Les Cahiers du GERAD ISSN: 0711–2440

Risk-adverse optimization by Conditional Value-at-Risk and stochastic approximation

C. Audet, J. Bigeon, R. Couderc, M. Kokkolaras

G–2022–04

February 2022

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2022
– Bibliothèque et Archives Canada, 2022

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2022
– Library and Archives Canada, 2022

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Risk-adverse optimization by Conditional Value-at-Risk and stochastic approximation

Charles Audet [a, b]

Jean Bigeon [a, c]

Romain Couderc [a, b, d]

Michael Kokkolaras [a, e]

[a] *GERAD, Montréal (Qc), Canada, H3T 1J4*

[b] *Département de mathématiques et génie industriel, École Polytechnique de Montréal, Montréal (Qc), Canada, H3C 3A7*

[c] *Nantes Université, Centrale Nantes, CNRS, LS2N, 44000 Nantes, France*

[d] *Uniersité Grenoble Alpes, CNRS, Grenoble INP (Institute of Engineering University), G-SCOP, 38000 Grenoble, France*

[e] *Department of Mechanical Engineering, McGill University, Montréal (Qc), Canada, H3A 0C3*

charles.audet@gerad.ca
jean.bigeon@ls2n.fr
romain.couderc@grenoble-inp.fr
michael.kokkolaras@mcgill.ca

February 2022
Les Cahiers du GERAD
G–2022–04

Abstract : Engineering design is often faced with uncertainties, making it difficult to determine an optimal design. In an unconstrained context, this amounts to choose the desired trade-off between risk and performance. In this paper, an optimization problem with an adaptive risk level is stated using the Conditional Value-at-Risk (CVaR_α). Under mild conditions on the objective function and taking advantage of the noise, CVaR_α allows to smooth the problem. Then, a specific algorithm based on a stochastic approximation scheme is developed to solve the problem. This algorithm has two appealing properties. First, it does not use any estimation of quantile to compute the minimum of the CVaR_α of the noised objective function. Second, it uses only two function evaluations per iteration regardless of the problem dimension. A proof of convergence to a minimum of CVaR_α of the objective function is established. This proof is based on martingale theory and does not require any information about the differentiability or continuity of the function. Finally, test problems from the literature are combined in a benchmark set to compare our algorithm to a risk-neutral and a worst-case optimization algorithms. These tests prove the ability of the algorithm to be efficient in both cases, especially in large dimension.

Keywords: Risk-adverse optimization, Conditional Value-at-Risk, stochastic approximation, deriva tive-free optimization, non convex optimization

Acknowledgements: The authors are grateful to Matt Menickelly from Argonne National Laboratory for valuable suggestions and constructive discussions about the ROBOBOA algorithm.

1 Introduction

In optimization under uncertainty, the quality of an optimum is characterized by a compromise between the objective function value and its robustness. In the unconstrained case, the robustness of an optimum is its propensity to be less sensitive to uncertainties than another. In practice, an optimum of high performance with high variability is usually undesired. On the contrary, a robust optimum associated with low performance may also be unsuitable. Finding a compromise is the goal of robust optimization. Several approaches have been proposed depending on the amount of information available. The Robust Optimisation (RO) approach [4, 5, 10, 19] needs only an uncertainty set in which the uncertain variables vary. The Distributionally Robust Optimization (DRO) approach [8, 11, 29, 35] needs that the uncertainties belong to a family of distributions sharing some properties. Finally, the Risk-Adverse (RA) approach [31, 34] needs that the uncertainties be modeled by random variables whose the distribution is known or at least can be estimated with adequate accuracy.

In engineering contexts where information on uncertainties may be missing or completely absent, the use of RO or DRO seems particularly interesting. Nevertheless, these methods raise two problems: first, they are used in a worst-case setting, therefore the optimum found is robust with respect to all possible perturbations. Only the robustness is taken into account and this can lead to overly conservative solutions. Secondly, these methods typically require specific properties such as differentiability [5, 19] or convexity [4]. Many engineering problems resulting from experiments or simulations do not possess such properties. On the contrary, the RA approach requires more information on the problem, i.e, more data on the uncertainties, but this information can be used to transform the problem in a more suitable way for optimization. In addition, the RA approach allows the use of metrics that provide finer control over the performance/robustness trade-off than the worst-case solution. This trade-off is particularly interesting in an unconstrained context where the uncertainties can not lead to infeasibility. An example of these different measures applied on a function whose the decision variables are noised by truncated Gaussian variables are plotted on Figure 1. It is worth to notice that an algorithm based on robust optimization may lead to a loss of performance if the desired robustness degree was less conservative.

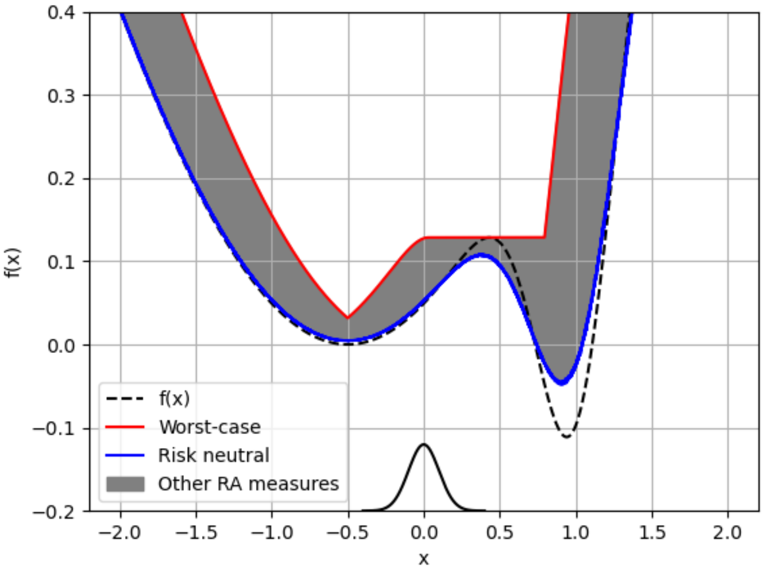


Figure 1: A function (dotted line) with two extreme measures for dealing with uncertainties on x: the expectation measure and the robust measure. The grey area represents all the other possible risk-adverse measures

Several measures of risk aimed at quantifying this trade-off have been developed, notably in finance, and are now used in several fields. Among these measures, Value-at-Risk (VaR_α) and Conditional Value at Risk (CVaR_α) (also called Expected Shortfall (ES) in [12], Average Value-at-Risk (AVaR_α) in [31] or superquantile in [26]) are particularly interesting. VaR_α corresponds to the $1-\alpha$ quantile tail of a distribution while the CVaR_α is the conditional mean of the tail portion of a distribution. The α parameter allows to easily manage the trade-off between performance and robustness. CVaR_α measure is also a coherent risk measure and benefits from properties such as convexity and monotony [34]. It has also the advantage of being able to be modeled as an infemum of an expectation function and thus no quantile estimation is needed to compute it. This measure has already been used in multidisciplinary design optimization (MDO) [18], in structural engineering design [26] and in a blackbox optimization context [37] (for details about blackbox optimization readers are reffered to [2]). In the two first approaches, the authors still assume differentiability of the objective function. In the latter, despite the authors' goal of reducing the computational cost, their algorithm requires hundred million function evaluations to solve problems with ten variables. In both cases, results are prohibitive in an engineering design context with time consuming function evaluations.

Computational costs of engineering risk-adverse optimization algorithm depend mainly on the number of evaluations used to estimate the measure of risk and the algorithm chosen to solve the stochastic problem. There exist methods to reduce the number of function evaluations whose the risk is costly to estimate but they may be limited: the approach used in [37] for instance still has a great cost in terms of function evaluations. The algorithm choice is also an efficient way to reduce the number of function evaluations. In stochastic programming there are two main classes of algorithms: Sample Average Approximation (SAA) [25, 30] and Stochastic Approximation (SA) [24]. The SAA is a two-parts method: the first step approximates the expected objective function by Monte Carlo sampling, then deterministic optimization methods may be used to solve the approximate problem. The principle of SA is to find the root of the gradient of the expected cost function. This method has the advantage of requiring less structural properties than the SAA-based methods . In an engineering context, where the objective function lacks structural properties, the SA-based methods seem to be the most appropriate. In particular, [14, 28] introduces the Smoothed Functional (SF) algorithm, which approximates the gradient of the expected objective function by its convolution with a multivariate distribution. This algorithm has the advantage of smoothing the function, which is extremely useful in a context of engineering optimization where derivatives may be nonexistent or hard to obtain. Moreover, like the Simultaneous Perturbation Stochastic Approximation [32] algorithm, the SF algorithm estimates gradients by simultaneously perturbing all components of the decision variables. This estimation requires only two function evaluations regardless of the problem dimension.

The goals of this paper are threefold. First, the proposed approach must give the ability for the user to choose its desired degree of robustness. Second, it must be computationally tractable in an engineering context, especially for dimensions of the order of tens variables. Third, it must be able to deal with non smooth and non convex problem as it may be the case in engineering. These contributions are organized as follows. In Section 2, we present the risk-adverse optimization problem, develop our CVaR_α approach and derive the following theoretical results. First, the problem of CVaR_α optimization may be expressed as a problem of expectation minimization. Therefore, none quantile estimation is needed to compute the minimum of CVaR_α, yielding savings on the computational burden of function evaluations. Second, the expectation of an objective function whose decision variables are perturbed by Gaussian noised may be seen as a convolution product. The key result is that under mild assumptions on the objective function, this convolution product is in fact infinitely differentiable. This result may be extended to any distribution by the Rosenblatt probabilistic transformation [27]. In Section 3, taking advantage of the previous results and inspired by Smoothed Functional algorithm [6], we propose a new algorithm requiring fewer function evaluations to optimize the objective function. A convergence proof of this algorithm to a robust optimum is provided in Section 4. In Section 5, this algorithm is compared with a derivative-free stochastic optimization algorithm [3] and a derivative-free robust optimization algorithm [19] to show its competitiveness from expectation to worst-case optimization. Finally, Section 6 draws conclusions and discusses future work.

2 The theoretical properties of Conditional Value-at-Risk approach for risk-adverse optimization

This section describes different formulations of CVaR_α including structural properties to show its interest in a context of risk-adverse optimization, where the derivatives of the objective functions may not exist.

2.1 The formulation of risk-adverse optimization problem with Conditional Value-at-Risk

This work studies risk-adverse optimization problem of the following form

$$\min_{\mathbf{x}\in\mathbb{R}^n} \; \Xi\left[f(\mathbf{x},\boldsymbol{\xi})\right] \tag{1}$$

where $f : \mathbb{R}^n \times \mathcal{S}_1 \to \mathcal{S}_2$ is a function and $\Xi : \mathcal{S}_2 \to \mathbb{R}$ is a measure allowing to handle the uncertainties $\boldsymbol{\xi} \in \mathcal{S}_1$. This formulation is general and allows to formulate a large number of problems according to the definitions of $\mathcal{S}_1$, $\mathcal{S}_2$ and $\Xi[\cdot]$. For instance:

- if $\mathcal{S}_1 = \Omega$, $\mathcal{S}_2 = \mathcal{L}^1(\Omega,\mathcal{F},\mathbb{P})$ and $\Xi[\cdot] = \mathbb{E}_{\boldsymbol{\xi}}[\cdot]$ where $(\Omega,\mathcal{F},\mathbb{P})$ defines a probability space, then Problem (1) is stochastic optimization problem, i.e. the risk-neutral optimization of the objective function under stochastic uncertainties.
- if $\mathcal{S}_1 = \mathcal{U} \subset \mathbb{R}^m$, $\mathcal{S}_2 = \mathcal{C}(\mathcal{U})$ (space of continuous function on $\mathcal{U}$) and $\Xi[\cdot] = \max_{\mathbf{u}\in\mathcal{U}}[\cdot]$, then Problem (1) is a robust optimization problem, i.e. the optimization of the worst case objective function under deterministic uncertainties.

Before defining $\mathcal{S}_1$, $\mathcal{S}_2$ and $\Xi[\cdot]$ for this work, several assumptions are stated concerning the nature of uncertainties and the nature of f in Problem (1). These assumptions will be used throughout.

