

$$\mathbb{E}_k\left[\|\nabla F_{S_k}(z_k) - \nabla F(z_k)\|^2\right] \leq \theta^2 \|R_{S_k}(z_k)\|^2 \tag{9}$$

where $R_{S_k}(z_k) := \frac{1}{\alpha}(z_{k+1} - z_k)$ is sufficient to guarantee linear convergence of the iterates.

2.3. Challenges associated to the CVaR$_\beta$ optimization

In the context of the CVaR$_\beta$ minimization, an adaptive sampling algorithm has been first proposed in [17] (see also [19] for a slightly different version), and it is recalled in Algorithm 1. Here, $\mathcal{P} : \mathcal{Z} \to C$, defined as $\mathcal{P}(z) = \mathrm{argmin}_{v\in C}\|z - v\|_{\mathcal{Z}}$, denotes the projection operator onto the convex set C. Algorithm 1 consists of an alternating optimization procedure. At every iteration k, a new set of i.i.d. samples S_k is drawn. Then, the approximated function

$$\widetilde{F}^\epsilon_{S_k}(z,t) := t + \frac{1}{(1-\beta)M_k} \sum_{j=1}^{M_k} g_\epsilon(f(z,\boldsymbol{\xi}_j) - t)$$

is first exactly minimized with respect to the t variable, and then a gradient step is performed with respect to z. Finally, the norm test (direct extension of (8))

$$\varrho = \frac{\sum_{\xi_j^k \in S_k} \left\| \frac{g_\varepsilon'(f(z_k,\xi_j^k)-t_k)\nabla f(z_k,\xi_j^k)}{1-\beta} - \nabla \widetilde{F}^\varepsilon_{S_k}(z_k,t_k) \right\|^2}{(M_k-1)(M_k)\theta^2 \| R_{S_k}(z_k)\|^2} \tag{10}$$

decides whether to increase the sample size. The symbol ∇ still denotes the gradient with respect only to the variable z . We highlight that at every iteration of Algorithm 1, t_k represents an approximation of $\mathrm{VaR}_\beta\left[f(z_k,\cdot)\right]$ computed with the current M_k samples. This property linking z_k and t_k will come at hand in the coming discussion.

Algorithm 1 Adaptive sampling algorithm for CVaR_β minimization.

Require $\mathbf{z}^0$, M_0, α.
Set $k = 0$.
Sample $S_0 = \left\{\xi_i^k\right\}_{i=1}^{M_0}$ from ρ
while Convergence not reached **do**
  $t_k \in \mathrm{argmin}_{t\in\mathbb{R}} \widetilde{F}^\varepsilon_{S_k}(z_k,t)$.
  $z_{k+1} = \mathcal{P}\left(z_k - \alpha \nabla \widetilde{F}^\varepsilon_{S_k}(z_k,t_k)\right)$.
  if Test (9) is not satisfied **then**
    $M_{k+1} = \varrho M_k$, with ϱ as in (10)
    Sample $S_{k+1} = \left\{\xi_i^{k+1}\right\}_{i=1}^{M_{k+1}}$ from ρ.
  end if
end while

In order to illustrate the computational challenges of Algorithm 1, we start by re-interpreting the quantities that are involved in the CVaR_β minimization and in the norm test. We first define the event

$$G_{z,t} := \{\xi : \ f(z,\xi) \geq t\},$$

and observe that, if $t = \mathrm{VaR}_\beta[f(z,\cdot)]$, then $\mathbb{P}(G_{z,t}) = 1-\beta$ and the gradient of $\widetilde{F}^\varepsilon$ satisfies

$$\nabla \widetilde{F}^\varepsilon(z,t) = \mathbb{E}\left[\frac{\nabla g_\varepsilon(f(z,\cdot)-t)}{1-\beta}\right] = \mathbb{E}\left[\frac{g_\varepsilon'(f(z,\cdot)-t)\nabla f(z,\cdot)}{1-\beta}\right] \approx \mathbb{E}\left[\nabla f(z,\cdot)|G_{z,t}\right], \tag{11}$$

i.e., $\nabla \widetilde{F}^\varepsilon(z,t)$ is an approximation (since $g_\varepsilon'(f(z,\xi)-t)$ is a smoothed version of the characteristic function $\mathbb{1}_{G_{z,t}}$) of the expected value of $\nabla f(z,\cdot)$ conditioned on the event $G_{z,t}$. To approximate $\nabla \widetilde{F}^\varepsilon$ at each iteration, Algorithm 1 considers the function $\widetilde{F}^\varepsilon_{S_k}$ evaluated at the current iterate (z_k,t_k), where t_k is a sample-based approximation of $\mathrm{VaR}_\beta\left[f(z_k,\cdot)\right]$. Consequently, $\nabla \widetilde{F}^\varepsilon(z_k,t_k)$ is estimated by the empirical average

$$\nabla \widetilde{F}^\varepsilon_{S_k}(z_k,t_k) = \frac{1}{(1-\beta)M_k} \sum_{\xi_j^k \in S_k} g_\varepsilon'(f(z_k,\xi_j^k)-t_k)\nabla f(z_k,\xi_j^k).$$

One first challenge is then due to term $g_\varepsilon'(f(z_k,\cdot)-t_k)$, since only very few samples actually contribute to the gradient estimation, namely those belonging to the so-called risk-region

$$G^\varepsilon_{z_k,t_k}[f] := \left\{\xi : \ g_\varepsilon'(f(z_k,\xi)-t_k) > 0\right\} = \left\{\xi : \ f(z_k,\xi) \geq t_k - \frac{\varepsilon}{2}\right\}.$$

For small values of ε, approximately a fraction $1-\beta$ of the samples belongs to $G^\varepsilon_{z_k,t_k}$ at every iteration, and this leads to a great waste of computational resources to get accurate gradient estimations.

Given this consideration, one might infer that Algorithm 1 would require a sequence of sample sizes that are approximately $\frac{1}{1-\beta}$ larger than the sample sizes used to minimize the risk-neutral objective functional $\mathbb{E}[f(z,\cdot)]$. Unfortunately, this is not the case, since Algorithm 1 sets the sample size according to the *variance* of the gradients, and thus it may lead to much larger values of M_k. To understand this, we analyze the theoretical version of the norm test (7),

$$\mathrm{Var}_k\left[\nabla \widetilde{F}^\varepsilon_{S_k}(z_k,t_k)\right] = \frac{\mathrm{Var}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]}{M_k} \leq \theta^2 \|\nabla \widetilde{F}^\varepsilon(z_k,t_k)\|^2,$$

and focus on the term $\mathrm{Var}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]$. A direct calculation shows that

$$\begin{aligned}\mathrm{Var}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right] &= \mathbb{E}_\rho\left[\left\|\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}-\mathbb{E}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]\right\|^2\right]\\ &= \mathbb{E}_\rho\left[\left\|\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right\|^2\right]-\left\|\mathbb{E}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]\right\|^2\\ &\approx \frac{1}{1-\beta}\mathbb{E}_\rho\left[\|\nabla f(z_k,\cdot)\|^2|G_{z_k,t_k}\right]-\left\|\mathbb{E}_\rho\left[\nabla f(z_k,\cdot)|G_{z_k,t_k}\right]\right\|^2.\end{aligned} \tag{12}$$

Hence, $\mathrm{Var}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]$ does not correspond to the variance of the gradient $\nabla f(z_k,\cdot)$ conditioned on the event G_{z_k,t_k}, as one may have intuitively guessed since the sampling is not restricted neither to G_{z_k,t_k} or to its smoothed version $G^\varepsilon_{z_k,t_k}$. On the contrary, as $\beta\to 1$, $\mathrm{Var}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]$ is dominated by $\frac{1}{1-\beta}\mathbb{E}_\rho\left[\|\nabla f(z_k,\cdot)\|^2|G_{z_k,t_k}\right]$, i.e., the second moment of the gradient conditioned to the risk-region. As we will show in Section 4, $\mathrm{Var}_\rho\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]$ can be orders of magnitude larger than $\mathrm{Var}_\rho\left[\nabla f(z_k,\cdot)\right]$, thus leading to much larger values of M_k compared to the risk-neutral setting.

In the next section, we propose to dynamically construct biasing densities $\widetilde{\rho}_k$ from which sampling the set S_k, with the goal of reducing the right hand side of (12), and thus use smaller sets S_k at each iteration. As a matter of fact, we will show that the variance of the gradients with respect to the adapted biasing distribution satisfies

$$\mathrm{Var}_{\widetilde{\rho}_k}\left[\frac{g_\varepsilon'(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)}{1-\beta}\right]\approx \mathrm{Var}_\rho\left[\nabla f(z_k,\cdot)|G_{z_k,t_k}\right], \tag{13}$$

These biasing densities will be dynamically constructed along the optimization and will be adapted to the current risk-region $G^\varepsilon_{z_k,t_k}$.

3. An adaptive importance sampling algorithm

In this section we present our adaptive algorithm that leverages importance sampling and reduced order models to largely reduce the computational cost of Algorithm 1.

