

# Fine-Tuned Robust Optimization

Alexandre Cesar Balbino Barbosa Filho<sup>a</sup>, Sergio Mauro da Silva Neiro<sup>b</sup>

a: Faculty of Chemical Engineering – Federal University of Uberlandia. Av. João Naves de Ávila, 2121 – Campus Santa Mônica – *Bloco 1K – Uberlandia, MG – Brazil*. E-mail: *alexandrecb.barbosa@ufu.br*

b: Faculty of Chemical Engineering – Federal University of Uberlandia. Av. João Naves de Ávila, 2121 – Campus Santa Mônica – *Bloco 1K – Uberlandia, MG – Brazil*. E-mail: *srgneiro@ufu.br*

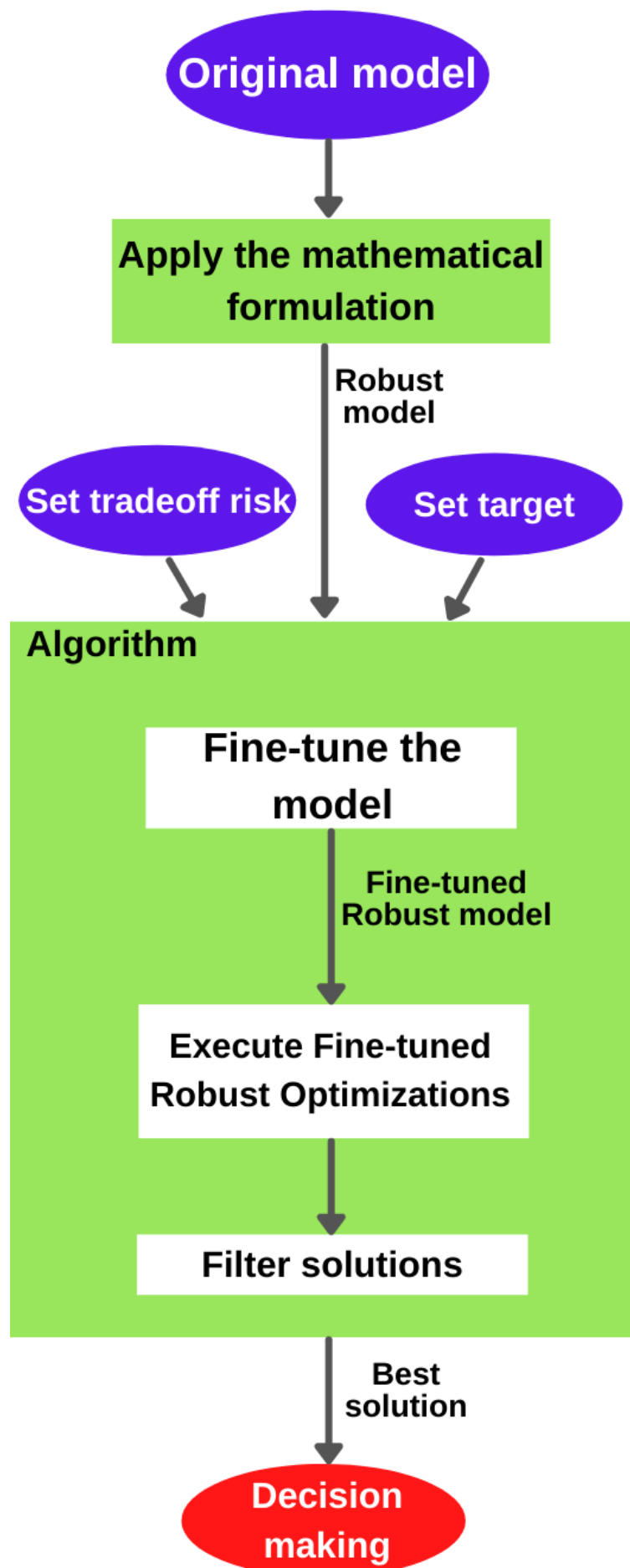
**Abstract:** This article provides a Robust Optimization framework with original concepts and fundamentals that combine ideals from relative regret models and static robust optimization. The algorithm uses a fine-tuning strategy to attain both robustness and ideality with a specified preset tradeoff risk. The framework comprises original concepts, a mathematical approach, and an algorithm. Statistical treatment based on the original framework concepts supports short, middle, or long-term decision-making settings. The framework is highly tractable as the algorithm forces the creation of a setting for a robust optimization in light of the specified risk. The framework can be applied in linear and nonlinear mathematical models as the objective function is monotonic in the domain of the active convex region. Several examples have been solved to gain a better understanding of the framework, and all results have showed high tractability and performance. The range of applications is wide. This manuscript provides an in-depth discussion of its philosophy, objective, original concepts, fields of application, as well as statistical and probabilistic fundamentals.

Keywords: Robust Optimization; Optimization Under Uncertainty; Robustness; Stochastic; Robust Solution.

## HIGHLIGHTS

- 1- The framework is highly tractable; the algorithm forces fine-tuning to grant feasible and tractable solutions with preset tradeoff risk.
- 2- The solution naturally approaches an ideal target while attaining robustness.
- 3- Decision making does not ignore the standard, nor the recent trend, increasing the feasibility of put it to practice.
- 4- The framework performs short, middle or long-term settings for decision making, which leads to outstanding long-run average performance.
- 5- Risk aversion is not always the best option for decision making when choosing uncertain values close to the mean.

## GRAPHICAL ABSTRACT



## 1. Introduction

A classical concern in mathematical programming is the fact that some constraint of a model may be violated if the input data vary, thereby producing an infeasible solution. Robust Optimization (RO) was first developed to address this concern and generate feasible solutions. Over the past years, RO has been used to handle optimization problems containing uncertainties in a model's input parameters, because it is practical and powerful to apply to real applications. Different concepts have been developed independently in this field, all of which cope with variation in the values of a model's input data, albeit with different RO objectives.

On account of the need to arise models that are immune to perturbations in the input data, Soyster (1973) was the first to provide an RO method containing the ideals of conservatism and robustness through the concept of robust counterpart. In this approach, static RO is applied to linear programming (LP) problems with  $j$  constraints and parameters represented as hyperspheres with center  $a_{ij}$  and radius  $\rho_i$  in the Euclidean space.  $\mathbf{e}$  is the vector of all-ones, and  $\mathbf{b}$  is the right-hand side. This RO method makes an ultraconservative adaptation to the convex feasible region to protect the solution from infeasibility (Eq. 1) for any perturbation in the interval  $[a_{ij} - \rho_i, a_{ij} + \rho_i]$ .

$$\min \mathbf{c}^T \mathbf{x}, \text{ s. t. } \mathbf{x}_1(\mathbf{a}_1 + \rho_1 \cdot \mathbf{e}) + \dots + \mathbf{x}_n(\mathbf{a}_n + \rho_n \cdot \mathbf{e}) \leq \mathbf{b} \quad (1)$$

Conservatism-based RO aims to improve the performance of a deterministic model by changing the convex feasible region of the original model so that it can provide feasible optimal solutions. Generally, this methodology does not use scenarios nor probability distributions. The main challenge of this methodology is to balance conservatism and robustness: as robustness increases, so does conservatism, and the solution becomes more deviated from a nominal solution by the deterministic model. Based on this challenge, studies based on different approaches have been developed over the past years, including El Ghaoui & Lebret (1997), El Ghaoui et al. (1998), Ben-Tal & Nemirovski (1998, 1999, 2000), and Bertsimas & Sim (2004).

To avoid over-conservatism, these approaches have changed the way the parameters are represented in the Euclidean space and solved the problem as either conic or quadratic. El Ghaoui & Lebret (1997) focused on least-squares problems to minimize worst-case residuals considering unknown-but-bounded uncertainties. El Ghaoui & Lebret (1998) developed an approach with uniform distributions for the uncertainties to quantify the effect of unknown-but-bounded deterministic perturbation of problem data in solutions from uncertain semidefinite programs. Constituting data as ellipsoidal in the Euclidean space, Ben-Tal & Nemirovski (1998) developed an RO approach to convex problems with unknown-but-bounded data immunized for all perturbations. Ben-Tal & Nemirovski (1999) replaced an LP problem by its robust counterpart (Eq. 2) to solve all *hard* constraints, and to make a tractable approach they contemplated the data as an ellipsoidal uncertainty set  $\xi$  to facilitate the optimization. Despite being convex, all these approaches considering either conic or quadratic problems lead to nonlinear models, adding complexity; therefore, increasing computational efforts and requiring more time to solve.

$$\min\{\mathbf{c}^T \mathbf{x} | \mathbf{A}\mathbf{x} \geq \mathbf{b} \forall (\mathbf{A}, \mathbf{b}) \in \xi\} \quad (2)$$

Ben-Tal & Nemirovski (2000) solved over 90 LPs with RO methods such as those developed by El Ghaoui & Lebret (1997), El Ghaoui et al. (1998), Ben-Tal & Nemirovski (1998, 1999). They showed the importance of RO for optimality, but pointed to the fact that small data perturbations can lead to infeasible solutions. The approaches provided by Ben-Tal & Nemirovski (1998, 1999, 2000) cannot be directly applied to discrete optimization (Assavapokee et al. 2008). All these cited approaches neglect probability distributions.

Bertsimas & Sim (2004) proposed an RO approach to control the degree of conservatism in the solution by terms of probabilistic bounds of constraint violation. To adjust the model's robustness, this approach also admits introducing a prespecified number  $\Gamma_i$  in the  $i_{th}$  probabilistic constraint. The cumulative distribution function is a function of  $\Gamma_i$ . The role of  $\Gamma_i$  is to adjust the robustness. The probabilistic bounds of constraints are also function of  $\Gamma_i$ . First, to make a linear optimization, the approach solves the linear equivalent formulation of the nonlinear original model. The problem considers an uncertainty  $\tilde{a}_{ij} \in J_i$ ,  $J_i$  as the uncertainty set, generated by a symmetric distribution with mean  $a_{ij}$  in the interval  $[a_{ij} - \hat{a}_{ij}, a_{ij} + \hat{a}_{ij}]$ .  $a_{ij}$  can change up to  $\hat{\Gamma}_i$ , and a coefficient  $a_{it}$  changes by  $(\Gamma_i - \hat{\Gamma}_i)\hat{a}_{it}$ . This approach considers different protection levels as part of its robust counterpart and as a function of a vector of variables, coefficients, and  $\Gamma$ .