Assumption 1. On the uncertainties, the following hold.

a. The uncertainty vector $\boldsymbol{\xi} = (\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) \in \Omega = \Omega_1 \times \Omega_2$ is a random vector where $\boldsymbol{\xi}_\mathbf{x}$ models the uncertainties on the decision variables and $\boldsymbol{\xi}_\mathbf{p}$ models the uncertainties on the parameters.

b. The random vectors $\boldsymbol{\xi}_\mathbf{x}$ and $\boldsymbol{\xi}_\mathbf{p}$ are independent.

c. The univariate marginal cumulative distribution functions of $\boldsymbol{\xi}_\mathbf{x}$ are known. That means, without loss of generality, that $\xi_\mathbf{x}$ may be assumed as a symmetric truncated Gaussian random vector and mapped to its original distribution by the use of the isoprobabilistic Rosenblatt transformation [27].

Assumption 2. Let $(\Omega,\mathcal{F},\mathbb{P})$ be a probability space and $\mathcal{X} \subset \mathbb{R}^n$ be a compact set, on the function f, the following hold.

a. The function $f(x,\cdot)$ is $\mathcal{F}$-measurable, for all $x \in \mathbb{R}^n$.

b. The function $f(x,\cdot)$ is bounded, for all $x \in \mathcal{X}$.

c. A map is already embedded in the function f to perform the Rosenblatt transformation.

Assumptions 1.a and 1.b are common assumptions in risk-adverse optimization with a probabilistic approach. Assumption 1.c is a strong assumption in an engineering context where the information on the uncertainties may be limited. Nevertheless, in cases where only bounds of uncertainties on the design variables are known, the use of an uniform distribution will allow to mimic the behavior of an interval approach. Assumption 2 is equivalent to say that for all $\mathbf{x} \in \mathbb{R}^n$, the function $f(x,\cdot)$ belongs to $\mathcal{L}^1(\Omega,\mathcal{F},\mathbb{P})$. This assumption ensures that the function is sufficiently regular so that its expectation exists and is well defined. It is worth to notice that no assumption is made on the continuity nor differentiability of the objective function with respect to $\mathbf{x}$. Moreover, Assumption 2.b ensures the existence of a minimum. Finally, Assumption 2.c is an assumption allowing to simplify the notation by avoiding the write of the Rosenblatt transfomation all along this work.

Consider now Problem (1) with the following elements $\mathcal{S}_1 = \mathbb{R}^n \times \Omega_2$, $\mathcal{S}_2 = \mathcal{L}^1(\Omega, \mathcal{F}, \mathbb{P})$ and $\Xi[\cdot] = \mathrm{CVaR}_\alpha[\cdot]$. The goal of Problem (1) becomes then to minimize the CVaR_α of the function $f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p})$ at a risk level of interest $\alpha \in (0,1]$ with respect to $\mathbf{x} \in \mathcal{X}$. That is, solve the risk-adverse optimization problem:

$$\min_{\mathbf{x}\in\mathcal{X}} \ \mathrm{CVaR}_\alpha(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p})), \tag{2}$$

where $\mathrm{CVaR}_\alpha(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}))$ is defined [31] by

$$\mathrm{CVaR}_\alpha(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p})) = \inf_{t\in\mathbb{R}} \left\{ t + \tfrac{1}{\alpha}\mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) - t)_+] \right\} \tag{3}$$

in which $(x)_+ = \max(x, 0)$ denotes the positive part of x. When the cumulative distribution function of $f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_p)$ is continuous for all $\mathbf{x} \in \mathcal{X}$, CVaR_α may be written as

$$\mathrm{CVaR}_\alpha(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p})) = t^*_\mathbf{x}(\alpha) + \tfrac{1}{\alpha}\mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) - t^*_\mathbf{x}(\alpha))_+] \tag{4}$$

where $t^*_\mathbf{x}(\alpha)$ is defined as the left-side quantile:

$$t^*_\mathbf{x}(\alpha) = \inf\{t \ : \ \mathbb{P}(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) \le t) \ge 1-\alpha\}. \tag{5}$$

This formulation has the advantage to give the user the ability to choose the desired degree of robustness. Choosing a level $\alpha = 1$ amounts to minimize the expectation function. In the opposite case, choosing a level near 0 for a distribution with bounded support, amounts to be as conservative as the worst-case approach. This is particularly useful to obtain a solution less sensitive to the uncertainties. However, this flexibility comes at a cost: smaller is the value of α, more function evaluations are needed to estimate CVaR_α, especially to estimate the left-side quantile $t^*_\mathbf{x}(\alpha)$. This problem may be fixed by using a different formulation from [31].

Proposition 1. *Let* $\alpha := \frac{1}{1+\beta_2} \in (0,1]$, *under Assumption 2 Problem (2) is equivalent to*

$$\min_{(\mathbf{x},t)\in\mathcal{X}\times\mathbb{R}} \rho_\alpha(\mathbf{x}, t) \tag{6}$$

where

$$\rho_\alpha(\mathbf{x}, t) := \mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) + (t - f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}))_+ + \beta_2(f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) - t)_+] \tag{7}$$

Proof. By Assumption 2, $Z := f(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) \in \mathcal{L}_1(\Omega, \mathcal{F}, \mathbb{P})$ and by setting $\alpha = \frac{1}{1+\beta_2}$, we have:

$$\begin{aligned} \mathrm{CVaR}_\alpha(Z) &= \inf_{t\in\mathbb{R}} \{t + \mathbb{E}[(1+\beta_2)(Z-t)_+]\} \\ &= \mathbb{E}[Z] - \mathbb{E}[Z] + \inf_{t\in\mathbb{R}} \{\ \mathbb{E}[t + (1+\beta_2)(Z-t)_+]\} \\ &= \mathbb{E}[Z] + \inf_{t\in\mathbb{R}} \mathbb{E}[(t-Z) + (Z-t)_+ + \beta_2(Z-t)_+] \\ &= \mathbb{E}[Z] + \inf_{t\in\mathbb{R}} \mathbb{E}[(t-Z)_+ + \beta_2(Z-t)_+] \\ &= \inf_{t\in\mathbb{R}} \mathbb{E}[Z + (t-Z)_+ + \beta_2(Z-t)_+]. \end{aligned}$$

As the function is continuous and convex in t, the infemum is a minimum, which completes the proof. □

Thus, by adding one extra variable, the problem can be formulated as an expectation minimization problem which does not require any estimation of the right-side quantile. That allows to reduce the number of function evaluations. However, Problem (6) remains difficult to solve for two main reasons: first, the function f lacks structural properties such as convexity and differentiability. Most gradient-based algorithms will fail to solve it. Second, the problem dimension may be large and derivative-free algorithms may not be efficient in this context.

2.2 The Conditional Value-at-Risk structural properties

The following results are inspired by [6, 28]. In these previous work, the function is perturbed with Gaussian noise in order to estimate its gradient by a convolution product between its gradient and a Gaussian function. In the present work, given that $\mathbf{x}$ is assumed to be already randomly perturbed by a truncated Gaussian distribution, the objective function of Problem (6) $\rho_\alpha(\mathbf{x}, t)$ may be directly seen as a convolution product and is differentiable with respect to $\mathbf{x}$. Moreover, the derivatives of the $\rho_\alpha(\mathbf{x}, t)$ may be calculated analytically. These results are formally stated in the following propositions.

Proposition 2. *Under Assumption 1 and Assumption 2, $\rho_\alpha(\mathbf{x}, t)$ is the convolution product between the functions:*

$$F(\mathbf{x}) = \int_{\Omega_2} h_\alpha(\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p})\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\boldsymbol{\xi}_\mathbf{p} \ \ \textit{and} \ \ G(\mathbf{x}) = \frac{e^{-\sum_{i=1}^n \frac{x_i^2}{2\sigma^2}}}{(\Phi(\mathbf{b}) - \Phi(\mathbf{a}))\sigma^{\frac{n}{2}}(2\pi)^{\frac{n}{2}}}.$$

where

$$h_\alpha(\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p}) = f(\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) + (t - f(\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}))_+ + \beta_2(f(\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) - t)_+, \tag{8}$$

Φ is the cumulative distribution function of multivariate Gaussian distribution and $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ are some vectors with $\mathbf{a} \leq \mathbf{b}$ defining the bounds of the multivariate truncated Gaussian distribution.

Proof. First, the probability density function for a truncated Gaussian distribution with support $[\mathbf{a}, \mathbf{b}] \subseteq \mathbb{R}^n$ of mean $\boldsymbol{\mu}$ and of standard deviation $\boldsymbol{\Sigma}$ is defined by:

$$G_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\mathbf{x}) = \frac{e^{-(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})}}{(\Phi(\mathbf{b}) - \Phi(\mathbf{a}))\det(\Sigma)^{\frac{1}{2}}(2\pi)^{\frac{n}{2}}}.$$

This equality may be rewritten as

$$\begin{aligned} G_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\mathbf{x}) &= \frac{e^{-\sum_{i=1}^n \frac{(x_i-\mu_i)^2}{2\sigma^2}}}{(\Phi(\mathbf{b}) - \Phi(\mathbf{a}))\sigma^{\frac{n}{2}}(2\pi)^{\frac{n}{2}}} \\ &= \frac{1}{\Phi(\mathbf{b}) - \Phi(\mathbf{a})}\prod_{i=1}^n G_{\mu_i,\sigma}(x_i). \end{aligned}$$

Then, by defining $\mathbf{X} = \mathbf{x} + \boldsymbol{\xi}_\mathbf{x}$ and since $\boldsymbol{\xi}_\mathbf{x}$ has a multivariate truncated Gaussian distribution, we obtain

$$\begin{aligned} \rho_\alpha(\mathbf{x}, t) &= \mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[f(\mathbf{x} + \boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) + (t - f(\mathbf{x} + \boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}))_+ + \beta_2(f(\mathbf{x} + \boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}) - t)_+] \\ &= \int_{\Omega_1\times\Omega_2} h_\alpha(\mathbf{x} + \boldsymbol{\xi}_\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p})\phi_{\boldsymbol{\xi}_\mathbf{x}}(\boldsymbol{\xi}_\mathbf{x})\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\boldsymbol{\xi}_\mathbf{x}d\boldsymbol{\xi}_\mathbf{p} \\ &= \frac{1}{\Phi(\mathbf{b}) - \Phi(\mathbf{a})}\int_{\Omega_1\times\Omega_2} h_\alpha(\mathbf{x} + \boldsymbol{\xi}_\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p})\prod_{i=1}^n G_{0,\sigma}(\xi_{x_i})\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\mathbf{X}d\boldsymbol{\xi}_\mathbf{p} \\ &= \frac{1}{\Phi(\mathbf{b}) - \Phi(\mathbf{a})}\int_{\Omega_1\times\Omega_2} h_\alpha(\mathbf{X}, t, \boldsymbol{\xi}_\mathbf{p})\prod_{i=1}^n G_{x_i,\sigma}(X_i)\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\mathbf{X}d\boldsymbol{\xi}_\mathbf{p} \\ &= \frac{1}{\Phi(\mathbf{b}) - \Phi(\mathbf{a})}\int_{\Omega_1\times\Omega_2} h_\alpha(\mathbf{X}, t, \boldsymbol{\xi}_\mathbf{p})\prod_{i=1}^n G_{0,\sigma}(x_i - X_i)\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\mathbf{X}d\boldsymbol{\xi}_\mathbf{p} \\ &= \frac{1}{\Phi(\mathbf{b}) - \Phi(\mathbf{a})}\int_{\Omega_1}\left(\int_{\Omega_2} h_\alpha(\mathbf{X}, t, \boldsymbol{\xi}_p)\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\boldsymbol{\xi}_\mathbf{p}\right)G_{\mathbf{0},\Sigma}(\mathbf{x} - \mathbf{X})d\mathbf{X}. \end{aligned}$$

Finally, the latter formulation may be seen as the convolution between the two following functions:

$$F(\mathbf{x}) = \int_{\Omega_2} h_\alpha(\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p})\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\boldsymbol{\xi}_\mathbf{p} \ \ \text{and} \ \ G(\mathbf{x}) = \frac{e^{-\sum_{i=1}^n \frac{x_i^2}{2\sigma^2}}}{(\Phi(\mathbf{b}) - \Phi(\mathbf{a}))\sigma^{\frac{n}{2}}(2\pi)^{\frac{n}{2}}}.$$

□

The following proposition gives an analytical way to calculate the derivatives of ρ_α, using mild conditions on the properties of the objective function.