3.1. Importance sampling

Importance Sampling (IS) is a classical variance-reduction technique to accelerate the estimation of expectations through Monte Carlo methods, see, e.g., Asmussen and Glynn [29, Chapter 5]. The driving idea is to consider a biasing distribution $\widetilde{\rho}$, satisfying $\mathrm{supp}(X\rho)\subset\mathrm{supp}(\widetilde{\rho})$, which should increase the sampling frequency of events that contribute the most to the estimation $\mathbb{E}[X]$. Then, the integral identity

$$\mathbb{E}[X]=\int_\Gamma X(\boldsymbol{\xi})\rho(\boldsymbol{\xi})d\boldsymbol{\xi}=\int_\Gamma X(\boldsymbol{\xi})w(\boldsymbol{\xi})\widetilde{\rho}(\boldsymbol{\xi})d\boldsymbol{\xi},$$

where $w(\boldsymbol{\xi}):=\frac{\rho(\boldsymbol{\xi})}{\widetilde{\rho}(\boldsymbol{\xi})}$, suggests to approximate $\mathbb{E}[X]$ using the *unbiased* Monte Carlo estimator $\frac{1}{M}\sum_{j=1}^M X(\boldsymbol{\xi}_j)w(\boldsymbol{\xi}_j)$, where $\{\boldsymbol{\xi}_j\}_{j=1}^M$ are i.i.d. samples from the $\widetilde{\rho}$ density, and $\{w(\boldsymbol{\xi}_j)\}_{j=1}^M$ are weights compensating the change in the sampling distribution. By choosing suitably $\widetilde{\rho}$, one can reduce the variance of the Monte Carlo estimator and obtain more accurate estimates at a reduced number of samples.

In our context, we would like to leverage IS to reduce the variance of the empirical gradient $\nabla\widetilde{F}^\varepsilon_{S_k}$ at every iteration of Algorithm 1. Since we have remarked that for every couple (z_k,t_k)

$$\mathrm{supp}\left(\frac{g_\varepsilon'(f(z_k,\boldsymbol{\xi})-t_k)\nabla f(z_k,\boldsymbol{\xi})\rho(\boldsymbol{\xi})}{1-\beta}\right)\subset G^\varepsilon_{z_k,t_k}[f],$$

a reasonable choice for the biasing density would be

$$\widetilde{\rho}(\boldsymbol{\xi})=\frac{\mathbb{1}_{G^\varepsilon_{z_k,t_k}[f]}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})}{\mathbb{P}(G^\varepsilon_{z_k,t_k}[f])},$$

which precisely attributes a non-zero, re-weighted, probability only to samples in $G^\varepsilon_{z_k,t_k}[f]$. For a given set of i.i.d. samples $S=\{\boldsymbol{\xi}_j\}_{j=1}^M$ drawn from $\widetilde{\rho}$, the IS estimator for $\nabla\widetilde{F}^\varepsilon(z_k,t_k)$ would then be

$$\nabla\widetilde{F}^{\varepsilon,\mathrm{IS}}_{S_k}(z_k,t_k)=\frac{1}{M}\sum_{j=1}^M\frac{g_\varepsilon'(f(z_k,\boldsymbol{\xi}_j)-t_k)\nabla f(z_k,\boldsymbol{\xi}_j)w(\boldsymbol{\xi}_j)}{1-\beta},$$

where $w(\boldsymbol{\xi}_j)=\mathbb{P}(G^\varepsilon_{z_k,t_k}[f])$ for any $j=1,\dots.M$. However, this promising approach suffers from few drawbacks. First of all, the generation of samples from $\widetilde{\rho}$ may be cumbersome. A standard acceptance-rejection algorithm generates a sample $\boldsymbol{\xi}$ from, e.g., the ρ density, evaluates $f(z_k,\boldsymbol{\xi})$ and accepts $\boldsymbol{\xi}$ if and only if $\boldsymbol{\xi}\in G^\varepsilon_{z_k,t_k}[f]$. This approach can be very computationally expensive whenever

the evaluation of f is costly, and it even worsens as β tends to one as this leads to higher rejection rates. Second, we will be interested in sampling the set S_k when only z_k is available, while t_k still has to be computed (as it is the case in Algorithm 1). Of course, since t_k is a sampled-based approximation of $\mathrm{VaR}_\beta[f(z_k,\cdot)]$, we could estimate it through several evaluations of $f(z,\cdot)$, which however may be again computationally expensive, and whose cost can actually be avoided as will show.

To overcome these computational challenges, we leverage Reduced Order Models (ROMs) to both estimate t_{k+1} and then check whether ξ belongs to a suitable risk-region with a reduced computational cost. This in turn induces a ROM-based biasing distribution. We do update the ROM to keep track of the evolution of the control variable z along the optimization, so that a sequence of biasing densities $\{\widetilde{\rho}_k\}_{k\in\mathbb{N}}$ is constructed. In the next paragraph, we recall the framework introduced in [20] to estimate CVaR_β using ROMs, and extend it to the optimization context.

3.2. Construction of the ROM-based biasing density

We assume that for every $z\in C$, we have available a Reduced Order Model (ROM) of the map $f(z,\cdot):\Gamma\to\mathbb{R}$ denoted by $f_r(z,\cdot)$, which is much faster to evaluate than $f(z,\cdot)$, and of a computable a-posteriori error bound $e_z:\Gamma\to\mathbb{R}^+$ such that

$$|f(z,\xi)-f_r(z,\xi)|\le e_z(\xi),\quad \rho\text{-a.e. } \xi\in\Gamma. \tag{14}$$

Suppose now that we are at the iteration k, we have computed z_{k+1} and for the moment, assume further that we can compute $\mathrm{VaR}_\beta[f(z_{k+1},\cdot)]$ and $\mathrm{VaR}_\beta[f_r(z_{k+1},\cdot)]$ exactly, and denote them by $\widehat{t}_{k+1}$ and $\widehat{t}^r_{k+1}$, respectively.

A first naive approach to construct a ROM-based biasing distribution from which to sample the next set S_{k+1} would be to consider the risk-region

$$G^\epsilon_{z_{k+1},\widehat{t}^r_{k+1}}[f_r]:=\left\{\xi:\ f_r(z_{k+1},\xi)\ge\widehat{t}^r_{k+1}-\frac{\epsilon}{2}\right\},$$

and consequently to define

$$\widetilde{\rho}(\xi)=\frac{\mathbb{1}_{G^\epsilon_{z_{k+1},\widehat{t}^r_{k+1}}[f_r]}(\xi)\rho(\xi)}{\mathbb{P}(G^\epsilon_{z_{k+1},\widehat{t}^r_{k+1}}[f_r])}. \tag{15}$$

However, $G^\epsilon_{z_{k+1},\widehat{t}_{k+1}}[f]\subset G^\epsilon_{z_{k+1},\widehat{t}^r_{k+1}}[f_r]$ does not necessarily hold, and thus (15) is not an admissible biasing density. Driven by [20], we instead define $\widehat{t}^{r,e}_{k+1}:=\mathrm{VaR}_\beta\left[f_r(z_{k+1},\cdot)-e_{z_{k+1}}\right]$,

$$G^{\epsilon,e}_{z_{k+1},\widehat{t}^{r,e}_{k+1}}[f_r]:=\left\{\xi:\ g'_\epsilon(f_r(z_{k+1},\xi)+e_{z_{k+1}}(\xi)-\widehat{t}^{r,e}_{k+1})>0\right\}=\left\{\xi:\ f_r(z_{k+1},\xi)+e_{z_{k+1}}(\xi)\ge\widehat{t}^{r,e}_{k+1}-\frac{\epsilon}{2}\right\},$$

and set

$$\widetilde{\rho}(\xi):=\frac{\mathbb{1}_{G^{\epsilon,e}_{z_{k+1},\widehat{t}^{r,e}_{k+1}}[f_r]}(\xi)\rho(\xi)}{\mathbb{P}(G^{\epsilon,e}_{z_{k+1},\widehat{t}^{r,e}_{k+1}}[f_r])}. \tag{16}$$

The next proposition is adapted from [20, Lemma 3.2].