Unlike the notion of robust counterpart first approached by Soyster (1973), Kouvelis & Yu (1997) considered robustness based on the notion of relative regret model. They admitted that a solution  $z(x_\xi, \xi)$  is robust as closer as it gets to an ideal solution  $z(\mathbf{x}, \xi)$ . This method solved all scenarios  $\forall \xi \in E$  one by one by minimizing the maximum regret  $f^*$  (Eq. 3).

$$\min_{\xi} f^*, s.t. z(\mathbf{x}, \xi) - z(x_\xi, \xi) \leq f^* ; f^* \geq 0; \xi \in E \quad (3)$$

Relative regret models-based RO aims to find an optimal solution for a model that considers the maximum regret as an objective function, i.e., to minimize the maximum regret, which is the difference between the optimal nominal solution and the solution of a scenario. The main challenge of this methodology is the high computational effort required as the number of scenarios increases. Research on different approaches has been explored over the past years to tackle this challenge, including Assavapokee et al. (2008), Baohua & Shiwei (2009), Jiang et al. (2013), Chen et al. (2014), and Xidonas et al. (2017).

Assavapokee et al. (2008) report on an approach to LP focusing on solving large-scale min-max regret and min-max relative regret RO problems under ambiguity for two-stage decision making. The first-stage problem must be a MILP model, and the second-stage problem must be an LP model. The methodology is intended to consider a finite scenario set by searching the best scenario among all possible realizations.

Xidonas et al. (2017) developed a min-max regret approach. The idea was to extend the min-max regret criterion to multiobjective robust portfolio optimization focusing on its application to portfolio management.

The regret-based robustness can be applied to Pareto solutions and has managerial usability to investment practitioners. In addition, other researchers have extended specific adaptations of regret models to related fields of engineering, including Baohua & Shiwei (2009), Jiang et al. (2013), Chen et al. (2014).

New concerns have emerged from the ideal of RO. Distributional RO, a class of approaches designed to cope with ambiguous probability distributions, generally considers chance-constraints and is usually applied in stochastic programs. Adaptive RO is a class of approaches designed to balance conservatism and robustness through multiple decision-making stages by adjusting some *wait and see* variables and adapting the robust counterpart of the problem. All these models are not related to the present framework, which is a static RO aiming at a maximization or minimization problem. This new approach may be called fine-tuned robust optimization.

Optimization under uncertainty and lack of information on probability distribution (e.g., RO without historical data) is deprived from historical trends to support decisions for the long run. The first steps in the present framework are aimed at a statistical treatment of the data with the original concepts of the framework, essential for performing since the short-term, the middle-term, and long-term setting for decision making which leads to outstanding long-run average performance.

No static RO framework has merged regret models, conservatism, and robustness. However, such a merger supports both a robust decision making and a target ideality. Two mutual benefits that reduce conservatism naturally by approaching a target ideality. In addition, if this decision making takes the most recent events as a starting point, a solution closer to ideality is achieved without ignoring the current tractability of operation, which is another basis for the present framework.

Based on the notions of conservatism and robustness, robust solutions deviate from a nominal solution, but if a regret model is added to the framework, the solution naturally approaches an ideal target at the same time that the balance of robustness is attained (as shown in section 3). Decisions based on events far from the present reality may not be feasible; therefore, when there are robust solutions based on recent trends, there is a greater chance of consistency in taking action because they are the occurrences that can be repeated or slightly modified, i.e., the decision making does not ignore the standard, which increases the feasibility of putting it into practice. To this end, it is necessary to consider some original concepts, statistical and probabilistic fundamentals of this framework, as well as probability distributions. These are the pillars of the present framework, which incorporates disjoint probability distributions.

The proposed framework is composed of a mathematical approach and an algorithm to make RO based on recent historical data and original concepts. Its concepts combine conservatism and robustness and are aimed to find a solution that draws near to a target solution (an ideal solution/perfect information), two concepts of RO in one. Balance between conservatism and robustness is achieved by adding a penalty factor that multiplies the expectance of an uncertainty. To perform this methodology, it is necessary to have at least one uncertainty that follows a normal distribution because the algorithm finds the best standard deviation values for each normal uncertainty; after that, static RO can be carried out in the final step of the algorithm to

find the best solution. This final robust solution is used for decision making based on a stochastic procedure to avoid exhaustive computational effort in the final algorithm step.

In this approach, epistemic uncertainties are also considered to have a normal distribution as it is subjective and naturally endowed by the central limit theorem. The framework serves for linear and nonlinear models under certain axioms, but the objective function (OF) must be monotonic in the domain of the convex feasible region (DCFR). The OF can be non-monotonic out of the DCFR. This framework can be applied to discrete and continuous optimization and to semidefinite programming (SDP) as all LP can be expressed as SDPs.

The framework of this present paper does not include probabilistic constraints, making it an easier resolution without needing to solve integrals or creating nonlinear constraints by introduction of cumulative distribution functions. Eventually, this increases the robustness of the problem by avoiding nonlinearity if there is none.

The present framework is highly tractable since the algorithm always forces fine-tuning to ensure feasible and tractable solutions with a specified risk. It embeds conservatism and robustness according to the risk specification to avoid using only the worst cases or cases generally delimited by some hyperplane geometry. Section 2 profoundly discusses the framework. Section 3 provides several linear and nonlinear examples that were solved using the framework. Section 4 concludes this paper.

## **2. Framework**

This section discusses the framework, namely its philosophy, objective, original concepts, fields of application, statistical and probabilistic fundamentals, mathematical approach, and algorithm. Before discussing why and how the procedures were performed, it provides their fundamentals and definitions in section 2.1. The general procedure consists of first performing a statistical treatment of data (section 2.1), then transforming (adapting) the original model into another (section 2.2), and finally running the algorithm to fine-tune the model and reach robust optimization.

### **2.1. Objective, philosophy, and first remarks**

To carry out an RO through this new approach that balances robustness and conservatism, the present framework incorporates an algorithm, a regret model with its own definitions and concepts that decrease the relative regret, and a mathematical formulation that makes the deterministic model more robust to parametric variations. To perform the RO, it is necessary to transform the original model into another, a robust model with penalty factors, through the mathematical formulation of the framework; finally, it is necessary to fine-tune the values of all penalty factors and standard deviations of normal uncertainties through an algorithm.

The philosophy of the framework is based on performing an RO which disregards discrete scenarios or a tree of scenarios and which is based on both the representation of random uncertainties by disjoint

probability distributions and the normal distribution for normal uncertainties or for epistemic uncertainties, if any.

In this framework, data are statistically treated to provide a tractable and flexible problem that considers the decision making from the short-term to the middle and long run because a data timeline is chosen. The following definitions for data treatment also increase tractability because it avoids the use of scenarios in the mathematical modeling and because the resulted decision making does not ignore the standard, nor the recent trend, increasing the feasibility of put it into practice. This also makes it possible to work with a relative regret model without the need to generate tree scenarios for the set of problem parameters or to aim at an exhaustive resolution of all of them as generally done in relative regret models. First, it is necessary to understand the general deterministic optimization of reference (GDOR<sup>1</sup>) and the recent reference timeline (RRT<sup>2</sup>), because they provide the overall limitation of the problem, i.e., the decision making will be built on the mean of historical data in the short, middle or long run as chosen by the decision maker. It must be chosen by the decision maker because it is necessary to decide whether it is going to be an RO based on the short, middle, or long run. The definitions and concepts used in the framework are as follows:

- i) GDOR: It is an optimization guide to determine the target ideal solution if it is not specified. Following this guide is important if the decision maker does not have in mind what the target ideal solution is like. The GDOR is based on a nominal value for each parameter, in such a way that this nominal value is an average of the nominal values that have already been used in optimizations carried out in the past. Historical GDOR data are the historical data chosen for each parameter by the decision maker; they are available in the timeline desired by the decision maker – e.g., the general reference timeline chosen may be the historical data of optimizations performed in the last 2 years or all data available on the total timeline. The objective function (OF) of the GDOR is the target ideal solution ( $f_{obj}^{GDOR}$ ) for the optimization problem.
- ii) RRT: It is the timeline considering a nominal value for each parameter, in such a way that this nominal value is an average of the nominal values that have been used in optimizations carried out in the most recent past. It refers to the most recent historical data as chosen by the decision maker. Historical RRT data are a subset of historical data from the GDOR timeline; they have a more recent timeline and were used in optimizations carried out during that RRT – e.g., historical optimization data from the last monthly planning or those used in the last two monthly planning optimizations in a given industry. The nominal value of each parameter that is not an uncertainty are, in the RRT, those that are actually used in the RO.

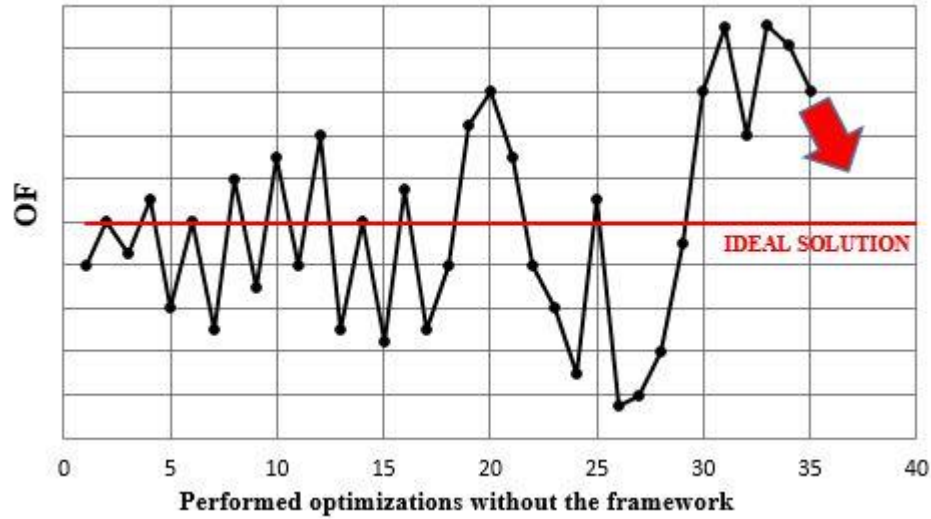
As the RRT embeds the most recent historical trend, this is the most likely or feasible standard to be repeated or slightly modified to happen. As such, it is reasonable that the decision making starts from this

<sup>1</sup> GDOR: General Deterministic Optimization of Reference – an optimization guide.