Proposition 3. *Under Assumption 1 and Assumption 2, $\rho_\alpha(\mathbf{x},t)$ is infinitely continuously differentiable with respect to $\mathbf{x}$ and its partial derivatives are*

$$\forall i \in [1,n],\ \frac{\partial}{\partial x_i}\rho_\alpha(\mathbf{x},t) = \frac{1}{\sigma^2}\mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[\boldsymbol{\xi}_{x_i} h_\alpha(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p})].$$

with h_α defined in (8).

Proof. Given that $\rho_\alpha(\mathbf{x},t)$ may be seen as a convolution product and F is integrable, as the Gaussian function is infinitely continuously differentiable, $\rho_\alpha(\mathbf{x},t)$ is also infinitely continuously differentiable. Moreover, by noting $*$ the convolution product, the derivatives of $\rho_\alpha(\mathbf{x},t)$ are:

$$\begin{aligned}
\forall i \in [1,n],\ \frac{\partial}{\partial x_i}\rho_\alpha(\mathbf{x},t) &= \frac{\partial}{\partial x_i}\big(F * G\big)(\mathbf{x}) \\
&= \Big(F * \frac{\partial}{\partial x_i}G\Big)(\mathbf{x}) \\
&= \int_{\Omega_1}\left(\int_{\Omega_2} h_\alpha(\mathbf{X},t,\boldsymbol{\xi}_p)\phi_{\boldsymbol{\xi}_\mathbf{p}}(\boldsymbol{\xi}_\mathbf{p})d\boldsymbol{\xi}_\mathbf{p}\right)\frac{(X_i - x_i)}{\sigma^2}G_{\mathbf{0},\Sigma}(\mathbf{x}-\mathbf{X})d\mathbf{X} \\
&= \frac{1}{\sigma^2}\mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[h_\alpha(\mathbf{X},t,\boldsymbol{\xi}_\mathbf{p})(X_i - x_i)] \\
&= \frac{1}{\sigma^2}\mathbb{E}_{\boldsymbol{\xi}_\mathbf{x},\boldsymbol{\xi}_\mathbf{p}}[\xi_{x_i} h_\alpha(\mathbf{x}+\boldsymbol{\xi}_\mathbf{x},t,\boldsymbol{\xi}_\mathbf{p})].
\end{aligned}$$

□

These results are particularly interesting in an engineering context where the derivatives of f may be nonexistent or difficult to compute. Indeed, without conditions on the continuity of the objective function and knowing the univariate cumulative distribution functions of the uncertainties on the decision variables, Problem (6) may be seen as a minimization problem of an infinitely differentiable function whose the derivatives may be estimated. The next section will be dedicated to fully exploit these results to obtain an efficient minimization algorithm.

3 Stochastic approximation for Conditional Value at Risk optimization

In this section, an algorithm based on the stochastic approximation scheme [6, 7, 15] is presented to minimize the objective function in Problem (6). Then, practical considerations are addressed including the choice of sequences a^k and c^k, the choice of stopping criterion and the ease of implementation of the algorithm.

3.1 Risk-adverse Optimization by Stochastic Approximation (ROSA) algorithm

There are many different stochastic algorithms allowing to solve Problem (6). However, since f may be non-differentiable and hard to compute, SAA-type algorithms are not suitable. Indeed, the convergence properties of SAAs are based on regularity conditions of f and therefore might not converge. On the contrary, SA-type algorithms only require regularity conditions on the expectation of the objective function. With respect to $\mathbf{x}$, ρ_α is infinitely continuously differentiable since Propositions 2 and 3. With respect to t, ρ_α is piecewise linear and convex which ensures the Gateau differentiability of ρ_α in this direction. Thus, SA-type algorithm are well suited for Problem (6). However, the difference of regularity of ρ_α with respect to $\mathbf{x}$ and t leads to use different schemes to update the variables.

On the one hand, the Smoothed Functional (SF) scheme first introduced by [14, 28], seems particularly well suited for the decision variables $\mathbf{x}$ of Problem (6). This algorithm is derived from a convolution between the noisy objective function and a multivariate Gaussian distribution. The resulting algorithm has the remarkable feature of estimating the gradient of the objective function by simultaneously perturbing all the decision variables. This estimation requires only one or two evaluations of the objective function regardless of the problem dimension. For more details on SF-like algorithms, the reader is referred to [6, 16]. In this work, the objective function is already perturbed by Gaussian noised and its expectation is therefore a convolution product as proved in Proposition 2. The SF update strategy may be directly applied on the decision variables $\mathbf{x}$. The goal of the SF scheme is to track the asymptotic behavior of the following ordinary differential equation

$$\dot{\mathbf{x}}(u) = -\nabla_{\mathbf{x}}\rho_\alpha(\mathbf{x}(u), t(u)).$$

This is a ordinary differential equation, called Euler's equation, which converges to the set

$$H_{\mathbf{x}} = \{\mathbf{x} | \nabla_{\mathbf{x}}\rho_\alpha(\mathbf{x}, t) = 0\}.$$

On the other hand, the Stochastic Subgradient (SS) scheme [7] seems to be well suited for the decision variable t of Problem (6) since the directional derivatives with respect to t are known. The goal of the SS scheme is to track the asymptotic behavior of the following ordinary differential inclusion

$$\dot{t}(u) \in -\partial_t\rho_\alpha(\mathbf{x}(u), t(u))$$

where $\partial_t\rho_\alpha$ is the subdifferential of ρ_α with respect to t. This ordinary differential inclusion converges to the set

$$H_t = \{t | 0 \in \partial_t\rho_\alpha(\mathbf{x}, t)\}.$$

The ROSA algorithm is the combination of these schemes and thus leads to the set of critical point of ρ_α in which local minima are contained. The detailed version of the algorithm is given in Algorithm 1.

3.2 Practical considerations

In stochastic approximation, the choice of the step sizes a^k and c^k is critical for the efficiency of the algorithm. In practice, the step sizes allow to choose the best trade off between the exploration of the space of design variables and the efficiency of the local search. Assumption 3, although restrictive, leaves a large choice of sequences ensuring the convergence. That is why, from the very beginning, it has been the subject of numerous work for instance in [33] or more recently [13, 36]. However, these methods are not necessarily adapted to this work since the update of the artificial variable t is different. Thus, two different methods are used to choose the step sizes a^k and c^k.

Firstly, the step sizes a^k are used to update the decision variables $\mathbf{x}^k$. In this work, the optimal choice of a^k depends on three elements. First, as mentioned in [36], the step sizes depend on the ratio between the initial error $\mathbf{x}^0 - \mathbf{x}^*$, where $\mathbf{x}^*$ is the solution of the problem, and the level of noise $\boldsymbol{\xi}^{\boldsymbol{k}}$. The problem is that neither the initial error nor the level of noise is known in practice. However, this ratio $\tilde{r}$ may be estimated at $\mathbf{x}^0$ by sampling L different values of $\boldsymbol{\xi}$ and calculating with Equation (13). If the ratio between the initial error and the level of noise is large, then the step sizes a^k must be small in order to better ensure local convergence. On the contrary, if this ratio is small, then the step sizes must be large in order to better explore the space. Second, specifically in this work, the step sizes depend on the desired degree of robustness. Indeed, smaller is the value of α, greater is the value of β_2, so more important is the instability of the function h_α. Thus, the degree of robustness may be interpreted as an additional level of noise. Therefore, when α is near to 1, the step sizes must stay the same, while when α is close to 0, the step sizes must be reduced if it was large and increased if it was small. Finally, the step sizes may also depend on the dimension of the problem and on the maximal "size" of the compact set $\mathcal{C}$. In the common case where $\mathcal{C}$ is an hyperrectangle, its size is the difference between the lower bounds and the upper bounds. In this case, the dependency is clear: the step sizes must increase with the size of $\mathcal{C}$ and decrease with the dimension. These three rules lead to the heuristic described in Algorithm 2 when $\mathcal{C}$ is an hyperrectangle. They are used to set the initial

Algorithm 1 **Risk-Adverse by Stochastic Approximation algorithm (two measurements version)**

```
Initialization:
An iteration counter $k = 0$
A starting point $\mathbf{x}_0$, $t_1^0 = 1$ and $t_2^0 = -1$
A compact set $\mathcal{X}$
Two sequences $a^k$ and $c^k$ calculated with Algorithm 2
for k = 1, 2, ..., N do
    Simulate $\boldsymbol{\xi}_{\mathbf{x}^k} \in \mathbb{R}^n$ as a truncated Gaussian random vector
    Calculate
        $h_\alpha(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) = f(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^1_{\mathbf{p}^k}) + (t_1^k - f(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^1_{\mathbf{p}^k}))_+ + \beta_2(f(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - t_1^k)_+$
        $h_\alpha(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k}) = f(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^2_{\mathbf{p}^k}) + (t_2^k - f(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^2_{\mathbf{p}^k}))_+ + \beta_2(f(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^2_{\mathbf{p}^k}) - t_2^k)_+$        (9)
    Update $\mathbf{x}^k$
        $\mathbf{x}^{k+1} = \mathbf{x}^k - a^k \frac{h_\alpha(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - h_\alpha(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k})}{2} \boldsymbol{\xi}_{\mathbf{x}^k}$        (10)
    Update $t_1^k$ and $t_2^k$ if $k/10 = 0$ or $k > N/10$
        $t_1^{k+1} = t_1^k - c^k \tilde{g}(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k})$
        $t_2^{k+1} = t_2^k - c^k \tilde{g}(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k}),$        (11)
    where
        $\tilde{g}(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, t, \boldsymbol{\xi}_{\mathbf{p}}) = \begin{cases} -\beta_2 & \text{if} \quad t < f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \\ 0 & \text{if} \quad t = f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \\ 1 & \text{if} \quad t > f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \end{cases}$        (12)
    Project $(\mathbf{x}^{k+1}, t_1^{k+1}, t_2^{k+1})$ on $\mathcal{C} \subset \mathbb{R}^{n+2}$:
        $\mathbf{x}^{k+1}, t_1^{k+1}, t_2^{k+1} = \Pi_{\mathcal{C}}(\mathbf{x}^{k+1}, t_1^{k+1}, t_2^{k+1})$
end for
```

step size a^0. Then, the step sizes a^k are defined by Equation (16). This step sizes rule ensures to have the optimal convergence rate asymptotically and seems work well in practice with this choice of a^0.

Secondly, the step sizes c^k are used to update the additional variable t^k. As t is an additional variable, the choice of the optimal step sizes c^k do not depend on the structure of the problem but rather on the desired degree of robustness. That is why, in this work the initial step size is proportional to α and more or less large according to the value of $\tilde{r}$. With regard to the update of the step sizes c^k, it is different from a^k. Indeed, when α is small, i.e. β_2 is large, the variation of t^k from an iteration to another may be very important (see Equations (11) and (12)). That is why, the algorithm used in practice does not update the variable t^k at each iteration in the early step of the optimization process in order to ensure more stability (see Algorithm 1). So that the variations of t^k are not neglected during the following iterations, the step sizes c^k is chosen to decrease more slowly than the step sizes a^k (see Equation (16)).

Furthermore, the concern about the stopping criterion is a matter in for SA-based algorithms [6]. As many stochastic optimization algorithms, the convergence results are obtained at infinity. However in practice, the algorithm needs a stopping criterion. An interesting way to stop the algorithm from [6] is to calculate the mean norm between the k_i last iterates, for instance with $k_i = 10$. If this mean is inferior to a prespecified threshold then the algorithm is stopped. However in this work, the iterates t^k remains variable for a long time, especially when α is small. Therefore, a maximum number of evaluations $2N$ is used to stop the algorithm. This is coherent with the fact that the algorithm is designed for an engineering context where the number of evaluations is usually limited.