Proposition 1. *The following inclusion holds*

$$G^\epsilon_{z_{k+1},\widehat{t}_{k+1}}[f]\subset G^{\epsilon,e}_{x_{k+1},\widehat{t}^{r,e}_{k+1}}[f_r].$$

Proof. From (14) it follows that $f(z_{k+1})\ge f_r(z_{k+1},\cdot)-e_{z_{k+1}}$ which implies $\widehat{t}_{k+1}=\mathrm{VaR}_\beta[f(z_{k+1},\cdot)]\ge\mathrm{VaR}_\beta\left[f_r(z_{k+1},\cdot)-e_{z_{k+1}}\right]=\widehat{t}^{r,e}_{k+1}$ due to the monotonicity of VaR_β. Then, taking a $\xi\in G^\epsilon_{z_{k+1},\widehat{t}_{k+1}}[f]$, the sequence of inequalities

$$f_r(z_{k+1},\xi)+e_{z_{k+1}}(\xi)\ge f(z_{k+1},\xi)\ge\widehat{t}_{k+1}-\frac{\epsilon}{2}\ge\widehat{t}^{r,e}_{k+1}-\frac{\epsilon}{2},$$

shows that $\xi\in G^{\epsilon,e}_{z_{k+1},\widehat{t}^{r,e}_{k+1}}[f_r]$. □

Proposition 1 guarantees that (16) is a suitable biasing density. However, both $\widehat{t}_{k+1}$ and $\widehat{t}^{r,e}_{k+1}$ are not computable quantities. In a numerical implementation, at iteration k we underestimate the future t_{k+1} (which is computed by minimizing $\widetilde{F}^\epsilon(z_{k+1},t)$ with respect to t at iteration $k+1$) via

$$t^{r,e}_{k+1}=\mathrm{argmin}_{t\in\mathbb{R}}\ t+\frac{1}{(1-\beta)M_{\mathrm{Trial}}}\sum_{j=1}^{M_{\mathrm{Trial}}}g^\epsilon\left(f_r(z_{k+1},\xi^{\mathrm{Trial}}_j)-e_{z_{k+1}}(\xi^{\mathrm{Trial}}_j)-t\right),$$

where $\left\{\xi^{\mathrm{Trial}}_j\right\}_{j=1}^{M_{\mathrm{Trial}}}$ are distributed according to ρ. The new i.i.d. samples S_{k+1} are then drawn from the biasing density

$$\widetilde{\rho}^{k+1}(\xi):=\frac{\mathbb{1}_{G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}[f_r]}(\xi)\rho(\xi)}{\mathbb{P}(G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}[f_r])}.$$

In principle, it might occur that the estimated $t^{r,e}_{k+1}$ is larger than the actual t_{k+1} (computed at iteration $k+1$). In our numerical simulations we never encountered such case which, however, can be easily avoided by recomputing $t^{r,e}_{k+1}$ with a larger value of M_{Trial} (as asymptotically $t^{r,e}_{k+1}$ converges to $\widehat{t}^{r,e}_{k+1}$, hence it will be eventually smaller than t_{k+1} [30]), followed by another acceptance-rejection algorithm to generate a new set S_{k+1}.

3.3. The algorithm

Our adaptive importance sampling algorithm is described in Algorithm 2. At every iteration, an alternating procedure (Steps 5–6) identical to Algorithm 1 is carried out, although now the IS estimator

$$\widetilde{F}^{\epsilon,\text{IS}}_{S_k}(z,t) := t + \frac{1}{(1-\beta)M_k}\sum_{\xi^k_j\in S_k} g_\epsilon(f(z,\xi^k_j)-t)w_k,$$

is employed, with weights $w_k = \mathbb{P}\left(G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}[f_r]\right)$. Condition (7) now reads

$$\widetilde{\mathbb{E}}_k\left[\|\nabla\widetilde{F}^{\epsilon,\text{IS}}_{S_k}(z_k,t_k)-\nabla\widetilde{F}^{\epsilon}(z_k,t_k)\|^2\right] = \frac{\text{Var}_{\widetilde{\rho}_k}\left[\frac{g'_\epsilon(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)w_k}{1-\beta}\right]}{M_k} \le \theta^2\|\nabla\widetilde{F}^{\epsilon}(z_k,t_k)\|^2,$$

where $\widetilde{\mathbb{E}}_k$ is the expectation with respect to the set of M_k i.i.d. random variables distributed according to $\widetilde{\rho}_k$. The practical norm test is thus,

$$\varrho = \frac{\sum_{\xi^k_j\in S_k}\left\|\frac{g'_\epsilon(f(z_k,\xi^k_j)-t_k)\nabla f(z_k,\xi^k_j)w_k}{1-\beta} - \nabla\widetilde{F}^{\epsilon,IS}_{S_k}(z_k,t_k)\right\|^2}{(M_k-1)(M_k)\theta^2\|R_{S_k}(z_k)\|^2}. \tag{17}$$

Notice that a direct calculation shows

$$\text{Var}_{\widetilde{\rho}_k}\left[\frac{g'_\epsilon(f(z_k,\cdot)-t_k)\nabla f(z_k,\cdot)w_k}{1-\beta}\right] \approx \frac{w_k}{1-\beta}\mathbb{E}_\rho\left[\|\nabla f(z_k,\cdot)\|^2|G_{z_k,t_k}\right] - \left\|\mathbb{E}_\rho\left[\nabla f(z_k,\cdot)|G_{z_k,t_k}\right]\right\|^2, \tag{18}$$

where the approximation is due to the fact that g'_ϵ is a smoothed version of $\mathbb{1}_{G^\epsilon_{z_k,t_k}}$ and that t_k is only a sample-based estimate of $\text{VaR}_\beta[f(z_k,\cdot)]$. It follows that, if along the optimization the ROM model becomes increasingly accurate, then w_k tends to $(1-\beta)$, and we obtain claim (13). We refer to Section 4 for a quantitative comparison between (12) and (18).

Once the next sample size has been computed, we construct a reduced order model for the new iterate z_{k+1}. We then draw a set of M_{Trial} samples, evaluate the a-posteriori error estimator for each of these samples, compute $t^{r,e}_{k+1}$ and estimate the size of the region $G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}$ (lines 11–14). Finally, a standard acceptance-rejection algorithm is used to generate samples $\{\boldsymbol{\xi}_j\}_{j=1}^{M_{k+1}}$ distributed according to $\widetilde{\rho}^{k+1}$. Notice that both the computation of $t^{r,e}_{k+1}$ and the test if $\boldsymbol{\xi}_{\text{test}}$ belongs to $G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}$ rely exclusively on the ROM f_r, so that the full-order model is never used between lines 11 and 23.

We conclude this section remarking that the efficient construction of a ROM along the optimization is highly problem dependent and may be nontrivial. In Section 4, we discuss in detail an approach in the context of PDE-constrained optimization under uncertainty. Here we limit to make the interesting, and maybe surprising, observation: the ROM does not need to be particularly over accurate in the risk-region. While surely an overall accurate ROM leads to sharper estimates of $t^{r,e}_{k+1}$, the ROM is used here essentially to exclude realization of $\boldsymbol{\xi}$ that do not contribute to the CVaR_β. Thus, large values of the a-posteriori error estimator outside the "true" risk region $G^\epsilon_{z_k,t_k}$ are those problematic, since they may actually lead to the acceptance of unnecessary samples in the acceptance-rejection step.

3.4. Convergence analysis

In this subsection we analyze the convergence of Algorithm 2. Readers more interested in implementation details and numerical results can proceed directly to Section 4.

We start by remarking that to analyze the convergence of Algorithm 2 it is sufficient to study Algorithm 1. Indeed, the additional complexity of Algorithm 2 due to the ROM-based construction of the biasing distribution only affects the number of samples needed to satisfy a condition on the gradients (see (22)). While [17] proposes Algorithm 1 without a convergence analysis, a first study of an alternating procedure for the CVaR_β minimization has been proposed in [19], under however the hypothesis that $\widetilde{F}$ is jointly strongly convex for all $(z,t)\in\mathcal{Z}\times\mathbb{R}$. This assumption is somehow too strong since if $t\notin\text{supp}(f(z,\cdot))$, then $\widetilde{F}$ is only linear in t in a neighborhood of (z,t). In practice though, Algorithm 1 never explores these problematic regions since the minimization step in t guarantees that $t\in\text{supp}(f(z,\cdot))$.

Our analysis proceeds through a different path, consisting of reinterpreting the alternating procedure of Algorithm 1 as a single gradient step on a reduced functional defined only over the z variable. Under suitable assumptions detailed next, we argue that this reduced functional is strongly convex and has Lipschitz continuous gradients, so that the convergence of an adaptive sampling gradient descent algorithm directly follows from available results in literature.

To begin our analysis, we make the following assumption on the properties of $\widetilde{F}^\epsilon$ with respect to the variable t.

Algorithm 2 Adaptive importance sampling.

Require $\mathbf{z}_0$, M_0, $M_{\max}$, α, M_{trial}
Set $k = 0$.
Sample $S_k = \{\boldsymbol{\xi}_i\}_{i=1}^{M_k}$ from ρ.
while $\sum_{j=0}^{k} M_j \leq M_{\max}$ **do**
  Compute $t_k \in \operatorname{argmin}_{t\in\mathbb{R}} F^{\epsilon,\text{IS}}_{S_k}(z_k, t)$.
  $z_{k+1} = \mathcal{P}\left(z_k - \alpha \nabla F^{\epsilon,\text{IS}}_{S_k}(z_k, t_k)\right)$.
  Compute ϱ in (17).
  if $\varrho > 1$ **then** $M_{k+1} = \varrho M_k$
  end if.
  Construct ROM $f_r(z_{k+1}, \cdot)$
  Draw M_{Mtrial} samples from ρ: $S_{\text{Trial}} = \{\boldsymbol{\xi}_j\}_{j=1}^{M_{\text{trial}}}$.
  Compute a-posteriori error estimator: $|f(z_{k+1}, \boldsymbol{\xi}_j) - f_r(z_{k+1}, \boldsymbol{\xi}_j)| \leq e_{z_{k+1}}(\boldsymbol{\xi}_j)$, $j = 1, \dots, M_{\text{Trial}}$.
  Compute $t^{r,e}_{k+1} = \operatorname{argmin}_{t\in\mathbb{R}} t + \frac{1}{(1-\beta)M_{\text{Trial}}} \sum_{j=1}^{M_{\text{Trial}}} g^{\epsilon}\left(f_r(z_{k+1}, \boldsymbol{\xi}_j) - e_{z_{k+1}}(\boldsymbol{\xi}_j) - t\right)$.
  Estimate $\mathbb{P}\left(G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}\left[f_r\right]\right) = \#\left\{\boldsymbol{\xi} \in S_{\text{Trial}} : \boldsymbol{\xi} \in G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}\right\} / M_{\text{Trial}}$.
  Importance sampling step: $n = 0$.
  while $n \leq M_{k+1}$ **do**
    Draw a sample $\boldsymbol{\xi}_{test}$ from density ρ.
    if $\boldsymbol{\xi}_{test} \in G^{\epsilon,e}_{z_{k+1},t^{r,e}_{k+1}}\left[f_r\right]$ **then**
      Add sample $\boldsymbol{\xi}_{test}$ to S_{k+1}.
      $n \leftarrow n + 1$.
    end if
  end while
end while

Assumption 1. *For every $z \in C$, the minimization problem*

$$\min_{t\in\mathbb{R}} \widetilde{F}^{\epsilon}(z, t)$$

admits an unique minimum $t^\star(z)$ and defines a differentiable map $h : C \to \mathbb{R}$ such that $h(z) = t^\star(z)$.