<sup>2</sup> RRT: Recent Reference Timeline: a recent part of GDOR's timeline



reference, i.e., from the average of these recent historical data rather than from imaginable, unpredictable, or subjective scenarios. The robust model fine-tuned by the framework will thus provide robustness to both the model and the optimization problem under uncertainty, while establishing optimal decision making based on recent conditions (RRT). Thus, one can reach as close as possible to an ideal solution when having normal uncertainties in the model, as illustrated by the arrow in Fig. 1. The red line in Fig. 1 refers to the hypothetical ideal solution  $f_{obj}^{GDOR}$ .



**Fig.1.** The red arrow points to where the solutions of the framework will lie, i.e., close to the target ideal solution  $f_{obj}^{GDOR}$ .

Let us take a simple example to show the use of the GDOR and the RRT. Let us assume that one uncertainty is the market price of something. Therefore, this uncertainty in the GDOR would be the price that would lead to the ideal target solution  $f_{obj}^{GDOR}$ . However, as the closest period of time is the RRT, it may not be feasible to use the GDOR price for the period of the process. One occurrence is the inflation price of any product that generally increases along the timeline; hence, it is more feasible to use the price of the RRT and determine decisions that lead to the ideal target solution considering recent conditions based on recent trends.

- iii) Relative regret (RR (%)): How much the OF value of a performed optimization with RRT data ( $f_{obj}^{rob(RRT)}$ ) distances in absolute value from the OF value of the ideal target solution ( $f_{obj}^{GDOR}$ ).

Relative regret is calculated in terms of percentage deviation according to Eq. (4):

$$RR (\%) = \frac{|f_{obj}^{GDOR} - f_{obj}^{rob(RRT)}|}{f_{obj}^{GDOR}} 100 \quad (4)$$

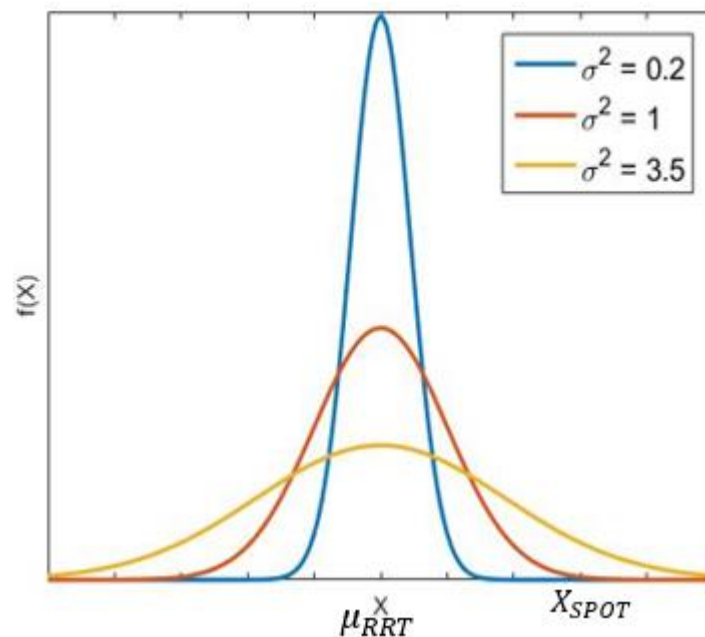
- iv) Average relative regret (%) (ARR): It is the arithmetic mean of the relative regrets. The ARR value is used in the criterion for choosing the fine-tuned standard deviation value of an uncertainty that follows a normal distribution, as will be seen in the algorithm (section 2.4).



## 2.2. Statistical and probabilistic fundamentals of the framework

Among all possible values for a normal uncertainty, there will always be a specific value for a normal uncertainty that makes the solution as close as possible to the ideal solution. This value is defined as  $X_{SPOT}$ . Because monotonic OF preserves the order relation, the values closer to the  $X_{SPOT}$  value will always provide the values that are the closest possible to the ideal solution. Additionally, monotonic objective functions incorporate the principle of superposition; thus, each normal uncertainty will have an independent individual contribution to the problem, i.e., an already fine-tuned standard deviation value will not affect the fine-tuning of another standard deviation. Even for nonlinear models, the OF is monotonic in the DCFR. Therefore, fine-tuning a standard deviation will not affect the fine-tuning of another.

The best standard deviation for a normal uncertainty is the one that provides the greatest chance (best chance on average) for the random value of the normal uncertainty to be generated as close as possible to  $X_{SPOT}$ . In Fig. 2, the value of uncertainty  $X$  – which, once substituted in the mathematical model, would cause  $f_{obj}^{rob(RRT)}$  to draw as near as possible to  $f_{obj}^{GDR}$  – is  $X_{SPOT}$ , and the fine-tuned bell curve would be the one with the highest probability of this value being generated randomly, which in this case would be  $\sigma = \sqrt{3.5}$  (orange curve). Because it is a Monte Carlo simulation, the greater the number of samples, the more statistically accurate the final results will be, due to the law of large numbers.



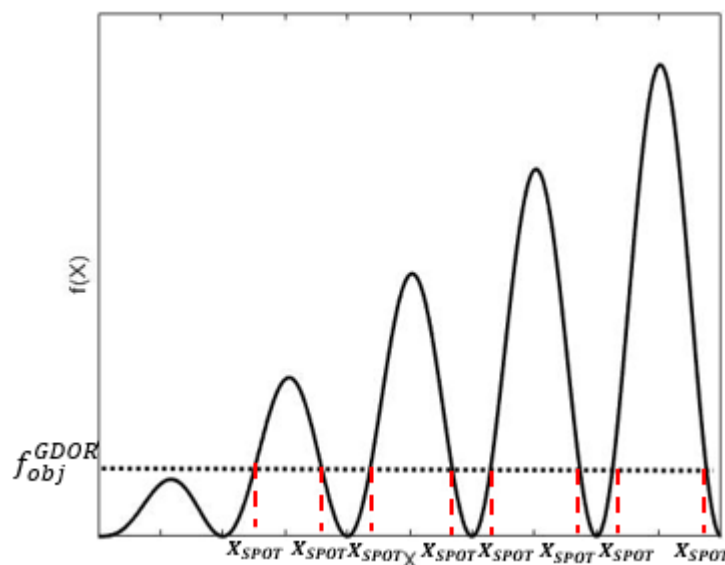
**Fig. 2.** Graph illustrating the probability density function (PDF) of a normal distribution for different values of standard deviation and nominal mean equal to  $\mu_{RRT}$ , where a  $X_{SPOT}$  value is assumed for observance in the abscissa.

If the uncertain parameter with normal PDF were transformed into a variable in the optimization matrix to determine the optimal value for  $X_{SPOT}$ , to which it minimizes RR, the major consequences are an increase in the model's complexity and a decrease of robustness; hence, the model would become frequently and probably nonlinear or more nonlinear if it already were nonlinear. In calculating  $X_{SPOT}$  analytically, if the

optimization provided an infeasible solution,  $X_{SPOT}$  would not be determined and the entire analytical methodology would go wrong, and all effort would have been in vain. Another positive point for not analytically calculating  $X_{SPOT}$  is that the parameters aimed to assemble the mathematical model (i.e., the parameter that does not appear on the left or right-hand side of the model's constraints, but influence their activation) can also be fine-tuned through this framework. Thus, the conclusion is that  $X_{SPOT}$  cannot be analytically calculated.

In this framework, OF that are monotonic along the DCFR will have more stability (statistical confidence) in the ARR value associated with each different standard deviation value, because this type of functions do not show oscillation or changes in the order relation. When the OF is not monotonic in the DCFR, for each different fine-tuning the ARR may not converge to the same value if the OF in the problem is too sensitive to variations and if the number of optimizations made for fine-tuning is too low (see section 3). Monotonic OF is a non-oscillatory function that tends to be more stable, dispensing a very large number of Monte Carlo simulations to satisfy the framework objective (section 3).

While monotonic functions have just one  $X_{SPOT}$ , non-monotonic functions can have more than one  $X_{SPOT}$  if it is out of the DCFR as shown in Fig. 3. They can also be very sensitive to variations – e.g., the bigger the sensitivity of the OF, the bigger the difficulty to fine-tune the standard deviation value of uncertainty  $X$ . This difficulty exists because there would be several values for the uncertainty  $X$  that would cause the OF of ( $X$ ) to achieve the value of the ideal solution  $f_{obj}^{GDOR}$ . In other words, there would be several values of standard deviation that would lead to an uncertainty value that would make OF achieve  $f_{obj}^{GDOR}$ , but these different standard deviations could achieve this less frequently. To circumvent this and to infer a reliable decision-making criterion for choosing the best standard deviation, it would be necessary to perform a very large number of Monte Carlo simulations. That is why this problem can be avoided and such a large number of Monte Carlo simulations is unnecessary when the function is monotonic in the DCFR.



**Fig. 3.** Example of a non-monotonic function that oscillates and makes it difficult for the framework to work if the number of performed optimizations is small during the fine-tuning.

Note that  $f_{obj}^{GDOR}$  is reachable by the function in Fig. 3, but generally this ideal target does not need to be reachable by the function since it is a goal and the objective is to draw as near to it as possible. This framework can be applied to fields such as Engineering (e.g., production and/or distribution planning and scheduling, project and production design), Economics (e.g., portfolio optimization), Operational Research, Artificial Intelligence, Process Control and Automation, and Theory and Simulation, i.e., fields with monotonic objective functions in the DCFR. In the case of Process Control and Automation, an example is an MPC project to which OF is the setpoint of a controlled variable,  $f_{obj}^{GDOR}$  can be the reference setpoint value that maximizes the process profit,  $f_{obj}^{rob(RRT)}$  is the most feasible recent setpoint to reach due to recent special causes in the plant,  $X_{SPOT}$  would be the value of the manipulated variable that would draw  $f_{obj}^{rob(RRT)}$  closer to  $f_{obj}^{GDOR}$ , and  $\mu_{LTRR}$  would be the recent average value used for the manipulated variable.