Last but not least, it is worth to notice that the technical implementation of this algorithm is only based on Algorithms 1 and 2. No additional library or code is necessary to obtain the numerical results

Algorithm 2 **The heuristic to choose a^k and c^k**

```
Sample L i.i.d truncated Gaussian vector $\boldsymbol{\xi}_{\mathbf{x}^0}^{(i)} \in \mathbb{R}^n$
Calculate the estimation of the ratio
    $\tilde{r} = \frac{1}{L}\sum_{i=1}^{L} \frac{|f(\mathbf{x}^0, \boldsymbol{\xi}_{\mathbf{p}^0}^{(i)}) - f(\mathbf{x}^0 + \boldsymbol{\xi}_{\mathbf{x}^0}^{(i)}, \boldsymbol{\xi}_{\mathbf{p}^0}^{(i)})|}{||T^R(\boldsymbol{\xi}_{\mathbf{x}^0}^{(i)})||}$    (13)
    where $T^R$ is the Rosenblatt transformation.
Set the first terms such that:
    $a^0 = \begin{cases} -\frac{4^{2+\log_{10}(\frac{\alpha}{\tilde{r}})}\max(u_i-\ell_i)}{\min(10,n)} & \text{if } \tilde{r} > 1 \\ \min\left(20\max(u_i-\ell_i), \frac{4^{2-\log_{10}(\alpha\tilde{r})}\max(u_i-\ell_i)}{\min(10,n)}\right) & \text{otherwise.} \end{cases}$    (14)
    $c^0 = \begin{cases} \alpha & \tilde{r} > 1 \\ 100\alpha & \text{otherwise.} \end{cases}$    (15)
Set
    $a^k = \frac{a^0}{k+1} \text{ and } c^k = \frac{c^0}{(k+1)^{0.501}}$    (16)
    for all $k \in \{1, ..., N\}$.
```

of the next section. This is perhaps the most interesting advantage for this algorithm to be used and tested by any type of audience.

4 Convergence analysis of the ROSA algorithm

This section studies the convergence of the ROSA algorithm to a minimum of the function ρ_α, i.e. a minimum of $\mathrm{CVaR}_\alpha(f())$. This study is mainly based on the convergence of stochastic approximation scheme.

4.1 Assumptions

In order to prove the convergence, the following assumptions hold.

Assumption 3. The step sizes a^k and c^k with $k \geq 0$ are positive and satisfy the requirements:

$$\sum_k a^k = +\infty, \sum_k c^k = +\infty \text{ and } \sum_k \max(a^k, c^k)^2 < +\infty.$$

Assumption 4. The algorithm is run on a hyperrectangle $\mathcal{C} = \mathcal{X} \times \mathcal{T}_1 \times \mathcal{T}_2 \subset \mathbb{R}^{n+2}$ with a provided projection operator $\Pi_{\mathcal{C}}$ and with $\mathcal{T}_1$ and $\mathcal{T}_2$ two segments of $\mathbb{R}$.

Now, by defining the following set

$$C(\mathbf{x}, t_1, t_2) = \begin{cases} \{0\} & \text{if } (\mathbf{x}, t_1, t_2) \in \mathrm{int}(\mathcal{C}) \\ \mathcal{N}_{\partial\mathcal{C}}(\mathbf{x}, t_1, t_2) & \text{if } (\mathbf{x}, t_1, t_2) \in \partial\mathcal{C} \end{cases} \tag{17}$$

where $\mathcal{N}_{\partial\mathcal{C}}(\mathbf{x}, t_1, t_2)$ is the the infinite convex cone generated by the outernormals at $(\mathbf{x}, t_1, t_2)$ of the faces on which $(\mathbf{x}, t_1, t_2)$ lies. Then, the projected ordinary differential inclusion, whose the iterates (10) and (11) try to track the asymptotic behavior, may be defined by

$$\begin{bmatrix} \dot{\mathbf{x}}(u) \\ \dot{t_1}(u) \\ \dot{t_2}(u) \end{bmatrix} \in \frac{1}{2}\begin{bmatrix} \nabla_{\mathbf{x}}\rho_\alpha(\mathbf{x}(u), t_1(u)) + \nabla_{\mathbf{x}}\rho_\alpha(\mathbf{x}(u), t_2(u)) \\ 2\partial_t\rho_\alpha(\mathbf{x}(u), t_1(u)) \\ 2\partial_t\rho_\alpha(\mathbf{x}(u), t_2(u)) \end{bmatrix} + \mathbf{z}, \quad \mathbf{z}(u) \in -C(\mathbf{x}, t_1, t_2), \tag{18}$$

where $\mathbf{z}$ is the projection term, i.e., the minimum force to keep $(\mathbf{x}, t_1, t_2)$ in $\mathcal{C}$.

Assumption 5. The (unknown) cumulative distribution function of $f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}})$ is continuous.

Assumption 3 is a common in stochastic algorithm [7, 15] to ensure that the algorithm does not stop before convergence. Assumption 4 may seem odd in a context of unconstrained optimization. It is mainly a practical assumption because it is usually hard to verify a priori the boundness of a function on a non compact set. Finally, Assumption 5 is hard to verify a priori, but is not very restrictive since it is equivalent to assume that $\forall x \in \mathbb{R}, \mathbb{P}(f(\mathbf{x}+\boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_p) = x) = 0$, i.e., that the function is not constant on large areas.

Now, before presenting the main results of this section, the algorithm update rules (10) and (11) will be modified to fit in with stochastic approximation schemes [15]. On the one hand, the update rule (10) may be transformed, for $i \in [1, n]$, as follows

$$\begin{aligned}
x_i^{k+1} = x_i^k - a^k \Bigg(& \xi_{x_i^k} \frac{h(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - h(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k})}{2} \\
& - \mathbb{E}_{\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}} \left[\xi_{x_i^k} \frac{h(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - h(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k})}{2} \right] \\
& + \mathbb{E}_{\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}} \left[\xi_{x_i^k} \frac{h(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - h(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k})}{2} \right] + Z_i^k \Bigg) \\
= x_i^k - a^k & \left(\frac{\nabla_\mathbf{x} \rho_\alpha(\mathbf{x}^k, t_1^k) + \nabla_\mathbf{x} \rho_\alpha(\mathbf{x}^k, t_2^k)}{2} + M_i^k + Z_i^k \right)
\end{aligned}$$

where the gradient appears as the result of Proposition 3,the noised sequence is

$$\begin{aligned}
M_i^k = & \xi_{x_i^k} \frac{h(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - h(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k})}{2} \\
& - \mathbb{E}_{\boldsymbol{\xi}_\mathbf{x}, \boldsymbol{\xi}_\mathbf{p}} \left[\xi_{x_i^k} \frac{h(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}^1_{\mathbf{p}^k}) - h(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}^2_{\mathbf{p}^k})}{2} \right].
\end{aligned} \tag{19}$$

and Z_i^k is the constraint term which take the iterates given by Equation (10) back to the hyperrectangle $\mathcal{C}$ if it is not in $\mathcal{C}$. On the other hand, the update rules (11) may be transformed as follows

$$\begin{aligned}
t_1^{k+1} &= t_1^k - c^k(\delta_1(\mathbf{x}^k, t_1^k) + M_{t_1}^k + Z_{t_1}^k) \\
t_2^{k+1} &= t_2^k - c^k(\delta_2(\mathbf{x}^k, t_2^k) + M_{t_2}^k + Z_{t_2}^k)
\end{aligned}$$

with

$$\begin{aligned}
\delta_1(\mathbf{x}^k, t_1^k) &= \mathbb{E}_{\xi_\mathbf{x}, \xi_\mathbf{p}}[\tilde{g}(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}_{\mathbf{p}^k})] \\
\delta_2(\mathbf{x}^k, t_2^k) &= \mathbb{E}_{\xi_\mathbf{x}, \xi_\mathbf{p}}[\tilde{g}(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}_{\mathbf{p}^k})]
\end{aligned} \tag{20}$$

and the noised sequences are

$$\begin{aligned}
M_{t_1}^k &= \tilde{g}(\mathbf{x}^k + \boldsymbol{\xi}_{\mathbf{x}^k}, t_1^k, \boldsymbol{\xi}_{\mathbf{p}^k}) - \delta_1(\mathbf{x}^k, t_1^k), \\
M_{t_2}^k &= \tilde{g}(\mathbf{x}^k - \boldsymbol{\xi}_{\mathbf{x}^k}, t_2^k, \boldsymbol{\xi}_{\mathbf{p}^k}) - \delta_2(\mathbf{x}^k, t_2^k).
\end{aligned} \tag{21}$$

In order to fit in with the general scheme, it is necessary to show that $\delta_1(\mathbf{x}, t)$ and $\delta_2(\mathbf{x}, t)$ are subgradients of ρ with respect to t, it is the subject of the next lemma.

Lemma 1. *For all $(\mathbf{x}, t) \in \mathcal{C}$, the following equalities hold*

- $\delta_1(\mathbf{x}, t) = \delta_2(\mathbf{x}, t) := \delta(\mathbf{x}, t)$
- $\delta(\mathbf{x}, t) \in \partial \rho_\alpha(\mathbf{x}, t)$

Proof. The first point comes from the symmetry of the probability density function $\phi(\boldsymbol{\xi}_\mathbf{x})$, in fact for $(\mathbf{x}, t) \in \mathcal{C}$:

$$\delta_1(\mathbf{x}, t) = \mathbb{E}_{\xi_\mathbf{x}, \xi_\mathbf{p}}[\tilde{g}(\mathbf{x} + \boldsymbol{\xi}_\mathbf{x}, t, \boldsymbol{\xi}_\mathbf{p})]$$

$$
\begin{aligned}
&= \int_{\Omega} \tilde{g}(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, t, \boldsymbol{\xi}_{\mathbf{p}}) \phi(\boldsymbol{\xi}_{\mathbf{x}}) \phi(\boldsymbol{\xi}_{\mathbf{p}}) d\boldsymbol{\xi}_{\mathbf{x}} d\boldsymbol{\xi}_{\mathbf{p}} \\
&= -\int_{\Omega} -\tilde{g}(\mathbf{x} - \boldsymbol{\xi}_{\mathbf{x}}, t, \boldsymbol{\xi}_{\mathbf{p}}) \phi(-\boldsymbol{\xi}_{\mathbf{x}}) \phi(\boldsymbol{\xi}_{\mathbf{p}}) d\boldsymbol{\xi}_{\mathbf{x}} d\boldsymbol{\xi}_{\mathbf{p}} \\
&= \delta_2(\mathbf{x}, t)
\end{aligned}
$$

because $\phi(\boldsymbol{\xi}_{\mathbf{x}}) = \phi(-\boldsymbol{\xi}_{\mathbf{x}})$. Thus, it remains to prove that $\delta(\mathbf{x}, t) \in \partial_t \rho_\alpha(\mathbf{x}, t)$.
Let $(\mathbf{x}, t_0) \in \mathcal{C}$, the random subdifferential $\partial_t h(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, t, \boldsymbol{\xi}_{\mathbf{p}})$ may be defined as:

$$
\partial_t h(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, t_0, \boldsymbol{\xi}_{\mathbf{p}}) \in \begin{cases} \{-\beta_2\} & \text{if} \quad t_0 < f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}), \\ [-\beta_2, 1] & \text{if} \quad t_0 = f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}), \\ \{1\} & \text{if} \quad t_0 > f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}). \end{cases}
$$

Moreover, since h is continuous and convex with respect to t for all $(\boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \in \Omega$ and ρ is proper, then by applying Theorem (7.47) of [31], for $(\mathbf{x}, t_0) \in \mathcal{C}$

$$
\begin{aligned}
\partial_t \rho(\mathbf{x}, t_0) &= \int_{\Omega} \partial_t h(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, t_0, \boldsymbol{\xi}_{\mathbf{p}}) d\mathbb{P}(\boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) + \mathcal{N}_{\mathcal{T}}(t_0) \\
&= \left\{ \int_{\Omega} \nu(\mathbf{x}, \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) d\mathbb{P}(\boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \mid \nu(\mathbf{x}, \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \in \partial_t h(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, t_0, \boldsymbol{\xi}_{\mathbf{p}}) \ a.s. \right\} + \mathcal{N}_{\mathcal{T}}(t_0)
\end{aligned}
$$

where ν are integrable multivalued functions and $\mathcal{N}_{\mathcal{T}}(t_0)$ is the normal cone of the segment $\mathcal{T}$. Since, in the algorithm, $\tilde{g}$ has been built such that it belongs to $\partial_t h(\mathbf{x}+\boldsymbol{\xi}_{\mathbf{x}}, t, \boldsymbol{\xi}_{\mathbf{p}})$ almost surely for any $(\mathbf{x}, t) \in \mathcal{C}$, then a direct consequence of this theorem is that $\delta(\mathbf{x}, t) \in \partial_t \rho(\mathbf{x}, t_0)$ almost surely for all $(\mathbf{x}, t) \in \mathcal{C}$. □

4.2 Analysis

The convergence of the algorithm is based on the theorem 6.2 of [15], to fill in the conditions of this theorem a sequence of lemmas and propositions are provided in the following order:

- Lemma 3 analyzes the martingale difference sequence associate with the update rules (11) and (10).
- Proposition 4 shows that the resulting martingale is almost surely convergent.
- Lemma 4 shows that the noise on the estimations of the gradient and subdifferential tends to zero when the number of iterations tends to infinity.