Notice that, due to the differentiability of $\widetilde{F}^{\epsilon}$ with respect to t, $\partial_t \widetilde{F}^{\epsilon}(z, h(z)) = 0$ trivially holds and, assuming the uniqueness of $t^\star(z)$, the local existence and smoothness of h follow from the implicit function theorem. Assumption 1 allows us to introduce the reduced functional

$$\widehat{F}^{\epsilon}(z) := h(z) + \frac{1}{1-\beta}\mathbb{E}\left[g^{\epsilon}(f(z,\cdot) - h(z))\right],$$

and a straight calculation shows that

$$\nabla \widehat{F}^{\epsilon}(z) = \nabla \widetilde{F}^{\epsilon}(z, h(z)) + \partial_t \widetilde{F}^{\epsilon}(z, h(z))\nabla h(z) = \nabla \widetilde{F}^{\epsilon}(z, h(z)), \tag{19}$$

where the term $\partial_t \widetilde{F}^{\epsilon}$ cancels due to the definition of h. From (19), it follows that the alternating procedure

$$t_k = \min_{t\in\mathbb{R}} \widetilde{F}^{\epsilon}(z_k, t), \quad z_{k+1} = \mathcal{P}\left(z_k - \alpha \nabla \widetilde{F}^{\epsilon}(z_k, t_k)\right),$$

is equivalent to the iteration

$$\tilde{z}_{k+1} = \mathcal{P}\left(\tilde{z}_k - \alpha \nabla \widehat{F}^{\epsilon}(\tilde{z}_k)\right),$$

in the sense that $\tilde{z}_{k+1} = z_{k+1}$ and $t_k = h(z_k)$ for every k. Our goal is then to study the convergence of an adaptive sampling gradient algorithm applied to the reduced functional $\widehat{F}^{\epsilon}$. To do so, we make the following assumptions on $f(z, \cdot)$.

Assumption 2. *For ρ-a.e. $\boldsymbol{\xi}$, the mapping $z \to f(z, \boldsymbol{\xi})$ is twice differentiable, and there exist two positive constants $\mu_f, L_f \in \mathbb{R}^+$, such that $\forall\ \delta z \in \mathcal{Z}$*

$$\mu_f \|\delta z\|^2 \leq (\delta z, H_f(z, \boldsymbol{\xi})\delta z) \leq L_f \|\delta z\|^2 \quad \rho\text{-a.e. } \boldsymbol{\xi},$$

$H_f(z, \boldsymbol{\xi})$ being the Hessian of f at $(z, \boldsymbol{\xi})$. Furthermore, there exists a positive constant $C_f \in \mathbb{R}$ such that $\forall z \in C$, $\|\nabla f(z, \cdot)\| \leq C_f$ ρ-a.e.

Assumption 2 implies that f is strongly convex, and has Lipschitz continuous gradients ρ-a.s. This in turn allows us to show that the same properties hold true for $\widehat{F}^{\epsilon}$.

Lemma 2. *The functional $\widehat{F}^{\epsilon}$ is strongly convex over C and has Lipschitz continuous gradients.*

Proof. We first prove the Lipschitz continuity of gradients. Using that $\sup_{t\in\mathbb{R}} |g_\epsilon'(t)| \le 1$, Eq. (19), Assumption 2, and the Lipschitz continuity of f, h and g_ϵ',

$$\begin{aligned}
\left\|\nabla \widehat{F}^\epsilon(x) - \nabla \widehat{F}^\epsilon(y)\right\| &\le \frac{1}{1-\beta}\mathbb{E}\left[\left\|g_\epsilon'(f(x,\cdot)-h(x))\nabla f(x,\cdot) - g_\epsilon'(f(y,\cdot)-h(y))\nabla f(y,\cdot)\right\|\right] \\
&\le \frac{1}{1-\beta}\mathbb{E}\Big[g_\epsilon'(f(x,\cdot)-h(x))\|\nabla f(x,\cdot)-\nabla f(y,\cdot)\| \\
&\quad + (g_\epsilon'(f(x,\cdot)-h(x)) - g_\epsilon'(f(y,\cdot)-h(y)))\|\nabla f(y,\cdot)\|\Big] \\
&\le \frac{1}{1-\beta}\left(\|x-y\|L_f + C_f\mathbb{E}\left[g_\epsilon'(f(x,\cdot)-h(x)) - g_\epsilon'(f(y,\cdot)-h(y))\right]\right) \\
&\le L_{\widehat{F}^\epsilon}\|x-y\|,
\end{aligned}$$

for a suitable positive constant $L_{\widehat{F}^\epsilon}$. Next, we focus on the strong convexity. Denoting with $\mathcal{H}_{\widehat{F}^\epsilon}$ the Hessian of $\widehat{F}^\epsilon$, the proof consists in showing that

$$\min_{\delta z\in\mathcal{Z}} \frac{(\delta z, \mathcal{H}_{\widehat{F}^\epsilon}(z)\delta z)}{\|\delta z\|^2} =: \mu_{\widehat{F}^\epsilon} > 0, \quad \forall z \in \mathcal{Z}.$$

To do so, we will actually leverage the Hessian of $\widetilde{F}^\epsilon$, that is,

$$\mathcal{H}_{\widetilde{F}^\epsilon}(z,t) := \begin{pmatrix} \widetilde{F}^\epsilon_{zz}(z,t) & \widetilde{F}^\epsilon_{tz}(z,t) \\ \widetilde{F}^\epsilon_{zt}(z,t) & \widetilde{F}^\epsilon_{tt}(z,t)\end{pmatrix}$$

where, using the shorthand notation $\bar{g}_\epsilon = g_\epsilon(f(z,\cdot)-h(z))$, direct calculations show that

$$\begin{aligned}
(\delta z, \widetilde{F}^\epsilon_{zz}(z,t)\delta z) &= \frac{1}{1-\beta}\mathbb{E}\left[\bar{g}_\epsilon''(\delta z, \nabla f(z,\cdot))(\nabla f(z,\cdot),\delta z) + \bar{g}_\epsilon'(\delta z, H_f(z,\cdot)\delta z)\right], \\
(\delta t, \widetilde{F}^\epsilon_{zt}(z,t)\delta z) &= (\delta z, \widetilde{F}^\epsilon_{tz}(z,t)\delta t) = -\frac{\delta t}{1-\beta}\mathbb{E}\left[\bar{g}_\epsilon''(\nabla f(z,\cdot),\delta z)\right], \\
(\delta t, \widetilde{F}^\epsilon_{tt}(z,t)\delta t) &= \frac{\delta t^2}{1-\beta}\mathbb{E}\left[\bar{g}_\epsilon''\right],
\end{aligned}$$

As a matter of fact, it is well-known $\mathcal{H}_{\widehat{F}^\epsilon}(z)$ corresponds to the Schur complement of $\mathcal{H}_{\widetilde{F}^\epsilon}$ evaluated in $(z, h(z))$, namely

$$\mathcal{H}_{\widehat{F}^\epsilon}(z) = \widetilde{F}^\epsilon_{zz}(z,h(z)) - \widetilde{F}^\epsilon_{tz}(z,h(z))\widetilde{F}^\epsilon_{tt}(z,h(z))^{-1}\widetilde{F}^\epsilon_{zt}(z,h(z)),$$

so that

$$\begin{aligned}
(\delta z, \mathcal{H}_{\widehat{F}^\epsilon}(z)\delta z) &= (\delta z, \widetilde{F}^\epsilon_{zz}(z,h(z))\delta z) - (\delta z, \widetilde{F}^\epsilon_{tz}(z,h(z))\widetilde{F}^{-1}_{tt}(z,h(z))\widetilde{F}^\epsilon_{zt}(z,h(z))\delta z) \\
&= \frac{1}{1-\beta}\mathbb{E}\left[\bar{g}_\epsilon'(\delta z, H_f(z,\cdot)\delta z) + \bar{g}_\epsilon''|(\nabla f(z,\cdot),\delta z)|^2\right] \\
&\quad - \frac{1}{1-\beta}\frac{\left(\mathbb{E}\left[\bar{g}_\epsilon''(\nabla f(z,\cdot),\delta z)\right]\right)^2}{\mathbb{E}\left[\bar{g}_\epsilon''\right]}.
\end{aligned} \tag{20}$$