### 2.3. Mathematical formulation

This section presents the schematic of the framework's mathematical formulation. It includes transforming the original deterministic model into a robust deterministic model that is then fine-tuned by an algorithm (section 2.4) aiming at robustness and targeting ideality. It also provides the axioms underlying the functioning of the framework. Any deterministic linear or nonlinear model in the problem domain with at least one parameter that can be considered as an uncertainty following a normal distribution can be mathematically written in the way herein proposed. The problem to be optimized (Eq. 5) is either a maximization problem or a minimization problem where the objective function and the constraints are mathematical combinations of  $x, \varepsilon, \theta, \alpha, \xi$ :

$$\begin{aligned}
 & \min_x f(x, \varepsilon, \theta, \alpha, \xi) \quad (5) \\
 & \text{ou} \\
 & \max_x f(x, \varepsilon, \theta, \alpha, \xi) \\
 & s. t. \quad h(x, \varepsilon, \theta, \alpha, \xi) \leq 0 \\
 & \quad g(x, \varepsilon, \theta, \alpha, \xi) \geq 0 \\
 & \quad w(x, \varepsilon, \theta, \alpha, \xi) = 0 \\
 & \quad x_i^L \leq x_i \leq x_i^U \\
 & \quad \varepsilon_j^L \leq \varepsilon_j \leq \varepsilon_j^U \\
 & x_i \in x; x \in X; \varepsilon_j \in \varepsilon; \varepsilon_j \in A_j; \theta_k \in \theta; \theta_k \in B_k; \alpha_l \in \alpha; \alpha \in \mathbb{R}^+; X, A_j, B_k \in \mathbb{R} \\
 & x = \begin{bmatrix} x_1 \\ \vdots \\ x_{i=iN} \end{bmatrix}; \varepsilon = [\varepsilon_j]; \theta = [\theta_1 \dots \theta_{k=kN}]; \alpha = [\alpha_1 \dots \alpha_{l=lN}]; \\
 & \xi = [E\{\xi_1\} \dots E\{\xi_{m=mN}\}]
 \end{aligned}$$

$f(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})$  is the OF (monotonic in DCFR) to be optimized.  $\mathbf{x}$  is the vector of decision variables.  $\boldsymbol{\varepsilon}$  is the vector containing the uncertain parameters that follows a normal distribution and have values randomly generated according to Monte Carlo sampling before solving the optimization problem (therefore having a deterministic nature in the optimization problem).  $\boldsymbol{\theta}$  is the vector of exact and deterministically defined parameters for the optimization problem.  $\boldsymbol{\alpha}$  is the vector of penalty factors.  $\boldsymbol{\xi}$  is the vector containing the expected values  $E\{\xi_m\}$  of each uncertainty  $m$  that is represented by a non-normal probability distribution (in this framework, uncertainties of type  $\xi_m$  are not mandatory).  $\mathbf{h}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})$  is the vector of constraints that are equal to or lower than zero.  $\mathbf{g}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})$  is the vector of constraints greater than or equal to zero.  $\mathbf{w}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})$  is the vector of equations equal to zero, where the optimization problem must have at least one of these three vectors.  $x_i^L$  is the minimum limit value for the decision variable  $i$ .  $x_i^U$  is the maximum limit value for the decision variable  $i$ .  $\varepsilon_j^L$  is the minimum limit value for the uncertain parameter  $\varepsilon_j$ .  $\varepsilon_j^U$  is the maximum limit value for the uncertain parameter  $\varepsilon_j$ .  $X$  is the set of possible values for  $x_i$  in the real domain  $\mathbb{R}$ .  $A_j$  is the set of possible values for  $\varepsilon_j$  in the real domain  $\mathbb{R}$ .  $B_k$  is the set of possible values for  $\theta_k$  in the real domain  $\mathbb{R}$ .  $iN$ ,  $kN$ ,  $lN$  and  $mN$ , respectively, are the number of elements in the vectors  $\mathbf{x}$ ,  $\boldsymbol{\theta}$ ,  $\boldsymbol{\alpha}$ , and  $\boldsymbol{\xi}$ . Note: there can be no division by zero. The probability distributions in this framework are disjoint, which makes estimation easier regardless of the problem dimensionality.

The framework can be used for problems with linear programming (LP), nonlinear programming (NLP), mixed integer linear programming (MILP), mixed integer nonlinear programming (MINLP), or semidefinite programming (SDP) formulation. The axioms are as follows:

- Axiom *i*): The OF must have increasing only or decreasing only monotonic behavior along the DCFR; therefore, the OF partial derivatives in relation to the normal uncertainty(ies) and to  $x_i$  must have the same mathematical signs in the DCFR. When the OF is monotonic in the DCFR, the order relation is preserved and the problem becomes either convex or strictly convex, or concave or strictly concave. Thus, for different amounts of performed optimizations ( $N^{BIG}$ ), the solution concentrations in the average relative regret (ARR) graph vs each  $N^{BIG}$  realization remains with preserved order ranges (Figs. 5 to 27). This favors the stimulus of a standard deviation value over the OF to have and maintain characteristics inherent to each different value.
- Axiom *ii*): There must be at least one normal uncertainty ( $\varepsilon_j$ ) in the problem, and the framework can only be used if the stimulus action of each normal uncertainty over the OF value (i.e., the influence caused by a normal uncertainty in the OF value) is independent of the stimulus of any other normal uncertainty.

If the term(s) linked to a normal uncertainty is linearly independent (LI) of the other terms linked to other normal uncertainties in the model, then the framework can be used, because the stimulus in the OF will be exclusive (uninfluenced by other terms of other normal uncertainties) and independent (independent of the

existence of other terms of other normal uncertainties) of the other stimuli. So it is only necessary to check whether a normal uncertainty satisfy this axiom *ii*) if it is an uncertainty that is multiplying/dividing some decision variable of the optimization problem, or forming a power term with the variable, i.e. a multiplicative uncertainty. Any other case will satisfy this axiom without the need to run this check, e.g. additive uncertainties in the model (OF and / or constraints). In addition, if the problem has just one normal uncertainty, there is no need to check axiom *ii*).

Uncertainties can appear anywhere in the model, i.e., in OF and/or constraints, but there can be no multiplication/division/potentialiation between different normal uncertainties (e.g.,  $\varepsilon_1 \varepsilon_2$ ). Non-multiplication between normal uncertainties prevents the possibility of changing the behavior of the OF from increasing to decreasing or vice versa (thus satisfying axiom *i*) and *ii*)). If the  $\varepsilon_j$  stimuli are independent, the OF has a unique behavior for each  $\varepsilon_j$  stimulus, and there may be superposition of independent stimuli to the behavior of the OF in relation to the normal uncertainties  $\varepsilon_j \forall j$ . Eq. (6) shows an example of testing a deterministic model to find out whether the terms of the multiplicative uncertainties are exclusive and independent, which  $\varepsilon_1$  and  $\varepsilon_2$  are because  $\xi_1 x_4$ ,  $\varepsilon_2 x_3$ ,  $x_3$  and  $x_4$  are unlinked to  $\varepsilon_1 x_1$ ,  $x_1$  and  $x_2$ . As such, the framework can be used to this model. As the term  $\varepsilon_3$  is an additive uncertainty in the model, its stimulus over the OF is exclusive and independent, i.e.,  $\varepsilon_3$  is impartial in relation to  $\varepsilon_1 x_1$  and  $\varepsilon_2 x_3$ , as well as its stimulus over the OF is independent of the influence of  $\varepsilon_1 x_1$  and  $\varepsilon_2 x_3$ , which are the terms linked to the other normal uncertainties. If an uncertainty is independent, the values of the terms linked to it do not depend on the values of other terms unlinked to it.

$$\begin{aligned} \min_x f &= x_1 x_2 + \varepsilon_1 x_1 + \varepsilon_2 x_3 + \varepsilon_3 + \xi_1 x_4 \quad (6) \\ \text{s.t. } x_1 x_2 &\leq A \\ x_3 + x_4 &\leq B \end{aligned}$$

- Axiom *iii*): Uncertainties that fit well to some non-normal probability distribution ( $\xi_m$ ) must be represented by their expected values in the robust model ( $E\{\xi_m\}$ ) and have an  $\alpha_l$  linked (multiplying) to each expected value ( $\alpha_l E\{\xi_m\}$ ) of this uncertainty in the robust model. There will be an  $\alpha_l$  for each non-normal uncertainty ( $\xi_m$ ), which is important to regulate the conservatism and robustness of the model. If having two or more expected values that represent different uncertainties (e.g.,  $E\{\xi_1\}E\{\xi_2\}$ ) which are multiplied in an inequality or equation term, there will be a new and exclusive  $\alpha_l$  linked to this combined term (e.g.  $\alpha_3 E\{\xi_1\}E\{\xi_2\}$ ). It is possible to have a multiplication between a normal uncertainty and some expected value of another uncertainty or with some term containing expected values of other uncertainties (e.g.,  $\varepsilon_1 \alpha_1 E\{\xi_1\}$  or  $\varepsilon_1 \alpha_3 E\{\xi_1\}E\{\xi_2\}$ ).
- Axiom *iv*): It is optional to introduce  $\alpha_l$  into the model that refers to general deterministic optimization of reference (GDOR), since its ideal solution must be feasible. In addition, there is

no insertion of  $E\{\xi_m\}$  or randomness in GDOR; hence, if there is  $\alpha_l$  in it, there would be a multiplication  $\alpha_l \xi_m$  instead of  $\alpha_l E\{\xi_m\}$ .

- Axiom v): A non-normal uncertainty ( $\xi_m$ ) is not mandatory in the mathematical model used for the optimization problem of the present framework; however, at least one normal uncertainty ( $\varepsilon_1$ ) is necessary to follow the steps of the framework's algorithm.
- Axiom vi): It is reasonable to place values for  $\alpha_l$  that are not very different from the numerical value 1, so that the OF is not penalized too much (in the sense of having its value modified upwards or downwards). It is also reasonable to admit a value for  $\alpha_l$  whose multiplication with the expected value of an uncertainty ( $\alpha_l E\{\xi_m\}$ ) results in a numerical value within the lower and upper limits of values available for this uncertainty in historical data, thus giving reasonableness in statistical inference of uncertainties.
- Axiom vii): If the uncertainty is epistemic or has historical data that did not fit well with some probability distribution, it is considered to have a normal distribution, that is, the uncertainty will be of type  $\varepsilon_j$ . The greater the amount of historical data of an uncertainty, the more this is true, due to the central limit theorem.