The formal proof is given finally in Theorem 1 where the different assumptions of the theorem 6.2 are stated from the previous results. In order to proof Lemma 3, the following result from [23] is used.

Lemma 2. *Let $\mathcal{A}$ be a sigma field, $\mathbf{X}$ and $\mathbf{Y}$ be some random vectors such that $\mathbf{Y}$ is independent of $\mathcal{A}$ and $\mathbf{X}$ is $\mathcal{A}$ measurable. Then, for all measurable bounded function Ψ*

$$
\begin{cases} \mathbb{E}(\Psi(\mathbf{X}, \mathbf{Y}) | \mathcal{A}) = \phi(\mathbf{X}) \\ \phi(\mathbf{x}) = \mathbb{E}(\Psi(\mathbf{x}, \mathbf{Y})). \end{cases}
$$

This Lemma will be used to prove that the noised sequences defined in Equations (19) and (21) are martingale difference sequences and have bounded second order moment.

Lemma 3. *Let M_i^k for $i \in [1, n]$ and M_t^k be the noised sequence defined respectively in Equations (19) and (21) and $\mathcal{F}^k = \sigma(\mathbf{x}^k, t_1^k, t_2^k, \boldsymbol{\xi}_{\mathbf{x}^k}, \boldsymbol{\xi}^1_{\mathbf{p}^k}, \boldsymbol{\xi}^2_{\mathbf{p}^k})$ be a sequence of associated sigma fields. Then, $(M_i^k, \mathcal{F}^k)$ and $(M_t^k, \mathcal{F}^k)$, $k \geq 0$, are martingale difference sequences, i.e. $\forall i \in [1, n]$:*

$$
\mathbb{E}[M_i^{k+1} | \mathcal{F}^k] = 0, \ \mathbb{E}[M_{t_1}^{k+1} | \mathcal{F}^k] = 0 \quad \text{and} \quad \mathbb{E}[M_{t_2}^{k+1} | \mathcal{F}^k] = 0.
$$

Moreover, the following inequalities hold

$$
\mathbb{E}[|M_i^{k+1}|^2 | \mathcal{F}^k] \leq K_i, \ \mathbb{E}[|M_{t_1}^{k+1}|^2 | \mathcal{F}^k] \leq K_{t_1} \quad \text{and} \quad \mathbb{E}[|M_{t_2}^{k+1}|^2 | \mathcal{F}^k] \leq K_{t_2}
$$

for some $K_i > 0$ and $K_t > 0$.

Proof.

Let $i \in [1, n]$. By noting

$$\begin{aligned}
\mathbf{X}^{k+1} &= [\mathbf{x}^{k+1}, t_1^{k+1}, t_2^{k+1}],\\
\mathbf{Y}^{k+1} &= [\boldsymbol{\xi}_{\mathbf{x}^{k+1}}, \boldsymbol{\xi}^1_{\mathbf{p}^{k+1}}, \boldsymbol{\xi}^2_{\mathbf{p}^{k+1}}],\\
\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1}) &= \xi_{x_i^{k+1}} \frac{h(\mathbf{x}^{k+1} + \boldsymbol{\xi}_{\mathbf{x}^{k+1}}, t_1^{k+1}, \boldsymbol{\xi}^1_{\mathbf{p}^{k+1}}) - h(\mathbf{x}^{k+1} - \boldsymbol{\xi}_{\mathbf{x}^{k+1}}, t_2^{k+1}, \boldsymbol{\xi}^2_{\mathbf{p}^{k+1}})}{2},
\end{aligned}$$

we have

$$\begin{aligned}
\mathbb{E}[M_i^{k+1}|\mathcal{F}^k] &= \mathbb{E}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1}) - \mathbb{E}_{\mathbf{Y}}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right]|\mathcal{F}^k\right]\\
&= \mathbb{E}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})|\mathcal{F}^k\right] - \mathbb{E}\left[\mathbb{E}_{\mathbf{Y}}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right]|\mathcal{F}^k\right].
\end{aligned}$$

Yet, $\mathbb{E}_{\mathbf{Y}}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right]$ is a constant conditionning on $\mathcal{F}^k$ since $\mathbf{X}^{k+1}$ is determined by $\mathbf{X}^k$ via the recursive Equations (10) and (11) and that the expectation is taken over $(\boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}})$, thus

$$\mathbb{E}\left[\mathbb{E}_{\mathbf{Y}}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right]|\mathcal{F}^k\right] = \mathbb{E}_{\mathbf{Y}}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right].$$

Furthermore, $\mathbf{X}^{k+1}$ is build from the variables in $\mathcal{F}^k$ and therefore is $\mathcal{F}^k$-measurable, the vector Y^{k+1} is independent of $\mathcal{F}^k$ and Ψ_i is a bounded measurable function because of Assumptions 1, 2 and 4. Then, the Lemma 2 is applied

$$\mathbb{E}\left[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})|\mathcal{F}^k\right] = \mathbb{E}_{\mathbf{Y}}[\Psi_i(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})]$$

and so

$$\mathbb{E}[M_i^{k+1}|\mathcal{F}^k] = 0.$$

The same reasoning may be applied by using $\mathbf{X}^{k+1}$ and $\mathbf{Y}^{k+1}$ as above and by defining

$$\begin{aligned}
\Psi_{t_1}(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1}) &= \tilde{g}(\mathbf{x}^{k+1} + \boldsymbol{\xi}_{\mathbf{x}^{k+1}}, t_1^{k+1}, \boldsymbol{\xi}^1_{\mathbf{p}^{k+1}}),\\
\Psi_{t_2}(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1}) &= \tilde{g}(\mathbf{x}^{k+1} + \boldsymbol{\xi}_{\mathbf{x}^{k+1}}, t_2^{k+1}, \boldsymbol{\xi}^1_{\mathbf{p}^{k+1}})
\end{aligned}$$

For all $t \in \{t_1, t_2\}$, $\mathbb{E}_{\mathbf{Y}}\left[\Psi_t(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right]$ is a constant conditioning on $\mathcal{F}^k$ and Ψ_t is a bounded measurable function by definition, thus by applying the Lemma (2), we have again:

$$\mathbb{E}[M_t^{k+1}|\mathcal{F}^k] = \mathbb{E}\left[\Psi_t(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})|\mathcal{F}^k\right] - \mathbb{E}\left[\mathbb{E}_{\mathbf{Y}}\left[\Psi_t(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1})\right]|\mathcal{F}^k\right] = 0.$$

Thus, $(M_i^k, \mathcal{F}^k)$, $(M_{t_1}^k, \mathcal{F}^k)$ and $(M_{t_2}^k, \mathcal{F}^k)$ are martingale difference sequences. The last inequalities arise from the fact that Ψ_i, Ψ_{t_2} and Ψ_{t_1} are bounded measurable functions. □

Now, let $\mathbf{M}^k = (M_1^k, ..., M_n^k, M_{t_1}^k, M_{t_2}^k)$, the sequence $(\mathbf{M}^k, \mathcal{F}^k)$ is a vector martingale sequence and the following process may be defined

$$\mathbf{Z}^k = \sum_{l=0}^{k-1} \max(a^l, c^l)\mathbf{M}^{l+1}.$$

Proposition 4. *The sequence $(\mathbf{Z}^k, \mathcal{F}^k), k \geq 0$, is a zero mean, square integrable and almost surely convergent martingale.*

Proof. By definition $\mathbf{Z}^k$ is $\mathcal{F}^k$-measurable. It is a zero mean integrable martingale because

$$\mathbb{E}[\mathbf{Z}^k] = \sum_{l=0}^{k-1} \max(a^l, c^l)\mathbb{E}[\mathbf{M}^{l+1}] = \sum_{l=0}^{k-1} \max(a^l, c^l)\mathbb{E}[\mathbb{E}[\mathbf{M}^{l+1}|\mathcal{F}^l]] = 0.$$

Moreover, $\mathbf{Z}^k$ is square integrable as consequence of the second part of Lemmma 3. Thus, the process B^k can be defined by

$$\begin{aligned}
B^k &= \sum_{l=0}^{k-1} \mathbb{E}(||\mathbf{Z}^{l+1} - \mathbf{Z}^l||^2|\mathcal{F}^l) \\
&= \sum_{l=0}^{k-1} \max(a^k, c^k)^2\ \mathbb{E}(||\mathbf{M}^{l+1}||^2|\mathcal{F}^l) \\
&= \sum_{l=0}^{k-1} \max(a^k, c^k)^2\ \mathbb{E}((M_1^{l+1})^2 + ... + (M_n^{l+1})^2 + (M_{t_1}^{l+1})^2 + (M_{t_2}^{l+1})^2|\mathcal{F}^l) \\
&\leq (n+1)\max(K_1, .., K_n, K_{t_1}, K_{t_2}) \sum_{l=0}^{k-1} \max(a^l, c^l)^2
\end{aligned}$$

by the second part of Lemmma 3. From Assumption 3, it follows that

$$B^k \to B^\infty < \infty$$

almost surely and the claim follows from the martingale convergence theorem (see Theorem B.2 of [6]). □

Then, let p^k be a sequence of positive real numbers defined such that

$$\begin{aligned}
p^0 &= 0 \\
p^k &= \sum_{l=0}^{k-1} \max(a^l, c^l)
\end{aligned}$$

and consider the function $m(p) = \max\{k \mid p^k \leq p\}$.

Lemma 4. *Let $P > 0$, then $\forall \epsilon > 0$*

$$\lim_{k\to\infty} \mathbb{P}\left(\sup_{j\geq k} \max_{p\leq P} \left|\left| \sum_{l=m(jP)}^{m(jP+p)-1} \max(a^l, c^l)\mathbf{M}^l \right|\right| > \epsilon \right) = 0.$$

Proof. Let (u^k) the sequence defined, for $k \geq 0$, by

$$\begin{aligned}
u^k &= \sup_{j\geq k} \max_{p\leq P} \left|\left| \sum_{l=m(jP)}^{m(jP+p)-1} \max(a^l, c^l)\mathbf{M}^l \right|\right| \\
&= \sup_{j\geq k} \max_{p\leq P} \left|\left| \mathbf{Z}^{m(jP+p)} - \mathbf{Z}^{m(jP)} \right|\right|.
\end{aligned}$$

The sequence (u^k) is positive and decreasing and $u^k \to 0$ when $k \to \infty$ because $\mathbf{Z}^k$ is an almost surely convergente martingale by Proposition 4 and $m(jP) \to \infty$ when $j \to \infty$ by Assumption 3. Moreover, as each $\mathbf{Z}^k$ is integrable, so is each u^k. Then, by applying the Markov inequality, we obtain

$$\mathbb{P}\left(u^k > \epsilon\right) \leq \frac{\mathbb{E}[u^k]}{\epsilon}.$$

Then by taking the limit and knowing that interchange the expectation and the limit is justified (apply the Monotone Convergence Theorem for decreasing sequence with integrable first term for instance), we conclude that

$$\lim_{k\to\infty} \mathbb{P}\left(\sup_{j\geq k} \max_{p\leq P} \left|\left| \sum_{l=m(jP)}^{m(jP+p)-1} \max(a^l, c^l)\mathbf{M}^l \right|\right| > \epsilon \right) = 0.$$

□

Finally, consider the sequence $t^k = \frac{t_1^k + t_2^k}{2}$ and the hyperrectangle $\mathcal{C}' = \mathcal{X} \times \mathcal{T} \subset \mathbb{R}^{n+1}$. The final theorem may be stated.