Using the positivity of $\bar{g}''$ and the Cauchy-Schwarz inequality,

$$\frac{\left(\mathbb{E}\left[\bar{g}_\epsilon''(\nabla f(z,\cdot),\delta z)\right]\right)^2}{\mathbb{E}\left[\bar{g}_\epsilon''\right]} \le \mathbb{E}\left[\bar{g}_\epsilon''|(\nabla f(z,\cdot),\delta z)|^2\right]$$

which inserted into (20) leads to

$$(\delta z, \mathcal{H}_{\widehat{F}^\epsilon}(z)\delta z) \ge \frac{1}{1-\beta}\mathbb{E}\left[\bar{g}_\epsilon'(\delta z, H_f(z,\cdot)\delta z)\right] \ge \frac{\mu_f}{1-\beta}\mathbb{E}\left[\bar{g}_\epsilon'(f(z,\cdot)-h(z))\right]\|\delta z\|^2 \ge \frac{\mu_f}{2}\|\delta z\|^2,$$

where the last step uses that $\mathbb{E}\left[\bar{g}_\epsilon'(f(z,\cdot)-h(z))\right] \ge \frac{1}{2}\mathbb{E}\left[\mathbb{1}_{f(z,\cdot)-h(z)}\right] = \frac{1-\beta}{2}$. □

We are now almost ready to state a convergence result for Algorithm 1. We however need an additional assumption on the sample function F_{S_k}.

Assumption 3. *For every $z \in C$, the minimization problem*

$$\min_{t\in\mathbb{R}} F^\epsilon_{S_k}(z,t)$$

admits a unique minimum $t^\star_{S_k}(z)$ and defines a map $h_{S_k} : C \to \mathbb{R}$ such that $h_{S_k}(z) = t^\star_{S_k}(z)$.

Assumption 3 allows us to introduce the reduced version of F_{S_k},

$$\widehat{F}^\epsilon_{S_k}(z) := h_{S_k}(z) + \frac{1}{1-\beta}\mathbb{E}\left[g_\epsilon(f(z,\cdot)-h_{S_k}(z))\right].$$

Repeating verbatim the argument made for $\widehat{F}^\epsilon$, the alternating procedure of Algorithm 1 is equivalent to the gradient descent iteration

$$z_{k+1} = \mathcal{P}\Big(z_k - \alpha \nabla \widehat{F}^\epsilon_{S_k}(z_k)\Big). \tag{21}$$

The next theorem shows that choosing at each iteration a sample S_k such that

$$\mathbb{E}_k\left[\left\|\nabla \widehat{F}^\epsilon_{S_k}(z_k) - \nabla \widehat{F}^\epsilon(z_k)\right\|^2\right] \leq \theta^2 \|R(z_k)\|^2 \tag{22}$$

is sufficient to preserve linear convergence of the iterates $\{z_k\}_{k\geq 1}$.

Theorem 3. *Let $\{z_k\}_{k\geq 1}$ be the sequence generated by (21), with $\alpha < L_{\widehat{F}^\epsilon}$ and such that each S_k satisfies (22). Then, for sufficiently small $\theta > 0$, z_k converges linearly in expectation, i.e.,*

$$\mathbb{E}\left[\|z_{k+1} - z^\star\|\right] \leq \varrho^k \|z_0 - z^\star\|,$$

for some $\varrho \in [0, 1)$.

Proof. The claim follows directly from [17, Theorem 2.10] since due to Lemma 2, $\widehat{F}^\epsilon$ is strongly convex and has Lipschitz continuous gradients. □

Remark 1. *Similarly to (7), Condition (22) is not numerically implementable since $\widehat{F}$ is not computable. However, in addition $\nabla \widehat{F}^\epsilon_{S_k}(z_k)$ is a biased estimator of $\nabla \widehat{F}^\epsilon(z_k)$, since h_{S_k} is only an asymptotic unbiased estimator of h, see, e.g., Hong* et al. *[30]. Thus,*

$$\mathbb{E}_k\left[\left\|\nabla \widehat{F}^\epsilon_{S_k}(z_k) - \nabla \widehat{F}^\epsilon(z_k)\right\|^2\right] = \mathbb{E}_k\left[\left\|\nabla \widehat{F}^\epsilon_{S_k}(z_k) - \mathbb{E}_k\left[\nabla \widehat{F}^\epsilon_{S_k}(z_k)\right]\right\|^2\right] + \mathbb{E}\left[\left\|\mathbb{E}\left[\nabla \widehat{F}^\epsilon_{S_k}(z_k)\right] - \nabla \widehat{F}^\epsilon(z_k)\right\|^2\right],$$

where the first term corresponds to the variance of $\nabla \widehat{F}_{S_k}(z_k, t_k)$, while the second contribution is a bias term that could be further bounded using the Lipschitz continuity of g'_ϵ and Assumption 2,

$$\mathbb{E}_k\left[\left\|\mathbb{E}\left[\nabla \widehat{F}^\epsilon_{S_k}(z_k)\right] - \nabla \widehat{F}^\epsilon(z_k)\right\|^2\right] \leq \frac{C_f^2}{(1-\beta)^2}\mathbb{E}_k\left[|h(z_k) - h_{S_k}(z_k)|^2\right].$$

In our numerical experiments, we neglect the bias term and choose the next sample size by exclusively control the variance term using the practical norm test (17).

Remark 2 (Extension to general descent directions)**.** *The analysis readily extends to the more general iteration $z_{k+1} = \mathcal{P}\Big(z_k - \alpha \frac{1}{M_k}\sum_{j=1}^{M_k} p_k(\xi_j)\Big)$, $\{p_k(\xi_j)\}_{j=1}^{M_k}$ denoting a set of stochastic descent directions. To guarantee convergence is sufficient to assume that for every $k \geq 1$,*

$$\mathbb{E}_k\left[\left\|\sum_{j=1}^{M_k}\frac{1}{M_k}p_k(\xi_j) - \nabla \widehat{F}^\epsilon(z_k)\right\|^2\right] \leq \theta \|\nabla \widehat{F}^\epsilon(z_k)\|^2, \tag{23}$$

for a sufficiently small θ. The proof then follows again the same steps of [17, Theorem 2.10]. Note that by adding and subtracting $\nabla \widehat{F}^\epsilon_{S_k}(z_k)$, (23) reduces to the left hand side of (22), plus an additional bias term.

4. Numerical experiments

In this section we present numerical experiments arising from the field of PDE-constrained optimization under uncertainty. We consider the domain $D = (0, 1)^2$ whose boundary $\partial\Omega$ is divided into two disjoint sets, $\Gamma_D := (0, 1) \times \{0, 1\}$ and $\Gamma_N := \partial\Omega \setminus \Gamma_D$, on which homogeneous Dirichlet and Neumann boundary conditions are imposed. We denote by V the Sobolev space $H^1_{0,\Gamma_D}(D)$. The PDE-constrained optimization problem reads

$$\min_{z\in\mathcal{Z}} \mathrm{CVaR}_\beta\left[\frac{1}{2}\|y(z,\xi)\|^2_{L^2(D_0)} + \frac{\nu}{2}\|z\|^2_{\mathcal{Z}}\right], \tag{24}$$

where $D_0 := (0.5, 1) \times (0, 1) \subsetneq D$ and $y(z, \xi)$ is the solution of the random partial differential equation,

$$\int_D \kappa(\mathbf{x}, \xi)\nabla y(\mathbf{x}; z, \xi) \cdot \nabla v(\mathbf{x})\, d\mathbf{x} = \int_D l(\mathbf{x})v(\mathbf{x})\, d\mathbf{x} + \langle Bz, v\rangle, \quad \forall v \in V. \tag{25}$$

Here, $l(\mathbf{x})$ models three point source terms,

$$l(\mathbf{x}) := 10^2 \sum_{i=1}^{3} \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|}{L}\right),$$

where $L = 0.05$, $\mathbf{x}_j = [0.25, j0.25]$ for $j = 1, \ldots, 3$, and $B : \mathcal{Z} \to V'$ is a control operator representing a Neumann boundary control acting on the right vertical edge (denoted from now E),

$$\langle Bz, v\rangle = \int_E z(\mathbf{x})v(\mathbf{x})\, d\mathbf{x},$$

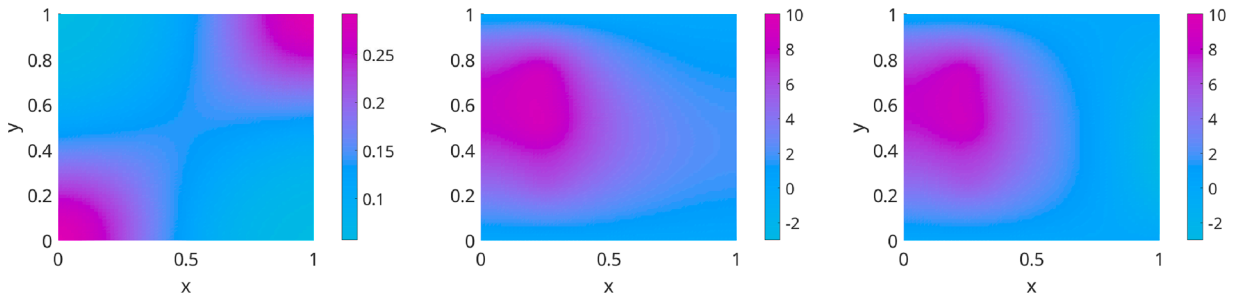


Fig. 1. Realization of κ_2 for $\boldsymbol{\xi} = (0.01, 0.1)$ (left panel) and the corresponding uncontrolled state variable $y(0, \boldsymbol{\xi})$ (center panel) and optimal state $y(z^\star, \boldsymbol{\xi})$ (right panel).