The constraints of the present framework are not reformulated because as both the expected value and  $\alpha_l$  are numbers, the only mathematical role that  $\alpha_l$  plays is to modify the absolute value of the expected value (or the multiplication between expected values, if any) in order to protect the model against the violation of constraints. In other words, multiplying  $\alpha_l$  with the expected value will only generate a new number that will represent the variability of a given uncertainty (as well as the role played by the expected value), leaving the physical meaning of the terms of the constraints unchanged.

The present framework's mathematical formulation does not explicitly include the equation of the probability density function of a normal distribution to calculate the expected value of an uncertainty. If it were included, the expected value would be its own average, i.e., the mean of the uncertainty values, thus not computing the possible variability of each uncertainty with normal distribution in the process because of the symmetry of the normal distribution. In addition, the analytical inclusion of standard deviations would probably make the model nonlinear if it were a decision variable in the model instead of being specified, making it harder to solve and unnecessarily increasing the optimization time. If the model were linear, considering the standard deviation as a decision variable would only make it unnecessarily nonlinear, increasing the optimization processing time and making it difficult to converge to a feasible solution, especially in mathematical models that use integrals (e.g., chance-constrained optimization).

Including a probability density function of a normal distribution in the model and not following the steps of the framework's algorithm would prevent the chance of an uncertainty with normal distribution to have its value generated close to  $X_{SPOT}$  (hypothetically shown in Fig. 2). If the expected value of the probability density function for an uncertainty with normal distribution were considered, the "uncertain" value of the uncertainty would be the average value  $\mu_{LRR}$  in the curve, and hence the problem would not be flexible

enough to allow the choice of the behavior of the uncertainty to draw  $f_{obj}^{rob(RRT)}$  as close as possible to  $f_{obj}^{GDOR}$ . For this reason, a normal distribution is considered for the random generation of the uncertainty value by Monte Carlo sampling before optimization is performed, instead of analytically including the probability density function in the model. In addition, this methodology allows the model to have a deterministic nature without having to solve integrals. Since it is not an analytical methodology, this Monte Carlo strategy allows the OF to be discontinuous in relation to  $\varepsilon_j$  and  $\varepsilon_j$  to be linear or nonlinear in nature.

The framework's mathematical properties for protection against violation of constraints are:

- i) When any  $\alpha_l$  appears in constraints of type  $\mathbf{h}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi}) \leq 0$ , and the term linked to  $\alpha_l$  on the left-hand side (LHS) of the present constraint is mathematically positive, decreasing the value of  $\alpha_l$  will increase the model's robustness.
- ii) When any  $\alpha_l$  appears in constraints of type  $\mathbf{h}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi}) \leq 0$ , and the term linked to  $\alpha_l$  on the LHS of the present constraint is mathematically negative, increasing the value of  $\alpha_l$  will increase the model's robustness.
- iii) When any  $\alpha_l$  appears in constraints of type  $\mathbf{g}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi}) \geq 0$ , and the term involving the  $\alpha_l$  on the LHS of the present constraint is mathematically positive, increasing the value of  $\alpha_l$  will increase the model's robustness.
- iv) When any  $\alpha_l$  appears in constraints of type  $\mathbf{g}(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi}) \geq 0$ , and the term surrounding the  $\alpha_l$  on the LHS of the present constraint is mathematically negative, decreasing the value of  $\alpha_l$  will increase the model's robustness.

The values for the penalty factors  $\forall \alpha_l \in \boldsymbol{\alpha}$  are specified to change inside a loop in the algorithm of the framework or the decision maker can specify it by trial and error according to the properties above. In both cases, the value of  $\alpha_l$  must follow axiom vi).

## 2.4. Algorithm

Fine-tuning is carried out so that the final robust solution approaches as close as possible an ideal solution  $f_{obj}^{GDOR}$ . Once the original deterministic model is transcribed to the framework's mathematical formulation, the step-by-step procedure of the framework's algorithm will fine-tune the robust model to determine the best value for the standard deviation of a normal uncertainty. The model's conservatism and robustness are regulated by choosing or determining (through a loop) the values of the penalty factors  $\alpha_l$ . In this framework, it is not necessary to control the risk of undesirable values being generated for uncertainties, as the framework fine-tunes the model so that decision making does not depend on the risk aversion of unusual



values generated for normal uncertainties to be undesirable, because uncertainties with higher values far from the average can generate better results.

Unlike the variability models, this framework does not control the variability of the solution, but rather finds an operating bell curve that is the best one to achieve the framework's objective. At the end of the robust model fine-tuning following the algorithm steps, the objective is not to find a solution with better or worse objective function (OF), but rather to fine-tune the model to find a robust result ( $f_{obj}^{rob(RRT)}$ ) that is as close as possible to the ideal solution itself ( $f_{obj}^{GDOR}$ ).

The algorithm is presented below focusing on how to fine-tune the value of the standard deviation of a normal uncertainty ( $\varepsilon_1$ ), in line with the regulation of the model's conservatism and robustness. In case the optimization problem has more than one normal uncertainty, according to axiom *ii*) the algorithm's functioning would be the same, but first it would be fine-tuned as  $\varepsilon_1$  and then as  $\varepsilon_2$  and so on. This shows that when fine-tuning a normal uncertainty ( $\varepsilon_j$ ), the others remain with their fixed and constant values, each equivalent to their mean in the recent reference timeline (RRT) data. The algorithm consists of the following steps:

- i) **Step *i*):** Perform a general deterministic optimization of reference (GDOR) that has a feasible solution and store  $f_{obj}^{GDOR}$  to calculate the relative regret (RR) (%) according to Eq. (4), or simply specify  $f_{obj}^{GDOR}$ .
- ii) **Step *ii*):** Apply the mathematical formulation to the deterministic model in use according to the axioms. Specify values for each existing  $\alpha_l$ , if there are non-normal uncertainties ( $\xi_m$ ). If the decision maker changes their mind and wants to modify some value of  $\alpha_l$  at any time (or even if  $\alpha_l$  is in a loop), it is necessary to perform a new fine-tuning of the optimization problem from Step *iii*).
- iii) **Step *iii*):** Define fixed nominal values for each uncertain parameter ( $\xi_m$ ) and other parameters that are not uncertainties, as being their RRT nominal values, fixing their values, even for all normal uncertainties except one of them. The normal uncertainty value being fine-tuned ( $\varepsilon_1$ ), in which the value has not been fixed, will be randomly generated by sampling via Monte Carlo simulation (in the optimizations of Step *iv*), which will enter the model as input data, and that the nominal average of  $\varepsilon_1$ , is the RRT  $\mu_{RRT}$  and the standard deviation is a percentage of  $\mu_{RRT}$ , as shown in Fig. 2. The fixed nominal values of deterministic parameters and other normal uncertainties are their averages from the RRT historical data. Uncertainties of  $\xi_m$  type will have as fixed values their expected values ( $E\{\xi_m\}$ ) of their historical data in the RRT.
- iv) **Step *iv*):** Assume different values for the standard deviation ( $\sigma$ ) of uncertainty  $\varepsilon_1$  and perform a large subjective quantity  $N^{BIG}$  (depending on the case, 50 is enough, as shown in Fig. 24) of optimizations for each of the standard deviation values assumed. At the end of each performed optimization, the RR (%) is calculated according to Eq. (4) and stored. If the set of performed

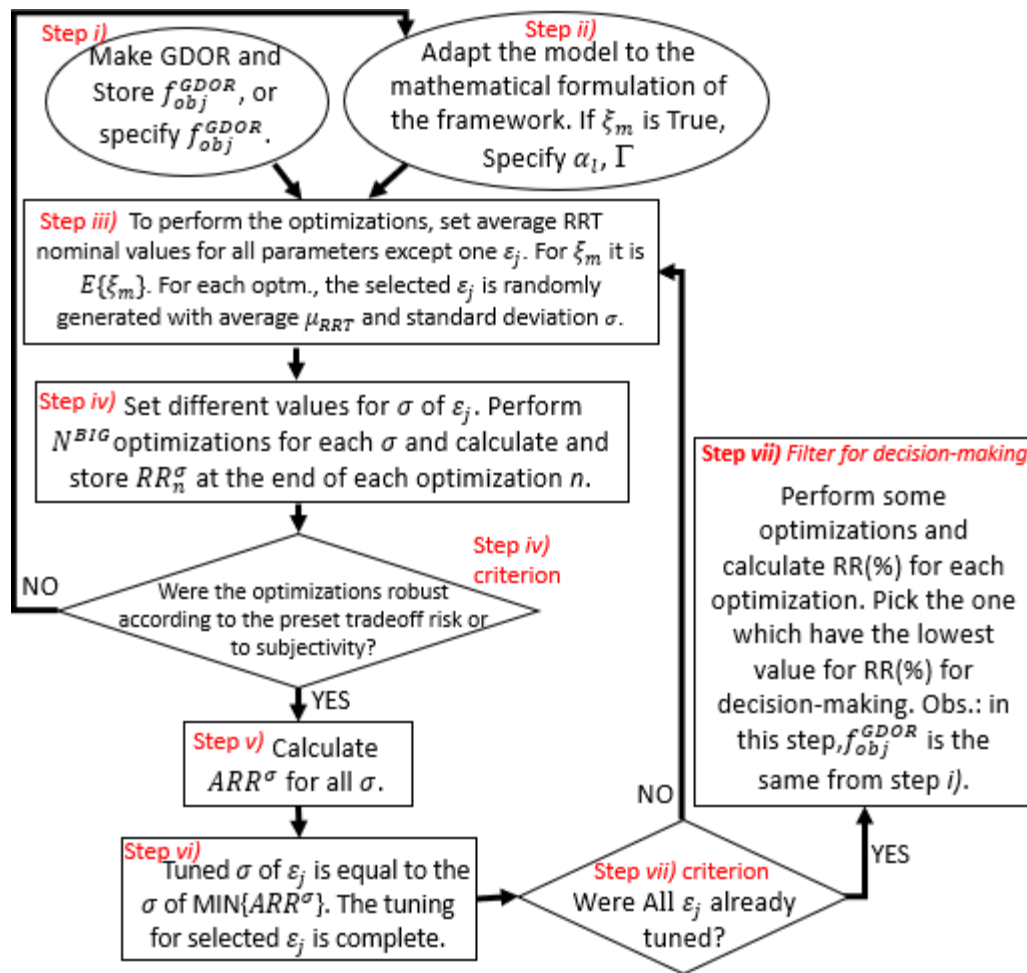
optimizations were not satisfactorily robust according to the subjectivity of the decision maker (offline analysis of the algorithm), i.e., according to what the decision maker judges/accepts to be robust, or according to a preset tradeoff risk  $\Gamma$  (conditional statement within the algorithm), return to Step *ii*) and specify a new  $\alpha_l$ .  $\Gamma$  and  $\alpha_l$  are specified according to the decision maker's will, but  $\alpha_l$  can be looped if not specified (as stated in Step *ii*).  $\Gamma$  is the percentage ratio between the number of optimizations that provided feasible solutions and the number of performed optimizations.