Theorem 1. *Under assumptions 1 to 5 and consider the ordinary differential inclusion*

$$\begin{bmatrix} \dot{\mathbf{x}}(u) \\ \dot{t}(u) \end{bmatrix} \in \begin{bmatrix} \nabla_{\mathbf{x}} \rho_\alpha(\mathbf{x}(u), t(u)) \\ \partial_t \rho_\alpha(\mathbf{x}(u), t(u)) \end{bmatrix} + \mathbf{z}, \quad \mathbf{z}(u) \in -C(\mathbf{x}, t) \tag{22}$$

Then, with probability one, all limit points of $(\mathbf{x}^k, t^k)$ are stationary points (i.e., points where $\mathbf{0}$ belongs to the right-hand side of (22)). Moreover, if there is a unique limit point (x^*, t^*) of the paths of (22) then $(x^k, t^k) \to (x^*, t^*)$ with probability one.

Proof. The proof of this theorem is made in two parts. The convergence of the iterates $\mathbf{x}^k, t_1^k$ and t_2^k is firstly stated, then we proof how it remains to the convergence of $\mathbf{x}^k$ and t^k.

The gradient $\nabla_{\mathbf{x}} \rho_\alpha(\mathbf{x}, t)$ is continuous, and thus bounded on $\mathcal{X}$, as a result of Proposition 3 under Assumptions 1 and 2. Moreover, by definition, the subgradients δ_1 and δ_2 defined in 20 are bounded by definition. Then, the theorem 6.2 of [15] may be applied using the result of Lemma 4.5 and Assumptions 3, 4 and 5. It states that, with probability one, the limit points of the iterates $(\mathbf{x}^k, t_1^k, t_2^k)$ obtained from (10) and (11) are stationary points of the ordinary differential inclusion (18). Moreover, if there is a unique limit point (x^*, t_1^*, t_2^*) of the paths of (18) then $(x^k, t_1^k, t_2^k) \to (x^*, t_1^*, t_2^*)$ with probability one.

Now, as ρ_α is convex along the direction t and that the cumulative distribution of f is continuous by Assumption 5, it is known [31] that the minimum of $\rho_\alpha(\mathbf{x}, t)$, over $t \in \mathbb{R}$, is attained at the left-side quantile $t^*_{\mathbf{x}}(\alpha) = \inf\{t : \mathbf{P}(f(\mathbf{x} + \boldsymbol{\xi}_{\mathbf{x}}, \boldsymbol{\xi}_{\mathbf{p}}) \leq t) \geq 1 - \alpha)$. Thus, the set of stationary point along the direction t

$$H_t = \{t | 0 \in \partial_t \rho_\alpha(\mathbf{x}, t) + z, \quad z \in C(\mathbf{x}, t)\}$$

contains a single element: $t^*_{\mathbf{x}}(\alpha)$ (provided that the bounds on t are chosen sufficiently large). So, the iterates t_1^k and t_2^k both converge to the same limit point $t^*_{\mathbf{x}}(\alpha)$ and by definition $t^k = \frac{t_1^k + t_2^k}{2}$ also converges to this point. The fact that $\nabla_{\mathbf{x}} \rho_\alpha(\mathbf{x}, t)$ is continuous allows to complete the proof. □

Remark 1. Note that the algorithm converges only to stationary points of the function ρ_α which may contain also saddle points or local maxima of the function. This may be particularly problematic in a context of minimization. In practice, as it has been discussed in Section 5.8 of [15], the local maxima and saddles points are often unstable and thus are not the limit points of the iterates. Intuitively, when the iterates converge to a non stable point, the noise is going to perturb the iterates which end by being attracted by a path leading to more stable point. However formally and without additional information on the noise, it is hard to avoid the convergence of the iterates to a non stable point. Readers are referred to Section 5.8 of [15] for the details.

5 Numerical experiments

In Section 5.1, data profiles [22] are adapted in the case of risk-adverse optimization. In Section 5.2, the algorithms and the test functions used for the comparison are presented. In Section 5.3, the results of this comparison are described and analysed.

5.1 Data profiles for risk-adverse optimization

Data profiles allow to assess if algorithms are successful in generating solution values close to the best objective function values. To identify a successful run, a convergence test is required. In a deterministic case, this test is based on the best function value found by the algorithm. However, when the function is stochastic, the best function values is unsuitable. In [9], the profiles are drawn from combinations of the mean and standard deviation of the stochastic objective function. The main drawback of this method is that it has no clear mathematical meaning. In this work, the mean and quantiles of the function under uncertainties are used for the convergence test. That allows to draw different graphs following the desired degree of robustness. For the mean, let μ^e be the best mean obtained by one algorithm on one problem after e evaluations, μ^0 be the mean of the function at the starting point and μ^* the best mean obtained by all tested algorithms on all run instances of that problem. Then, the problem is said to be solved in term of mean within the convergence tolerance τ when

$$\mu^0 - \mu^e \geq (1-\tau)(\mu^0 - \mu^*).$$

For the quantiles, let α be the desired degree of robustness and denote $\mathbf{x}_e$ the best iterates in term of quantile obtained by one algorithm on one problem after e evaluations, $t^*_{\mathbf{x}^0}(\alpha)$ the quantile of the function at the starting point $t^*_{\mathbf{x}^*}(\alpha)$ the quantile at the best solution obtained by all tested algorithms on all run instances of that problem. Then, the problem is said to be solved in terms of quantile of degree α within the convergence tolerance τ when

$$t^*_{\mathbf{x}^0}(\alpha) - t^*_{\mathbf{x}^e}(\alpha) \geq (1-\tau)(t^*_{\mathbf{x}^0}(\alpha) - t^*_{\mathbf{x}^*}(\alpha)).$$

An instance of a problem corresponds to a particular pseudo-random generator seeds. As the mean and the quantile of a function may be just an estimation, the number of samples used for the estimation must be adjusted to the desired convergence tolerance. In this work, we used 1000 independent identically distributed samples of $\boldsymbol{\xi}$ to estimate the mean and the quantile. Therefore, the convergence tolerance used are all greater than 0.01. Finally, the horizontal axis of a data profile represents groups of $n+1$ evaluations. The vertical axis corresponds to the proportion of problems solved within a given tolerance τ. Each algorithm has its curve to allow comparison of algorithms capability to converge to the best mean or quantile.

5.2 Algorithms and test problems used for the experiments

In order to underline the adaptability of the ROSA algorithm, two algorithms were chosen whose the treatment of uncertainties is opposite: the Robust-MADS algorithm [3] and the ROBOBOA algorithm [19]. Despite its name, Robust-MADS is a risk-neutral algorithm. To avoid ambiguity, we called it R-MADS in the following. Its goal is to optimize the convolution product between the uncertain objective function and a Gaussian kernel. When the decision variables are perturbed by Gaussian noised that amounts to look for the minimum of the expectation of the objective function (as it is seen in Proposition 2), i.e. solve Problem (1) with the setting described in the first example. The computational tests are conducted using version 3.9.1 of the NOMAD [17] software package. Conversely, ROBOBOA is a worst-case deterministic algorithm. Its goal is to optimize the maximum value of the objective function in the uncertainty set, i.e, solve Problem (1) with the setting described in the second example. It is based on the method of inexact method of outer approximations whose the principle is to solve alternatively an unconstrained minimization problem on the decision variables and a constrained maximization problem. The authors extend this method to derivative-free optimization and use manifold sampling to save computational burden of function evaluations. The computational tests are conducted using a personal implementation and with the advice of an author of ROBOBOA.

In order to draw a comparison between the three algorithms, 20 tests problems from the literature are used. These tests problems may be grouped into 7 classes of problems. Each problem is composed of an objective function and an distribution of the uncertainties. As the ROBOBOA is a deterministic algorithm, bounded distributions have been chosen in a way that the bounds of the distribution are the bounds of the uncertainty set for ROBOBOA. The vector $\boldsymbol{\xi}_{\mathbf{x}}$ and $\boldsymbol{\xi}_{\mathbf{p}}$ are composed of independent identically distributed variables except for the stochastic Rastrigin function where $\boldsymbol{\xi}_{\mathbf{x}}$ has dependent random variables in one of the problems. Here are the characteristics and the origin of the different test problems:

Stochastic Rosenbrock problem [1]

$$f_1(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) = \sum_{i=1}^{n-1}\left(10(x_i+\xi_{x_{i+1}}-(x_i+\xi_{x_i})^2)+\xi^1_{p_i}\right)^2+\left((1-(x_i+\xi_{x_i}))+\xi^2_{p_i}\right)^2.$$

Piecewise Continuous problem [20]

$$f_2(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) = 1-\prod_{i=1}^{2}\mathbf{1}_{\{x_i+\xi_{x_i}\geq 0\}}+\frac{1}{100}\sum_{i=1}^{2}(x_i+\xi_{x_i})^2$$

with $\mathbf{1}_{\{.\}}$ the indicator function.

Bertsimas problem [5]

$$f_3(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) = 2(x_1+\xi_{x_1})^2+\cdots-4.1(x_1+\xi_{x_1})(x_2+\xi_{x_2}).$$

Generator I problem [21]

$$f_4(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) = 1-\frac{3}{2\sqrt{2\pi}}e^{-2\sum_{i=1}^{2}(x_i+\xi_{x_i}-1.5)^2}-\frac{2}{\sqrt{2\pi}}e^{-50\sum_{i=1}^{2}(x_i+\xi_{x_i}-0.5)^2}.$$

Stochastic Powell problem [37]

$$\begin{aligned}f_5(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) = \sum_{i=1}^{n/4}\Big(&(x_{4i-3}+\xi_{x_{4i-3}}+10(x_{4i-2}+\xi_{x_{4i-2}}))^2+5(x_{4i-1}+\xi_{x_{4i-1}}-(x_{4i}+\xi_{x_{4i}})^2\\&+(x_{4i-2}+\xi_{x_{4i-2}}-2(x_{4i-1}+\xi_{x_{4i-1}}))^4+10(x_{4i-3}+\xi_{x_{4i-3}}-(x_{4i}+\xi_{x_{4i}})^4\Big)\\&+\xi_p\sqrt{1+100\sum_{i=1}^{n}(x_i-1)^2}.\end{aligned}$$

Stochastic Levy problem [37]

$$\begin{aligned}f_6(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) &= \sin^2(\pi w_1)+\sum_{i=1}^{n-1}(w_i-1)^2[1+10\sin^2(\pi w_i+1)]\\&\quad+(w_n-1)^2(1+\sin^2(2\pi w_n))+\xi_p\sqrt{1+10\sum_{i=1}^{n}(x_i-2)^2}\end{aligned}$$

with:

$$w_i = 1+\frac{x_i+\xi_{x_i}-1}{4}.$$

Stochastic Rastrigin problem [37]

$$\begin{aligned}f_7(\mathbf{x}+\boldsymbol{\xi},\boldsymbol{\xi}_\mathbf{p}) =&10n+\sum_{i=1}^{n}[(x_i+\xi_{x_i})^2-10\cos(2\pi(x_i+\xi_{x_i}))]\\&+\xi_p\sqrt{1+100\sum_{i=1}^{n}(x_i-1)^2}.\end{aligned}$$

In Table 1, $\mathcal{U}$ is the uniform distribution, $\mathcal{N}^{trunc}(0,1,[-3,3])$ the truncated Gaussian distribution, $\mathcal{U}^{disk}(0.5)$ the uniform distribution on a disk of radius 0.5, $\beta(2,2)$ the beta distribution, $Kuma(2,5,[-3,3])$ the Kumaraswamy distribution scaled on $[-3,3]$ and $FL(0.5,1,0)$ the Fatigue Life or Birnbaum-Sanders distribution.