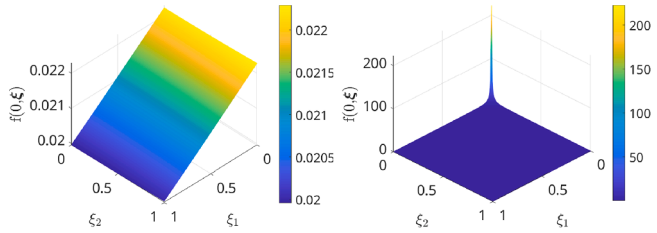


Fig. 2. Graphical representation of the map $\boldsymbol{\xi} \to f(0, \boldsymbol{\xi})$ for κ_1 and κ_2.

hence $\mathcal{Z} = L^2(E)$.

We will consider two different diffusion coefficients, namely

$$\kappa_1(\mathbf{x}, \boldsymbol{\xi}) := 1 + \frac{\xi_1}{2} \sin(\pi x) \sin(\pi y) + \frac{\xi_2}{5} \sin(\pi x) \sin(2\pi y),$$

representing a truncated Fourier expansion, and

$$\kappa_2(\mathbf{x}, \boldsymbol{\xi}) := 0.01 + \xi_1 \exp\left(\sin(\pi x) \sin(\pi y)\right) + \xi_2 \exp\left(\cos(\pi x) \cos(\pi y)\right).$$

The parameters $\boldsymbol{\xi} = (\xi_1, \xi_2)$ are uniformly distributed in $(0, 1)^2$. Problem (24) is a simple model of a heating process where the heat generated at specific locations has to be extracted on the right portion of the domain by applying a suitable flux along the right edge. It can be embedded into the notation of the previous sections by defining the map $f : \mathcal{Z} \times (0, 1)^2 \to \mathbb{R}$, with $f(z, \boldsymbol{\xi}) = \frac{1}{2}\|y(z, \boldsymbol{\xi})\|^2_{L^2(\Omega_0)} + \frac{\nu}{2}\|z\|^2_{L^2(E)}$. An instance of κ_2 is depicted in the left panel of Fig. 1, together with the corresponding uncontrolled state $y(0, \boldsymbol{\xi})$ on the center panel, and the optimal control state $y(z^\star, \boldsymbol{\xi})$ on the right panel, where we can see that heat is extracted from the right boundary, leading to a decreased temperature.

We choose two random variables ($M = 2$) so that an accurate reference solution of (24) can be computed using a tensorized Gauss-Legendre quadrature formula [31]. We emphasize that the two diffusion coefficients lead to very different properties of the objective function. Fig. 2 shows the values of the map $\boldsymbol{\xi} \to f(0, \boldsymbol{\xi})$, that is of the uncontrolled ($z = 0$) physical process. For κ_1, $f(0, \cdot)$ has a favorable behavior with overall minor variations across the parameters domain, whereas for κ_2, $f(0, \cdot)$ assumes very large values within a very tiny region around the vertex $(0, 0)$. This is an extremely challenging case for adaptive sampling algorithms since the uniform random sampling rarely generates samples very close to $(0, 0)$, which however are the dominant contributions to the CVaR_β and to the variance of the sampled gradients.

4.1. Dynamic update of the ROM

In this subsection we detail how the ROMs $f_r(z_{k+1}, \cdot)$ are computed on the fly during Algorithm 2 (line 10) at a negligible computational cost.

Let $\left\{V_r^{k+1}\right\}_{k\geq 0}$ be a sequence of finite dimensional subspaces, not necessarily nested, of V. At a given iterate (z_{k+1}, t_k), we consider the ROM

$$f_r(z_{k+1}, \cdot) := \frac{1}{2}\|y_r(z_{k+1}, \cdot)\|^2_{L^2(D_0)} + \frac{\nu}{2}\|z_{k+1}\|^2_{\mathcal{Z}},$$

where $y_r(z_{k+1}, \boldsymbol{\xi}) \in V_r^{k+1}$ is the solution of the reduced basis equation [32,33],

$$\int_D \kappa(\mathbf{x}, \boldsymbol{\xi}) \nabla y_r(\mathbf{x}; z_{k+1}, \boldsymbol{\xi}) \cdot \nabla v(\mathbf{x})\, d\mathbf{x} = \int_D l(\mathbf{x}) v(\mathbf{x})\, d\mathbf{x} + \langle B z_{k+1}, v\rangle, \quad \forall v \in V_r^{k+1}.$$

Clearly, this approach is computationally sound only if the subspace V_r^{k+1} is of small dimension and it can be obtained cheaply. To achieve this, we leverage that the calculation of t_k requires the evaluation of $f(z_k, \cdot)$ at all realizations $\boldsymbol{\xi} \in S_k$. This in turn implies

that a solution $y(z_k, \xi)$ to (25) is computed for each $\xi \in S_k$. We therefore consider a subset $\mathcal{I}_1 \subset S_k$ of the snapshots $\{y(z_k, \xi)\}_{\xi \in S_k}$, and extract the most significant modes using Proper Orthogonal Decomposition, see, e.g., Quarteroni et al. [32, Section 6]. However, as discussed in Section 3.2, the ROM needs to be accurate also in the complement of the risk-region. Hence, we enhance the POD basis using a Greedy approach that relies on a set of samples $\mathcal{I}_2$ that lie outside the current risk-region $G_{z_{k+1}, t^{r,e}_{k+1}}[f_r]$. The set $\mathcal{I}_2$ can be readily obtained by saving a few of the rejected samples during the acceptance-rejection construction of the biasing density. The Greedy procedure consists in iteratively adding the snapshots that maximize a reduced basis a-posterior error estimator $e^{\mathrm{RB}}_{z_{k+1}}$, which measures how well the current subspace V_r^{k+1} approximates the solution manifold of (25). In our implementation, we used a classical residual-based error estimator, see, e.g., Quarteroni et al. [32, Chapter 3]. The overall procedure to construct a ROM at each iteration is described by Algorithm 3. In our numerical implementation, we provide to Algorithm 3 two sets $\mathcal{I}_1$ and $\mathcal{I}_2$ consisting of, respectively, $\min\{M_k, 40\}$ randomly chosen elements of S_k, and of maximum 100 samples that lie outside $G_{z_{k+1}, t^{r,e}_{k+1}}[f_r]$. The tolerance is decreased along the outer iterations of Algorithm 2 as we wish to have a more accurate ROM as we approach convergence.

Algorithm 3 Greedy-enhanced construction of $f_r(z_{k+1}, \cdot)$.

Require $\{y(z_k, \xi)\}_{\xi \in \mathcal{I}_1}$, $\mathcal{I}_2$, Tol $\in \mathbb{R}^+$, $e^{\mathrm{RB}}_{z_{k+1}}$.
$V_r^{k+1} = \mathrm{POD}(\{y(z_k, \xi)\}_{\xi \in \mathcal{I}_1}, \mathrm{Tol})$.
while $\max_{\xi \in \mathcal{I}_2} e^{\mathrm{RB}}_{z_{k+1}}(\xi) \geq$ Tol **do**
    Choose $\xi^\star = \mathrm{argmin}_{\xi \in \mathcal{I}_2} e^{\mathrm{RB}}_r z_{k+1}(\xi)$.
    $V_r^{k+1} = V_r^{k+1} \cup y(z_{k+1}, \xi^\star)$.
    Set $\mathcal{I}_2 = \mathcal{I}_2 \setminus \xi^\star$.
end while
Return V_r^{k+1}.