- v) **Step  $\nu$ ):** Calculate, according to Eq. (7), the average relative regret (ARR) (%) for each of the assumed standard deviation values ( $ARR^\sigma$ ) for the uncertainty  $\varepsilon_1$ , and store the calculated values for future comparison in Step *vi*).  $N^{BIG}$  is the quantity of performed optimizations for each standard deviation value  $\sigma$  assumed for the uncertainty  $\varepsilon_1$ , and  $RR_n^\sigma$  is the value of the RR (%) of the optimization of the  $n$  realization for an assumed value for the standard deviation  $\sigma$ , to which  $RR_n^\sigma$  is calculated in Step *iv*) according to Eq. (4).

$$ARR^\sigma (\%) = \sum_{n=1}^N RR_n^\sigma \cdot \frac{1}{N^{BIG}} \quad (7)$$

- vi) **Step  $\nu i$ ):** The standard deviation  $\sigma$ , which is linked to the lowest value of  $ARR^\sigma$ , will correspond to the standard deviation value  $\sigma$  fine-tuned for the uncertainty  $\varepsilon_1$ , which will lead to a robust average solution that most closely reach  $f_{obj}^{ODGR}$ .
- vii) **Step  $\nu ii$ ):** Fine-tuning for  $\varepsilon_1$  is complete. Choose another normal uncertainty  $\varepsilon_j$  and perform the procedure again from step *iii*), under the same fine-tuning parameter conditions as the past normal uncertainties, until all  $\sigma_j$  of  $\varepsilon_j \forall j$  are fine-tuned. When all  $\varepsilon_j \forall j$  are fine-tuned, just perform the RO a few times as the decision maker wishes, randomly generating values for all  $\varepsilon_j \forall j$  with their fine-tuned  $\sigma_j$  before a realization is done and choose the performed optimization that grants the lowest RR value (%) according to the Eq. (4). This chosen RO will describe the best decision making to approach the ideal solution  $f_{obj}^{GDOR}$ .

The fine-tuning adjusts the individual influence of each normal uncertainty in the model to determine the standard deviation value of each normal uncertainty which this value has to imply in an average of solutions that comes closest to the ideal solution value. These steps are summarized in Fig. 4:



**Fig. 4.** Algorithm of the framework.

In fine-tuning, the risk of an optimization being infeasible ( $1 - \Gamma$ ) is already a conditional parameter specified by the decision maker instead of being calculated. In this new framework, it is unusual to do the cost-risk analysis because the risk is already imbued as a target condition in the algorithm (the fine-tuning is performed, and the chosen condition is satisfied).

Since step v) calculates an average, the higher the  $N^{BIG}$ , the more precise the fine-tuning is. Stochasticity in step vii) avoids the need for an exhaustive brute-force search to choose the best result. The final decision making stems from the best case of step vii) (the one with the lowest  $RR(\%)$ ).

Important comprehensions of the framework are:

- i) The algorithm fine-tunes the standard deviations and the penalty factors to lead to feasible solutions matching the specified  $\Gamma$ , which is a highly tractable form to solve the optimization problem, as it adapts the model.
- ii) The preset tradeoff risk  $\Gamma$  is appropriate when the decision maker wants to align with standardized metrics of optimization quality according to a business plan.

- iii) If  $\alpha_l$  is equal to 1, the model may tend to get closer to reality, since the estimated expected value  $E\{\xi_m\}$  will not change; therefore, it is recommended that the default value for  $\alpha_l$  of non-normal uncertainties be equal to 1 first, i.e., this value must be included in the loop.
- iv) The framework is consistent with the central limit theorem, given that epistemic uncertainties are well suited to the normal distribution. For example, demand is a great candidate to be an epistemic uncertainty when: i) production or consumption levels do not follow a constant pattern, or ii) there is no production or consumption trend to be followed, and even if there is historical data available for demand, turbulence in the data values caused by these two items will make it difficult to reliably adjust the data to a probability distribution.
- v) Standard deviations with close values can compete as the best fine-tuning value (e.g. Fig. 14), as the solutions will come out similar in the calculation of the ARR (considering that the behaviors of the normal curves will be similar). In linear problems, this competition is not significant, but for nonlinear problems, increasing  $N^{BIG}$  provides more statistical confidence to the fine-tuning. In all examples in section 3, only 4 different values were assumed for the standard deviation, to show the behavior of the trends more clearly from low to high range values for the standard deviation. Fig. 14 shows that the objective function is smooth at intervals of standard deviation between 2 and 5% for the normal uncertainty of the problem. It is not recommended that the standard deviation value exceeds 20% of the average to avoid the risk of randomly generating values that have no physical meaning (possibly negative values).
- vi) The value of the RR (%) also varies from model to model, with cases in which the RR has large, medium or low sensitivity to the different values of standard deviations for an uncertainty. This is natural because as the mathematical models differ, so do the results.
- vii) The  $\alpha_l$  values also influence the ARR value and therefore the fine-tuning of the standard deviation.
- viii) If the variation between values of  $ARR^\sigma$  is small, it means that the OF must have relevant levels (smooth in relation to the variables) for some ranges of the variables. However, if the variation between values of  $ARR^\sigma$  is high, it means that there is exponential or high order relation behavior for the OF (non-smooth) in other intervals of the feasible region (of the optimization problem).
- ix) If  $f_{obj}^{GDOR}$  is close to a smooth region of the OF, then there will be little variation in the values of  $ARR^\sigma$ , but if  $f_{obj}^{GDOR}$  is close to a non-smooth region of the OF, there will be considerable variations between the values of  $ARR^\sigma$ .

### 3. Application and computational results

This section aims to provide a better understanding of the framework by applying it to simple examples that abide by the axioms. First it is applied to a linear model and then it to two different nonlinear models derived from the linear one. In all cases,  $\Gamma$  was specified as 100%. The sample problem refers to a mechanical part seller who leaves the house with two different types of parts to sell from door to door. He works for a

company and earns as much as he sells as through sales commission. The example is the application of the framework to fine-tune the normal uncertainty  $\varepsilon_1$  of a planning problem to maximize the daily profit  $f(x, \varepsilon, \theta, \alpha, \xi)$ , knowing that there are limits to the gain from selling type 1 and type 2 parts due to production limitations on the part of the company and that the sale prices of the different types are uncertainties. The selling price for type 1 follows a normal distribution, while the selling price for type 2 follows an exponential distribution. The mathematical formulation of the example is described in Eq. (8):

$$\begin{aligned} \max_x f(x, \varepsilon, \theta, \alpha, \xi) &= x_1 \varepsilon_1 + x_2 \alpha_2 E\{\xi_2\} + x_1 \theta_1 + x_2 \theta_2 - \theta_3 & (8) \\ \text{s. t. } x_1 + x_2 - UB &\leq 0 \\ x_1 + x_2 - LB &\geq 0 \\ x_2 \alpha_2 E\{\xi_2\} - \theta_4 &\leq 0 \\ x_1 \varepsilon_1 - \theta_5 &\leq 0 \\ x_1 - 20 \geq 0; x_2 - 20 &\geq 0; x_1, x_2 \in \mathbb{I} \\ 0,5 \leq \varepsilon_1 &\leq 3,5 \\ x &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \varepsilon = [\varepsilon_1]; \theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5]; \alpha = [\alpha_2]; \xi = [E\{\xi_2\}] \end{aligned}$$

Where:  $x_1$  is the quantity of type-1 parts sold,  $\varepsilon_1$  is the selling price per type-1 part,  $x_2$  is the quantity of type-2 parts sold,  $\alpha_2$  is the penalty factor of the term that appears  $E\{\xi_2\}$ ,  $E\{\xi_2\}$  is the expected value of the sale price for type-2 part,  $\theta_1$  is the commission gain for selling type-1 part,  $\theta_2$  is the commission gain for selling type-2 part,  $\theta_3$  is the cost due to the use of a transportation vehicle, LB is the lower limit value for the sum  $(x_1 + x_2)$ , UB is the upper limit value for the sum  $(x_1 + x_2)$ ,  $\theta_4$  is the gain limit value per sale of type-2 parts,  $\theta_5$  is the gain limit value per sale of type-1 parts. This example do not use a loop to fine-tune the penalty factor  $\alpha_2$ .

Following the algorithm steps: it is first necessary to perform a GDOR, so that a comparison can be made as a criterion for fine-tuning the problem (since  $f_{obj}^{GDOR}$  is not being specified in the example, but is calculated); and to perform the GDOR, uncertain parameters become deterministic. In the example, 8 different GDORs were performed for two different cases (4 instances for each case) to run the algorithm for different situations and thus show the behavior of the RR and consequently the ARR and the selection of the best standard deviation value for each situation. For both the GDOR and the RRT, the first case takes into account  $\alpha_2 = 1$  and  $\theta_4 = 100$ , while the second case takes into account  $\alpha_2 = 0.85$  and  $\theta_4 = 70$ . Both cases are feasible, but when the value of  $\theta_4$  is changed from 100 to 70, the model becomes infeasible for  $\alpha_2 = 1$ ; therefore, it was necessary to change the value of the penalty factor according to the properties. The value adopted was  $\alpha_2 = 0.85$  to make the model more robust according to the mathematical property *i*). The values of the parameters used in all the GDORs are shown in Table 1.

**Table 1**

The values of the parameters used in all GDORs in the linear example.