Table 1: Problem parameters

	n	$\mathbf{x}^0$	Bounds	$\boldsymbol{\xi}_{\mathbf{x}}$	$\boldsymbol{\xi}_{\mathbf{p}}$	card($\boldsymbol{\xi}_{\mathbf{p}}$)
f_1	$\{2, 10, 50, 100\}$	$[-1.2, 1]^{n/2}$	$[-1.5, 1.5]^n$	$\mathcal{U}([-0.25, 0.25])$	$\mathcal{U}([-3, 3])$	$2n-2$
f_2	$\{2\}$	$[-7.5, -8.5]$	$[-10, 10]^2$	$\mathcal{N}^{trunc}(0, 1, [-3, 3])$	0	0
f_3	$\{2\}$	$[2.0, 2.0]$	$\begin{bmatrix} -1.2 & 3.2 \\ -0.5 & 4.5 \end{bmatrix}$	$\mathcal{U}^{disk}(0.5)$	0	0
f_4	$\{2\}$	$[0.8, 0.8]$	$[0, 2]^2$	$\mathcal{U}([-0.3, 0.3])$	0	0
f_5	$\{4, 12, 20, 40\}$	$[3.25, 4.6]^{n/2}$	$[-4, 5]^n$	$\beta(2, 2)$	$\mathcal{U}([-4, 4])$	1
f_6	$\{2, 10, 20, 50\}$	$[-7.2, 9.6]^{n/2}$	$[-10, 10]^n$	$Kuma(2, 5, [-3, 3])$	$\mathcal{U}([-3, 3])$	1
f_7^a	$\{2\}$	$[-4.6, -3.36]$	$[-5.12, 5.12]^2$	$\xi_{x_1} \sim \mathcal{U}[-0.5, 0.5]$ $\xi_{x_2} \sim \mathcal{U}[\xi_{x_1}, 1]$	$\mathcal{U}([-3, 3])$	1
f_7^b	$\{2, 10, 20, 50\}$	$[-4.6, -3.36]^{n/2}$	$[-5.12, 5.12]^n$	$FL(0.5, 1, 0)$	$\mathcal{U}([-3, 3])$	1

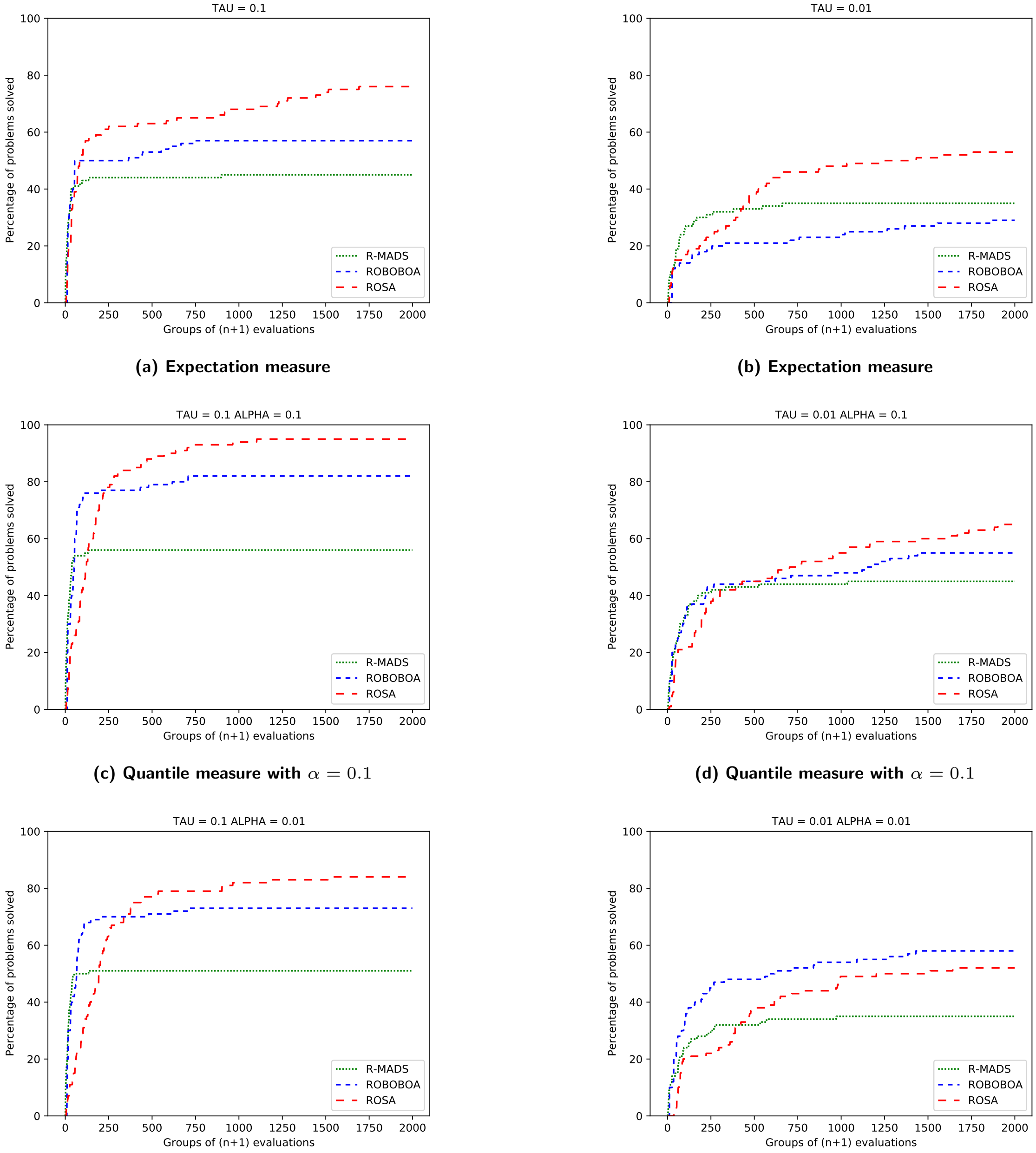
To the best of our knowledge, there are no comparisons between different algorithms in the risk-adverse optimization. As far as test problems are concerned, there are few that can be considered as references because they are not used in different papers. This selection of test problems from the literature fills this gap by creating a benchmark of test problems. It allows to test the algorithms on the key aspects of risk-adverse optimization in an engineering context. In particular, this selection has the following appealing properties:

- None of the functions used in the test problems has any particular mathematical structure. None of them is convex. The function Piecewise Continuous is not differentiable everywhere. Some functions also have the property of not having the same robust global minimum according to the desired degree of robustness. This means that the minimum found will be different if the optimization is in expectation or in the worst case. It is the case for example of the Bertsimas function or the Generator I function. Finally, the sizes of the test problems vary from small to large.
- Also, the uncertainty sets are diverse. In a deterministic setting, this means that the dimension of the uncertainty sets vary and that they are not only hyperrectangles. In a stochastic setting, this means that the distributions of uncertainties may be symmetric or not and may have dependency or not. The parameter uncertainties may also depend of $\mathbf{x}$.

5.3 Results

The data profiles have been computed using the previous benchmark. For each value of τ, the data profiles are plotted in terms of expectation, quantile of degree $\alpha = 0.1$ and quantile of degree $\alpha = 0.01$. In order to better illustrate the results according to the dimension of the problem, the benchmark of test problems have been divided into two groups: one with problem size inferior to 10 and another one with problem size strictly superior to 10. Each problem is run with ten different random seed to mitigate the randomness.

Results on test problems with dimension $n \leq 10$ are presented in Figure 2. The results for $\tau = 0.1$ are presented on the Figures 2a, 2c and 2e. In Figure 2a, i.e., for the optimization of the functions in expectation, the three algorithms have a similar efficiency with few evaluations. When a larger budget of evaluations is available, the ROSA algorithm solves the most problems, followed by ROBOBOA and finally by R-MADS. That may be counter intuitive since R-MADS is an algorithm designed for expectation minimization, but that seems due to the relatively large value of τ. In the case of optimization of quantile (Figures 2c and 2e), the same hierarchy between the three algorithms may be observed when a large number of evaluations is available. However, with few evaluations, it is worth to notice that R-MADS and ROBOBOA are more efficient than ROSA. The results for $\tau = 0.01$ are



(a) Expectation measure

(b) Expectation measure

(c) Quantile measure with $\alpha = 0.1$

(d) Quantile measure with $\alpha = 0.1$

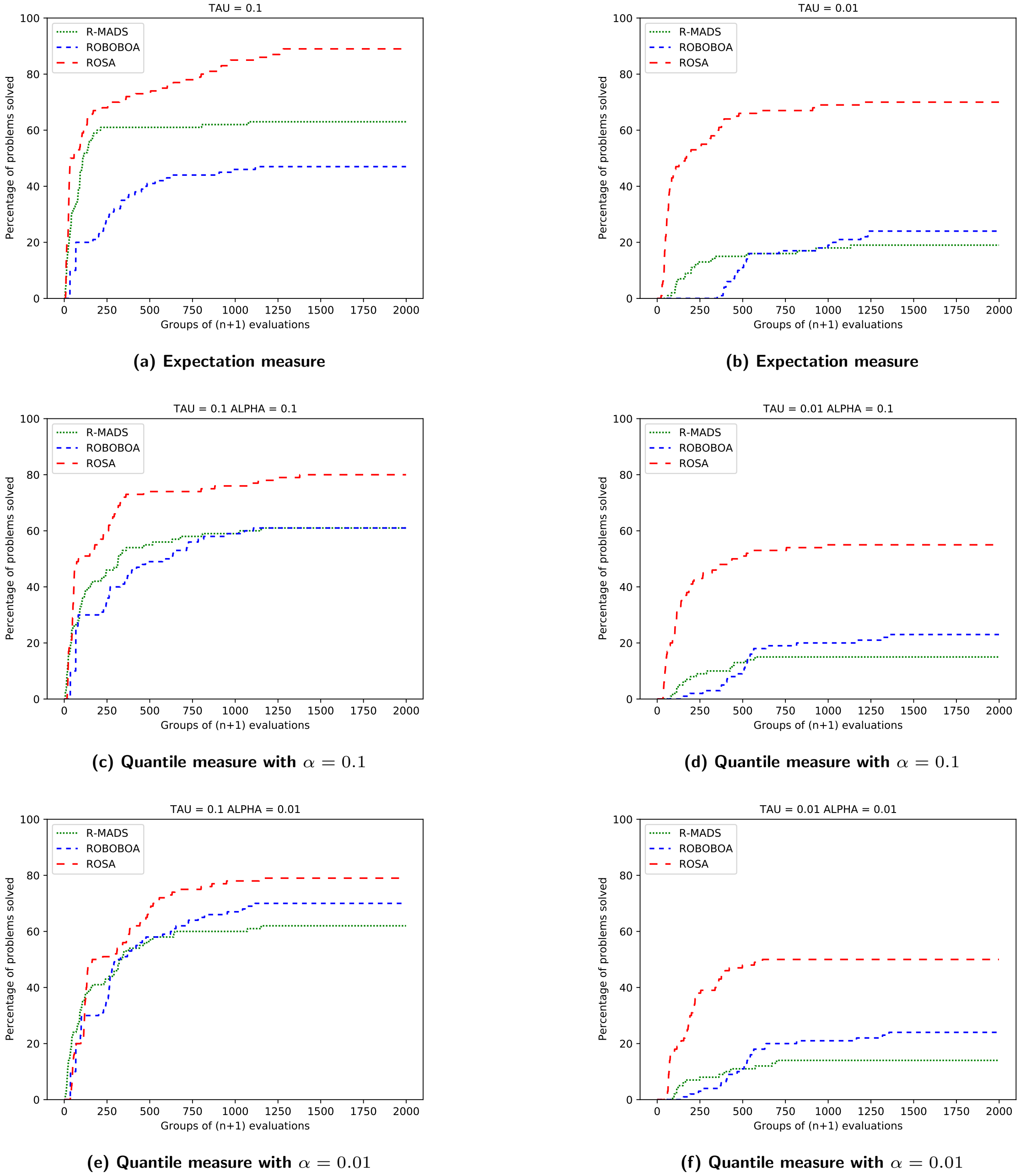
(e) Quantile measure with $\alpha = 0.01$

(f) Quantile measure with $\alpha = 0.01$

Figure 2: Results on test problems with $n \leq 10$ **for** $\tau = 0.1$ **(left) and** $\tau = 0.01$ **(right)**

presented on Figures 2b, 2d and 2f. First, the performance of R-MADS comparatively decreases when the degree of robustness increases and the contrary is observed for the ROBOBOA algorithm. This is consistent with what they were designed for. The ROSA algorithm remains the least efficient algorithm with few function evaluations but still has the highest percentage of problems solved, excepted for the highest degree of robustness where it is ROBOBOA.