Notice that, despite V_r^{k+1} is constructed using snapshots corresponding to both the iterates z_k and z_{k+1}, a standard residual-based a-posteriori error still provides a rigorous upper bound for the error $\|y(z_{k+1}, \cdot) - y_r(z_{k+1}, \cdot)\|^2_{L^2(D)}$. However, the latter is generally non zero even at realizations whose snapshots are included in V_r^{k+1}. The availability of $e^{\mathrm{RB}}_{z_{k+1}}$ permits to estimate the a-posterior error estimator $e_{z_{k+1}}$ for the ROM model since

$$
\begin{aligned}
|f(z_{k+1}, \xi) - f_r(z_{k+1}, \xi)| &= \frac{1}{2}\left| \|y(z_{k+1}, \xi)\|^2_{L^2(D_0)} - \|y_r(z_{k+1}, \xi)\|^2_{L^2(D_0)} \right| \\
&\leq \frac{1}{2} \|y(z_{k+1}, \xi) + y_r(z_{k+1}, \xi)\|^2_{L^2(D_0)} \|y(z_{k+1}, \xi) - y_r(z_{k+1}, \xi)\|^2_{L^2(D_0)} \\
&\leq \frac{1}{2} \|y(z_{k+1}, \xi) + y_r(z_{k+1}, \xi)\|^2_{L^2(D_0)} e^{\mathrm{RB}}_{z_{k+1}}(\xi).
\end{aligned}
$$

In practice, as $y(z_{k+1}, \xi)$ is not available, we use the estimator $e_{z_{k+1}}(\xi) := \|y_r(z_{k+1}, \xi)\|^2_{L^2(D_0)} e^{\mathrm{RB}}_{z_{k+1}}(\xi)$.

4.2. Comparison of Algorithms 1 and 2

We now discuss numerical experiments aimed at comparing the standard adaptive sampling algorithm (Algorithm 1) and its new variant leveraging importance sampling (Algorithm 2). All experiments are based on a $\mathbb{P}^1$ finite element discretization of the state equation with 1023 degrees of freedom. The smoothing parameter for the CVaR_β is $\varepsilon = 10^{-4}$. The optimization parameters are $\alpha = 0.5$, $\theta = 0.5$, $M_0 = 10$, $M_{trial} = 10^3$, and all algorithms are stopped when a computational budget of $M_{\max} = 10^6$ PDE solutions would be exceeded in the next iteration. For Algorithm 2, we do include the number of additional PDE solutions needed by the Greedy-enhanced construction of the ROM into the total count. The results shown are obtained starting from the initial guess $z_0 = 0$ and by averaging the randomness of the sample selection over 20 runs.

The top plots of Fig. 3 show the growth of the sample sizes M_k along the optimization for different values of β. The bottom plots show the empirical variance of the gradients that appears in the two norm tests (10) and (17). As argued in Section 2.3, increasing β raises the computational cost of adaptive sampling algorithms for two main reasons. First, sampling from the original distribution produces progressively fewer samples in the risk region, which are essential to accurately estimate the gradient of the risk-averse functional. Indeed, the latter is an approximation of the conditional expectation of the quantity of interest restricted to the risk-region (see (11)). Secondly, the empirical variance of the stochastic gradient $\xi \to \frac{g'_\varepsilon(f(z_k, \cdot) - t_k)\nabla f(z_k, \xi)}{1-\beta}$ increases (bottom row) as it is dominated by the second moment of $\nabla f(z_k, \xi)$. The variance growth is further amplified if the $f(z, \cdot)$ has already a strong variability in the risk-region as illustrated in the bottom-right panel for κ_2. As a result, the sample sizes M_k (top row) grow more rapidly, and hence the computational budget is exhausted the earlier the larger is the value of β. By using importance sampling, we indeed achieve a variance reduction and hence Algorithm 2 manages to perform additional iterations.

Next, we consider the convergence behavior of both algorithms. Fig. 4 shows the decay in expectation of the relative error with respect to the number of iterations and number of PDEs solved for different values of β. Since some lines are overlapping, we indicate with a circle marker the last iteration of Algorithm 1. The results are in agreement with the convergence analysis of Section 3.4 and with Theorem 3. The asymptotic convergence rate is the same for both Algorithms 1 and 2, but Algorithm 2 can perform additional iterations, reaching significant smaller relative errors in all cases.

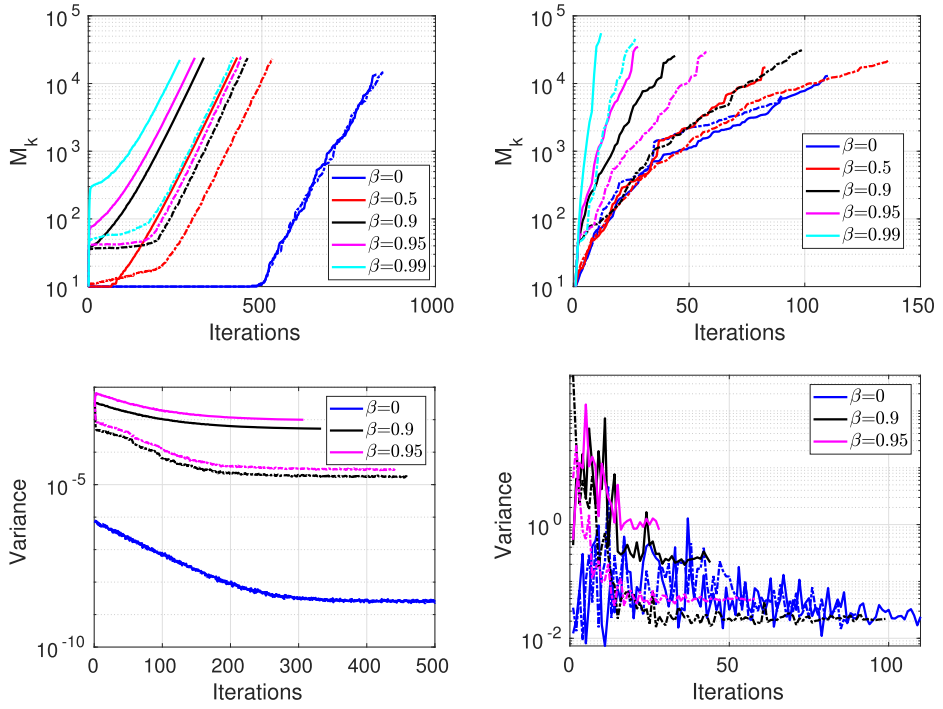


Fig. 3. Top row: growth of the sample size M_k with iterations. Bottom row: estimated empirical variance of the gradients along the iterations. Left panels refer to κ_1, the right panels to κ_2. The continuous lines refer to Algorithm 1 while the dashed-dotted ones to Algorithm 2.

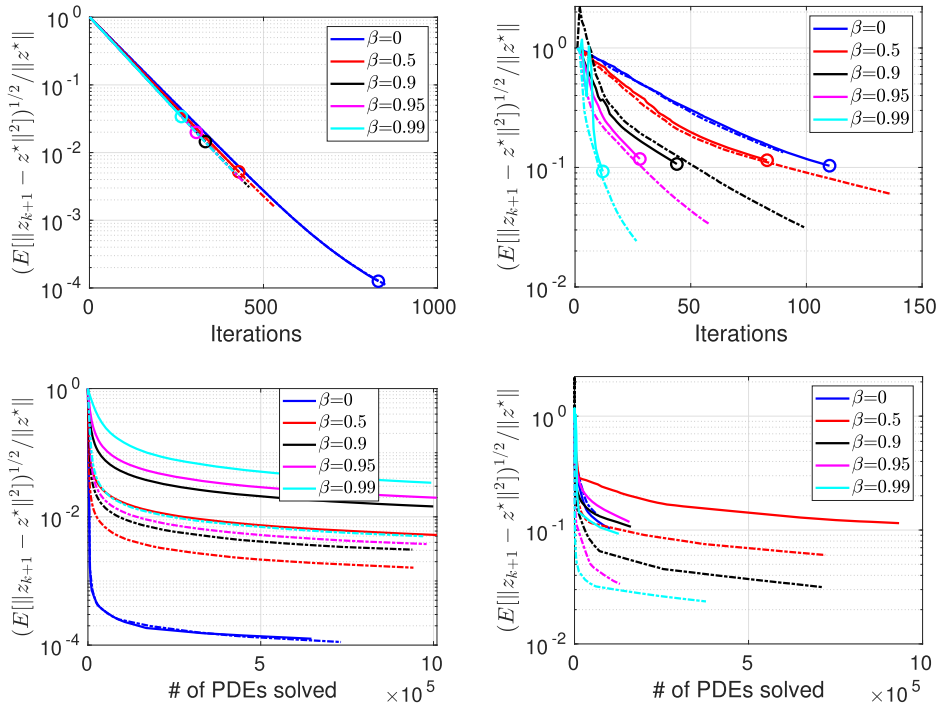


Fig. 4. Decay of the relative error with respect to the number of iterations (top) and number of PDEs solved (bottom). The left column refers to κ_1, while the right one to κ_2. The continuous line corresponds to Algorithm 1, the dashed-dotted lines to Algorithm 2. The circle marker highlights the last iteration of Algorithm 1.