Deterministic parameters of the GDOR	Values for GDOR considering the pair of cases (1 <sup>st</sup> case: [ $\alpha_2 = 1$ and $\theta_4 = 100$ ] and 2 <sup>nd</sup> case: [ $\alpha_2 = 0.85$ and $\theta_4 = 70$ ]) <sup>‡</sup>			
	First pair of instances	Second pair of instances	Third pair of instances	Fourth pair of instances
$\varepsilon_1$	1.9 \$/type 1	1.95 \$/ type 1	2.1 \$/ type 1	2.5 \$/ type 1
$\xi_2$	1.9 \$/ type 2	1.95 \$/ type 2	2.1 \$/ type 2	2.5 \$/ type 2
$\theta_1$	3 \$/ type 1	3 \$/ type 1	3 \$/ type 1	3 \$/ type 1
$\theta_2$	2.5 \$/ type 2	2.5 \$/ type 2	2.5 \$/ type 2	2.5 \$/ type 2
$\theta_3$	100 \$	100 \$	100 \$	100 \$
$\theta_5$	100 \$/ type 1	100 \$/ type 1	100 \$/ type 1	100 \$/ type 1
$LB$	70 parts	70 parts	70 parts	70 parts
$UB$	100 parts	100 parts	100 parts	100 parts

<sup>‡</sup>: Pair of cases: GDOR for both the 1<sup>st</sup> case ( $\alpha_2 = 1$  and  $\theta_4 = 100$ ) and the 2<sup>nd</sup> case ( $\alpha_2 = 0.85$  and  $\theta_4 = 70$ ).

For the linear problem, 8 fine-tuning problems are solved, 4 for each case. In all fine-tuning occurrences, the ARR value means an average deviation of the robust solution from the ideal solution. The nominal values of the parameters in the hypothetical RRT timeline of the example are as in Table 2. Please notice that  $\varepsilon_1$  is randomly generated according to a normal distribution.

**Table 2**

Nominal values of parameters in the hypothetical RRT timeline for all instances of the exemplary linear problem solved.

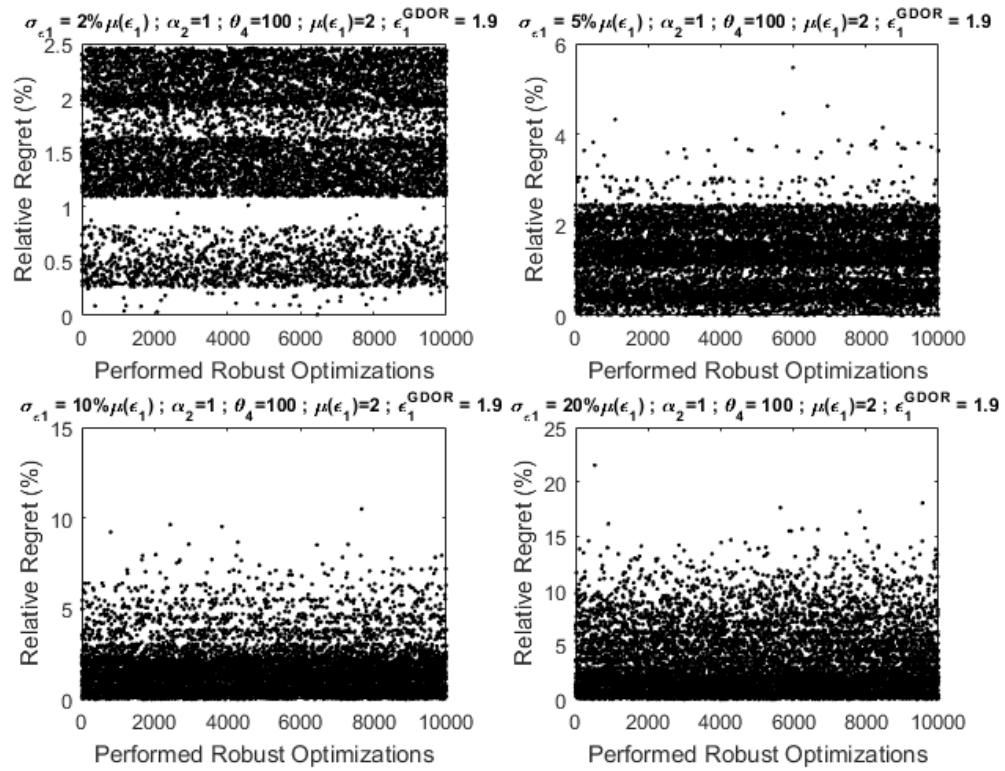
Parameters	Nominal value
$\varepsilon_1$	$(\mu = 2 \$; \pm\sigma = Z\%\mu)^{\ddagger}$
$E\{\xi_2\}$	2 \$
$\theta_1$	3 \$/type 1
$\theta_2$	2.5 \$/type 2
$\theta_3$	100 \$
$\theta_4$	The same value as $\theta_4$ of GDOR for the respective instance
$\theta_5$	100 \$/type 1
$LB$	70 parts
$UB$	100 parts

<sup>‡</sup>: In the example, it was assumed that Z can have values equal to 2, 5, 10 or 20%  $\mu$ .

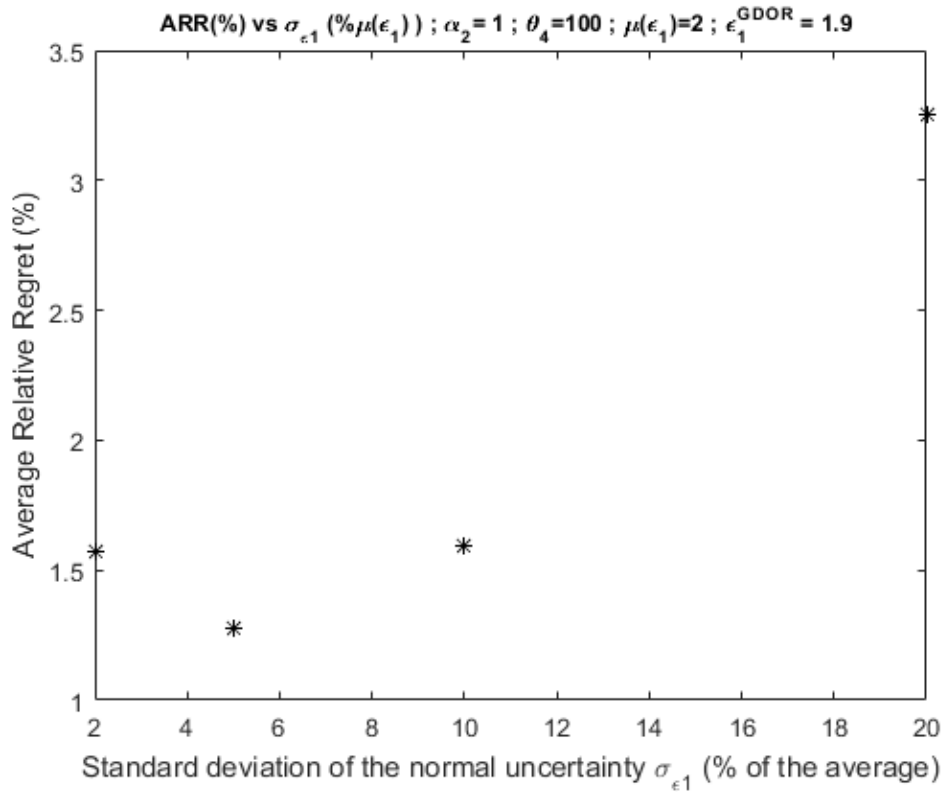
All examples were solved in MATLAB® software (R2019b, Mathworks, Natick, MA, USA). All the graphs for the first case (RRT timeline with  $\alpha_2 = 1$ ,  $\theta_4 = 100$ ,  $\mu(\varepsilon_1) = 2$ ) are shown in Figs. 5 to 12. The graphs of the second case (RRT with  $\alpha_2 = 0.85$ ,  $\theta_4 = 70$ ,  $\mu(\varepsilon_1) = 2$ ) are provided in Figs. 13 to 20. Some figures refer to the value of the relative regret (%) for each performed RO, while others refer to the value of the ARR (%) linked to the different standard deviation values used in the fine-tuning.

The statistical trend of Monte Carlo Sampling in the RR graphs is important to see the statistical and probabilistic fundamentals of the framework in action. In the practical application of the methodology, making graphs for RR or ARR is not necessary. In this paper, their intent is to show the stochastic behavior trend in the fine-tuning, matching the fundamentals provided in section 2.2.



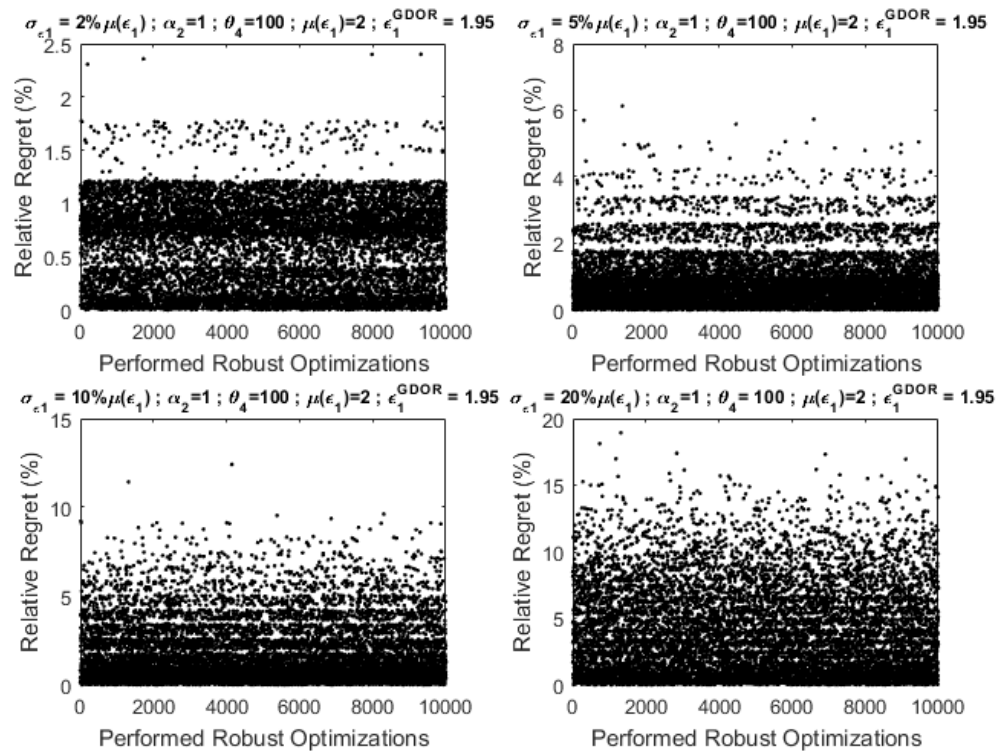


**Fig. 5.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20\%$  of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 1.9$ .