Results on test problems with dimension $n > 10$ are presented in Figure 3. The results for $\tau = 0.1$ are presented on the Figures 3a, 3c and 3e. As on the smaller test problems, the ROSA algorithm has the best percentage of problems solved for all degrees of robustness, the difference is that here, it has the same efficiency than the two others algorithms with few function evaluations. Here again, the



(a) Expectation measure

(b) Expectation measure

(c) Quantile measure with $\alpha = 0.1$

(d) Quantile measure with $\alpha = 0.1$

(e) Quantile measure with $\alpha = 0.01$

(f) Quantile measure with $\alpha = 0.01$

Figure 3: Results on test problems with $n > 10$ **for** $\tau = 0.1$ **(left) and** $\tau = 0.01$ **(right)**

hierarchy between MADS and ROBOBOA is reversed when the degree of robustness becomes greater. The results for $\tau = 0.01$ are presented on the Figures 3b, 3d and 3f. In these results, ROSA clearly outperforms the two other algorithms. The well behavior of ROSA algorithm in larger dimensions may be explained by the fact that the number of function evaluations by iteration does not depend on the dimension of the problem to solve. On the contrary, R-MADS is a direct search algorithm, therefore the number of function evaluations by iteration increases with the dimension. The same happens for ROBOBOA which is a trust region algorithm whose the models use more points depending on the dimension.

In summary, on low dimensions test problems the ROSA algorithm is competitive whatever the degree of robustness. However, it is less efficient than specific algorithms with few functions evaluations. On high dimensions test problems, the ROSA algorithm outperforms the other algorithm whatever the degree of robustness thanks to the remarkable feature of SF scheme on which it is based and the fact that none quantile needs to be estimated during the process. The last point explains the large difference in term of function evaluations between ROSA and the algorithm developed in [37] while both are based on stochastic approximation scheme. This shows its adaptability on various test problems and with various degree of robustness.

6 Concluding remarks

This paper introduces a way for risk-adverse optimization under stochastic uncertainties in an engineering context where function evaluations are time consuming. This is achieved by using the CVaR_α risk measure. This measure has three main advantages. First, the α parameter of the CVaR_α measure allow to choose the desired degree of robustness, i.e to choose from risk free to worst-case optimization. Second, by adding an extra variable the CVaR_α optimization does not require any evaluations of quantile. Third, when the distribution of the uncertain decision variables are known, the measure allows to smooth the problem under mild conditions on the original objective function.

From these two features, the ROSA algorithm was developed. This algorithm is based on stochastic approximation schemes: the smoothed functional scheme and the subgradient approximation scheme. As a result, this algorithm inherits the appealing property of using only two function evaluations by iteration to estimate the gradient regardless of the problem dimension. A convergence proof to a minimum of CVaR_α of the objective function is also provided. This proof is based on martingale theory and needs only that the objective function be bounded and measurable on a compact.

Finally, a test problem benchmark collection is proposed. This set groups problems of different nature and allow to test algorithms on many situations that may arise in practice. The algorithm is compared with two derivative free algorithm: the first one is a risk-neutral optimization algorithm and the second one is a worst-case optimization algorithm. Thanks to the α parameter, the ROSA algorithm is competitive with the both algorithms in dimensions inferior to 10 and outperforms them in the other cases.

Further work will be devoted to extend this approach to chance-constrained risk averse optimization, commonly called reliability-based design optimization.

References

[1] Audet, C., Dzahini, K.J., Kokkolaras, M., Le Digabel, S.: Stochastic mesh adaptive direct search for blackbox optimization using probabilistic estimates. Computational Optimization and Applications 79(1), 1–34 (2021). DOI 10.1007/s10589-020-00249-0

[2] Audet, C., Hare, W.: Derivative-Free and Blackbox Optimization. Springer Series in Operations Research and Financial Engineering. Springer International Publishing, Cham, Switzerland (2017). DOI 10.1007/978-3-319-68913-5. URL https://dx.doi.org/10.1007/978-3-319-68913-5

[3] Audet, C., Ihaddadene, A., Le Digabel, S., Tribes, C.: Robust optimization of noisy blackbox problems using the Mesh Adaptive Direct Search algorithm. Optimization Letters 12(4), 675–689 (2018). DOI 10.1007/s11590-017-1226-6. URL https://doi.org/10.1007/s11590-017-1226-6

[4] Ben-Tal, A., Ghaoui, L.E., Nemirovski, A.: Robust Optimization. Princeton University Press, 41 William street, Princeton, New Jersey, USA (2009)

[5] Bertsimas, D., Nohadani, O., Teo, K.M.: Robust Optimization for Unconstrained Simulation-Based Problems. Operations Research 58(1), 161–178 (2010). DOI 10.1287/opre.1090.0715. URL http://pubsonline.informs.org/doi/abs/10.1287/opre.1090.0715

[6] Bhatnagar, S., Prasad, H., Prashanth, L.: Stochastic Recursive Algorithms for Optimization, *Lecture Notes in Control and Information Sciences*, vol. 434. Springer London, London (2013). DOI 10.1007/978-1-4471-4285-0. URL http://link.springer.com/10.1007/978-1-4471-4285-0

[7] Borkar, V.S.: Stochastic approximation: a dynamical systems viewpoint. Cambridge University Press ; Hindustan Book Agency, Cambridge, UK : New York : New Delhi (2008). OCLC: ocn231580915

[8] Cheng, J., Delage, E., Lisser, A.: Distributionally Robust Stochastic Knapsack Problem. SIAM Journal on Optimization 24(3), 1485–1506 (2014). DOI 10.1137/130915315. URL http://epubs.siam.org/doi/10.1137/130915315

[9] Ciccazzo, A., Latorre, V., Liuzzi, G., Lucidi, S., Rinaldi, F.: Derivative-Free Robust Optimization for Circuit Design. Journal of Optimization Theory and Applications 164(3), 842–861 (2015). DOI 10.1007/s10957-013-0441-2. URL http://link.springer.com/10.1007/s10957-013-0441-2

[10] Conn, A.R., Vicente, L.N.: Bilevel derivative-free optimization and its application to robust optimization. Optimization Methods and Software 27(3), 561–577 (2012). DOI 10.1080/10556788.2010.547579. URL http://www.tandfonline.com/doi/abs/10.1080/10556788.2010.547579

[11] Delage, E., Ye, Y.: Distributionally Robust Optimization Under Moment Uncertainty with Application to Data-Driven Problems. Operations Research 58(3), 595–612 (2010). DOI 10.1287/opre.1090.0741. URL http://pubsonline.informs.org/doi/abs/10.1287/opre.1090.0741

[12] Emmer, S., Kratz, M., Tasche, D.: What is the best risk measure in practice? A comparison of standard measures. The Journal of Risk 18(2), 31–60 (2015). DOI 10.21314/JOR.2015.318. URL http://www.risk.net/journal-of-risk/technical-paper/2434913/what-is-the-best-risk-measure-in-practice-a-comparison-of-standard-measures

[13] Grimmer, B.: Convergence Rates for Deterministic and Stochastic Subgradient Methods without Lipschitz Continuity. SIAM Journal on Optimization 29(2), 1350–1365 (2019). DOI 10.1137/18M117306X. URL https://epubs.siam.org/doi/10.1137/18M117306X

[14] Katkovnik, V.Y., Kulchits, O.Y.: Convergence of a class of random search algorithms. Automation and Remote Control 33(8), 1321–1326 (1972)

[15] Kushner, H.J., Yin, G., Kushner, H.J.: Stochastic approximation and recursive algorithms and applications. No. 35 in Applications of mathematics. Springer, New York (2003)

[16] Lakshmanan, K., Bhatnagar, S.: Quasi-Newton smoothed functional algorithms for unconstrained and constrained simulation optimization. Computational Optimization and Applications 66(3), 533–556 (2017). DOI 10.1007/s10589-016-9875-4. URL http://link.springer.com/10.1007/s10589-016-9875-4

[17] Le Digabel, S.: Algorithm 909: NOMAD: Nonlinear Optimization with the MADS algorithm. ACM Transactions on Mathematical Software 37(4), 44:1–44:15 (2011). DOI 10.1145/1916461.1916468. URL http://dx.doi.org/10.1145/1916461.1916468

[18] Li, W., Xiao, M., Garg, A., Gao, L.: A New Approach to Solve Uncertain Multidisciplinary Design Optimization Based on Conditional Value at Risk. IEEE Transactions on Automation Science and Engineering 18(1), 356–368 (2021). DOI 10.1109/TASE.2020.2999380. URL https://ieeexplore.ieee.org/document/9118975/

[19] Menickelly, M., Wild, S.M.: Derivative-free robust optimization by outer approximations. Mathematical Programming 179(1–2), 157–193 (2020). DOI 10.1007/s10107-018-1326-9. URL http://link.springer.com/10.1007/s10107-018-1326-9

[20] Mirjalili, S., Lewis, A.: Obstacles and difficulties for robust benchmark problems: A novel penalty-based robust optimisation method. Information Sciences 328, 485–509 (2016). DOI 10.1016/j.ins.2015.08.041. URL https://linkinghub.elsevier.com/retrieve/pii/S0020025515006301

[21] Mirjalili, S., Lewis, A.: Benchmark function generators for single-objective robust optimisation algorithms. In: Decision Science in Action, pp. 13–29. Springer (2019)

[22] Moré, J., Wild, S.: Benchmarking derivative-free optimization algorithms. SIAM Journal on Optimization 20(1), 172–191 (2009). DOI 10.1137/080724083. URL http://dx.doi.org/10.1137/080724083

[23] Revuz, D., Yor, M.: Continuous martingales and Brownian motion. Springer, Berlin; New York (2008). OCLC: 661004155

[24] Robbins, H., Monro, S.: A stochastic approximation method. The annals of mathematical statistics pp. 400–407 (1951)

[25] Robinson, S.M.: Analysis of Sample-Path Optimization. Mathematics of Operations Research 21(3), 513–528 (1996). DOI 10.1287/moor.21.3.513. URL http://pubsonline.informs.org/doi/abs/10.1287/moor.21.3.513

[26] Rockafellar, R.T., Royset, J.O.: Risk measures in engineering design under uncertainty. In: Proc. International Conf. on Applications of Statistics and Probability in Civil Engineering (2015)

[27] Rosenblatt, M.: Remarks on a multivariate transformation. The annals of mathematical statistics 23(3), 470–472 (1952)

[28] Rubinstein, R.Y. (ed.): Simulation and the Monte Carlo Method. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., Hoboken, NJ, USA (1981). DOI 10.1002/9780470316511. URL http://doi.wiley.com/10.1002/9780470316511

[29] Scarf, H.: A min-max solution of an inventory problem. Studies in the mathematical theory of inventory and production (1958)

[30] Shapiro, A., Homem-de Mello, T., Kim, J.: Conditioning of convex piecewise linear stochastic programs. Mathematical Programming 94(1), 1–19 (2002). DOI 10.1007/s10107-002-0313-2. URL http://link.springer.com/10.1007/s10107-002-0313-2

[31] Shapiro, A., Ruszczynski, A., Dentcheva, D.: Lecture on Stochastic Programming: Modeling and Theory. SIAM, Philadelphia, USA (2009)

[32] Spall, J.: Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Transactions on Automatic Control 37(3), 332–341 (1992). DOI 10.1109/9.119632. URL http://ieeexplore.ieee.org/document/119632/

[33] Spall, J.C.: Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control. John Wiley & Sons, Inc., Hoboken, NJ, USA (2003). DOI 10.1002/0471722138. URL http://doi.wiley.com/10.1002/0471722138

[34] Tyrrell Rockafellar, R., Royset, J.O.: Engineering Decisions under Risk Averseness. ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 1(2), 04015003 (2015). DOI 10.1061/AJRUA6.0000816. URL http://ascelibrary.org/doi/10.1061/AJRUA6.0000816

[35] Van Parys, B.P.G., Goulart, P.J., Morari, M.: Distributionally robust expectation inequalities for structured distributions. Mathematical Programming 173(1–2), 251–280 (2019). DOI 10.1007/s10107-017-1220-x. URL http://link.springer.com/10.1007/s10107-017-1220-x

[36] Xu, Z., Dai, Y.H.: New stochastic approximation algorithms with adaptive step sizes. Optimization Letters 6(8), 1831–1846 (2012). DOI 10.1007/s11590-011-0380-5. URL http://link.springer.com/10.1007/s11590-011-0380-5

[37] Zhu, H., Hale, J., Zhou, E.: Simulation optimization of risk measures with adaptive risk levels. Journal of Global Optimization 70(4), 783–809 (2018). DOI 10.1007/s10898-017-0588-8. URL http://link.springer.com/10.1007/s10898-017-0588-8