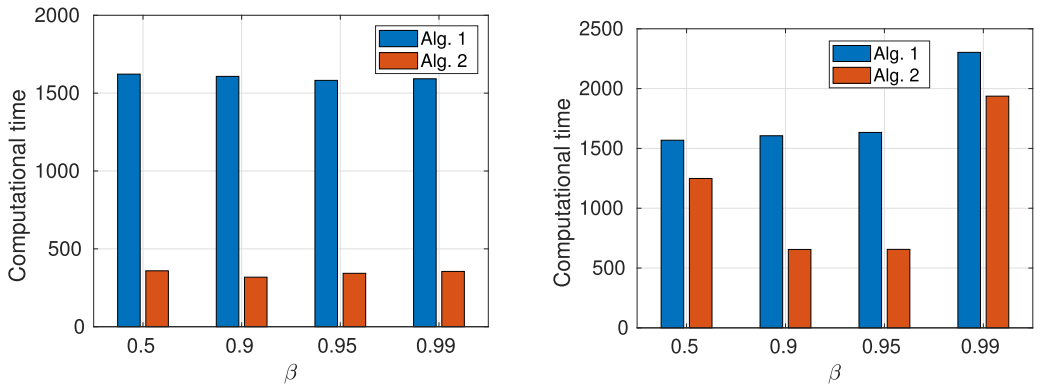


Fig. 5. Comparison of the computational times in seconds required by Algorithms 1 and 2 to reach approximately the same relative error. Left panel refers to κ_1, while the right one to κ_2.

Table 1
Ratio between computational times.

β	0.5	0.9	0.95	0.99	β	0.5	0.9	0.95	0.99
T_{IS}/T	0.22	0.19	0.21	0.22	T_{IS}/T	0.79	0.4	0.4	0.84

Finally, we compare the computational times. The results are obtained using a serial implementation, in which both the PDEs solves and the acceptance-rejection sampling are performed sequentially. To fairly compare the algorithms, as they exhaust the computational budget at different iterations, Fig. 5 reports the computational times required by both algorithms up to the iteration at which Algorithm 1 is stopped (circle marker in Fig. 4). This is a reasonable metric, since Fig. 4 shows that both algorithms reach approximately the same relative error. Table 1 reports the related speedup factors. In all cases, Algorithm 2 leads to an effective reduction of the computational times, sustaining the proposed methodology.

5. Conclusions

This work has presented an improved adaptive sampling algorithm that adjusts both the sample size and the biasing distribution at each iteration of the optimization process. By restricting the sampling to a neighborhood of the risk region, the algorithm enables the use of smaller sample size per iteration, while preserving the linear asymptotic convergence rate. As a consequence, the new methodology presents a better computational complexity measured in computational times, to achieve a desired relative error.

Future research could focus on further improving the on-the-fly construction of the reduced-order model (ROM). For instance, this could involve recycling the reduced basis subspace between iterations and augmenting it only when necessary. Another interesting direction would be the generation of the biasing distribution without relying on a ROM, by, for instance leveraging the information gathered from evaluating the full-order model at each iteration in combination with machine learning algorithms.

CRediT authorship contribution statement

Sandra Pieraccini: Writing – review & editing, Supervision, Methodology; **Tommaso Vanzan:** Writing – review & editing, Writing – original draft, Software, Methodology, Formal analysis, Conceptualization.

Data availability

No data was used for the research described in the article.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors are members of the INdAM-GNCS group. The first author acknowledges financial support by the European Union through PRIN project *Full and Reduced order modelling of coupled systems: focus on non-matching methods and automatic learning (FaReX)* (CUP:E53D23005510006). The second author has been partially supported by the INdAM-GNCS project GNCS 2024 - CUP E53C23001670001.

References

[1] L. Bottou, F.E. Curtis, J. Nocedal, Optimization methods for large-scale machine learning, SIAM Rev. 60 (2) (2018) 223–311.
[2] I. Goodfellow, Y. Bengio, A. Courville, D. Learning, Deep Learning MIT Press, 2016.
[3] P. Brandimarte, An Introduction to Financial Markets: A Quantitative Approach, John Wiley & Sons, 2017.
[4] F.J. Fabozzi, P.N. Kolm, D.A. Pachamanova, S.M. Focardi, Robust Portfolio Optimization and Management, John Wiley & Sons, 2007.
[5] J.R. Birge, F. Louveaux, Introduction to Stochastic Programming, Springer Science & Business Media, 2011.
[6] A. Shapiro, D. Dentcheva, A. Ruszczynski, Lectures on Stochastic Programming: Modeling and Theory, SIAM, 2021.
[7] H. Antil, D.P. Kouri, M.-D. Lacasse, D. Ridzal, Frontiers in PDE-Constrained Optimization, 163, Springer, 2018.
[8] J.M. Martínez Frutos, F.P. Esparza, Optimal Control of PDEs Under Uncertainty: An Introduction with Application to Optimal Shape Design of Structures, Springer, 2018.
[9] F. Nobile, T. Vanzan, A combination technique for optimal control problems constrained by random PDEs, SIAM/ASA J. Uncertainty Quantif. 12 (2) (2024) 693–721.
[10] R.T. Rockafellar, S. Uryasev, et al., Optimization of conditional value-at-risk, J. Risk 2 (2000) 21–42.
[11] R.T. Rockafellar, S. Uryasev, The fundamental risk quadrangle in risk management, optimization and statistical estimation, Surv. Oper. Res. Manage. Sci. 18 (1–2) (2013) 33–53.
[12] D.P. Kouri, T.M. Surowiec, Existence and optimality conditions for risk-averse PDE-constrained optimization, SIAM/ASA J. Uncertainty Quantif. 6 (2) (2018) 787–815.
[13] S. Curi, K.Y. Levy, S. Jegelka, A. Krause, Adaptive sampling for stochastic risk-averse learning, Adv. Neural Inf. Process. Syst. 33 (2020) 1036–1047.
[14] R.H. Byrd, G.M. Chin, J. Nocedal, Y. Wu, Sample size selection in optimization methods for machine learning, Math. Program. 134 (1) (2012) 127–155.
[15] M.P. Friedlander, M. Schmidt, Hybrid deterministic-stochastic methods for data fitting, SIAM J. Sci. Comput. 34 (3) (2012) A1380–A1405.
[16] R. Bollapragada, R. Byrd, J, Adaptive sampling strategies for stochastic optimization, SIAM J. Optim. 28 (4) (2018) 3312–3343.
[17] F. Beiser, B. Keith, S. Urbainczyk, B. Wohlmuth, Adaptive sampling strategies for risk-averse stochastic optimization with constraints„ IMA J. Numer. Anal. 43 (6) (2023) 3729–3765.
[18] Y. Xie, R. Bollapragada, R. Byrd, J. Nocedal, Constrained and composite optimization via adaptive sampling methods, IMA J. Numer. Anal. 44 (2) (2024) 680–709.
[19] S. Ganesh, F. Nobile, Gradient-based optimisation of the conditional-value-at-risk using the multi-level Monte Carlo method, J. Comput. Phys. 495 (2023) 112523.
[20] M. Heinkenschloss, B. Kramer, T. Takhtaganov, K. Willcox, Conditional-value-at-risk estimation via reduced-order models, SIAM/ASA J. Uncertainty Quantif. 6 (4) (2018) 1395–1423.
[21] M. Heinkenschloss, B. Kramer, T. Takhtaganov, Adaptive reduced-order model construction for conditional value-at-risk estimation, SIAM/ASA J. Uncertainty Quantif. 8 (2) (2020) 668–692.
[22] Z. Zou, D.P. Kouri, W. Aquino, A locally adapted reduced-basis method for solving risk-averse PDE-constrained optimization problems, SIAM/ASA J. Uncertainty Quantif. 10 (4) (2022) 1629–1651.
[23] M. Markowski, Efficient Solution of Smoothed Risk-Adverse Pde-Constrained Optimization Problems, Technical Report, Rice University, 2022. Ph.D. thesis.
[24] D.P. Kouri, T.M. Surowiec, Risk-averse PDE-constrained optimization using the conditional value-at-risk, SIAM J. Optim. 26 (1) (2016) 365–396.
[25] G. Ciaramella, F. Nobile, T. Vanzan, A multigrid solver for PDE-constrained optimization with uncertain inputs, J. Sci. Comput. 101 (1) (2024) 13.
[26] R.T. Rockafellar, S. Uryasev, Conditional value-at-risk for general loss distributions, J. Banking Finance 26 (7) (2002) 1443–1471.
[27] F.S. Hashemi, S. Ghosh, R. Pasupathy, On adaptive sampling rules for stochastic recursions, in: Proceedings of the Winter Simulation Conference, IEEE, 2014, pp. 3959–3970.
[28] C. Cartis, K. Scheinberg, Global convergence rate analysis of unconstrained optimization methods based on probabilistic models, Math. Program. 169 (2018) 337–375.
[29] S. Asmussen, P.W. Glynn, Stochastic Simulation: Algorithms and Analysis, 57, Springer, 2007.
[30] L.J. Hong, Z. Hu, G. Liu, Monte Carlo methods for value-at-risk and conditional value-at-risk: a review, ACM Trans. Model. Comput. Simul. (TOMACS) 24 (4) (2014) 1–37.
[31] I. Babuška, F. Nobile, R. Tempone, A stochastic collocation method for elliptic partial differential equations with random input data, SIAM J. Numer. Anal. 45 (3) (2007) 1005–1034.
[32] A. Quarteroni, A. Manzoni, F. Negri, Reduced Basis Methods for Partial Differential Equations: An Introduction, 92, Springer, 2015.
[33] J.S. Hesthaven, G. Rozza, B. Stamm, et al., Certified Reduced Basis Methods for Parametrized Partial Differential Equations, 590, Springer, 2016.