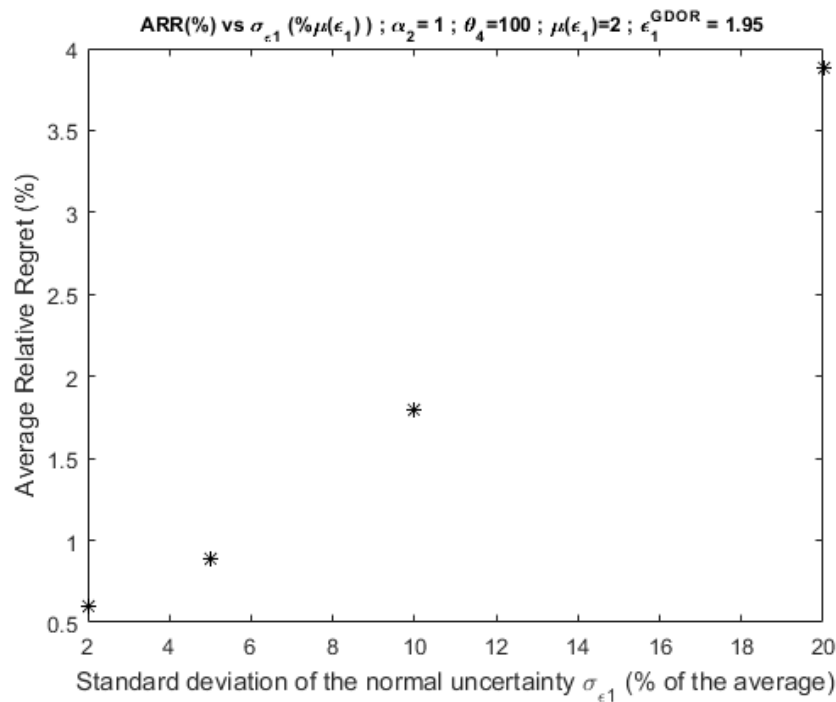


**Fig. 6.** ARR (%) vs  $\sigma_{\epsilon_1} (\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 1.9$ .

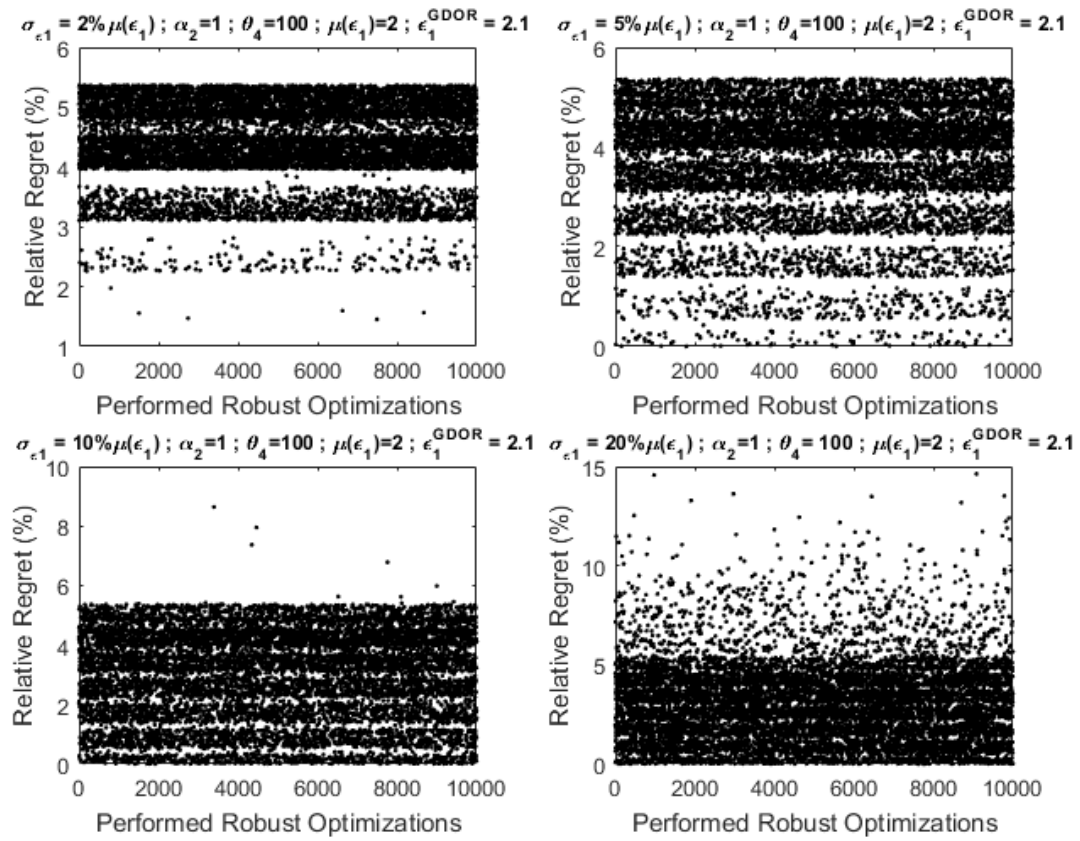




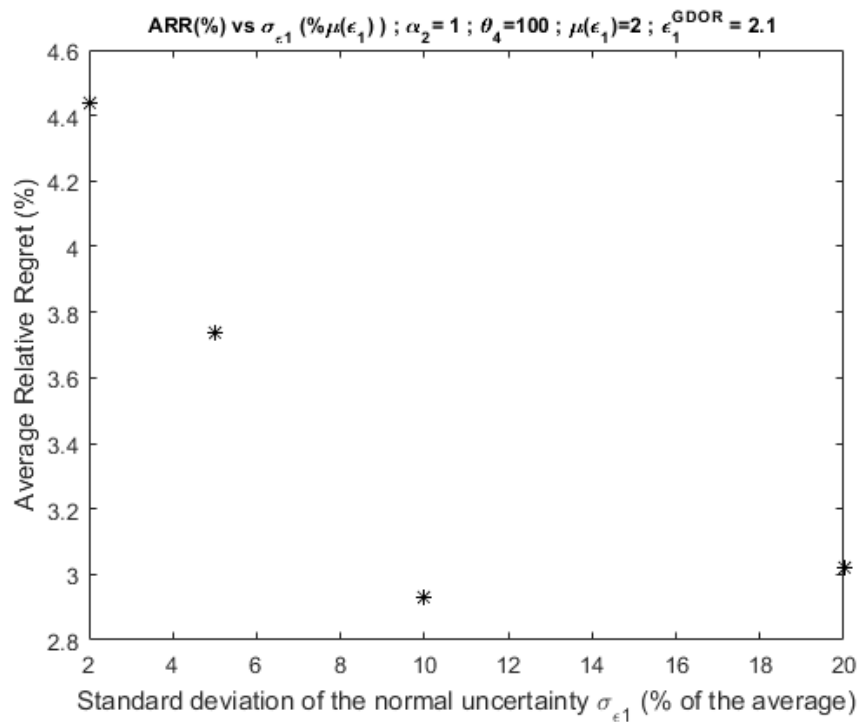
**Fig. 7.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20\%$  of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 1.95$ .



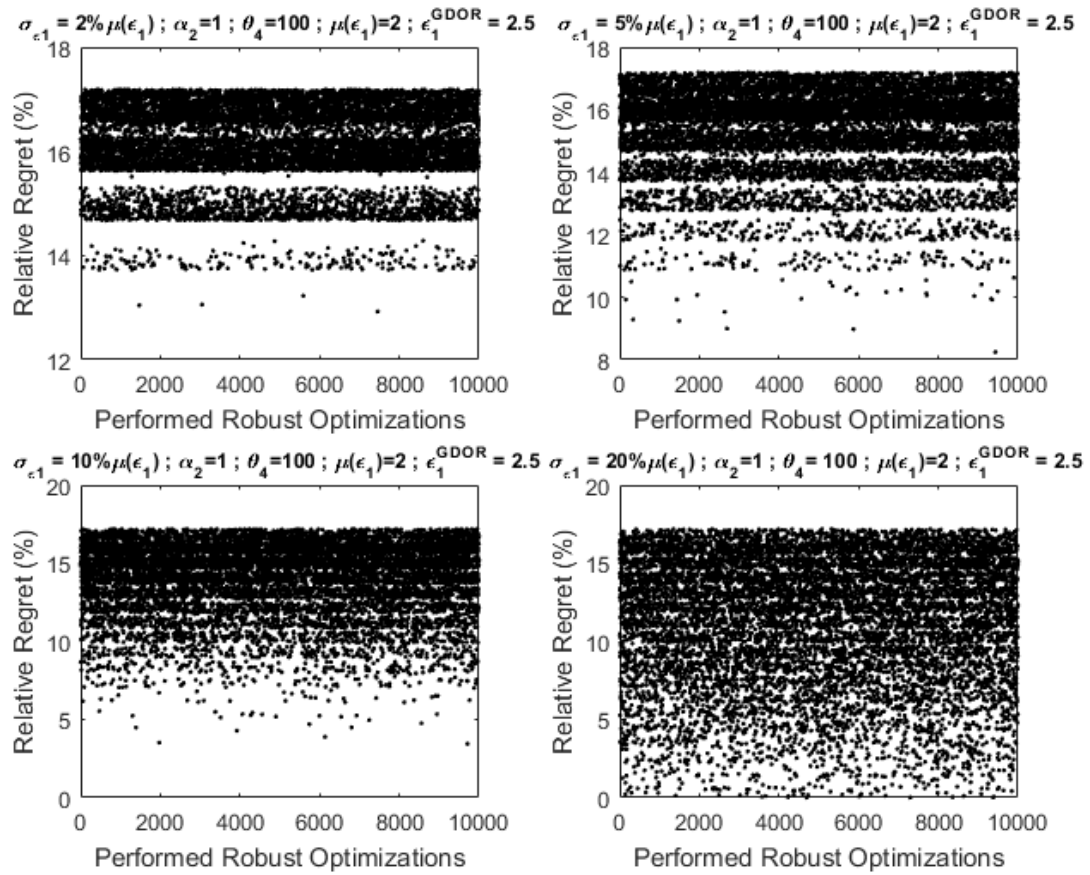
**Fig. 8.** ARR (%) vs  $\sigma_{\epsilon_1} (\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 1.95$ .



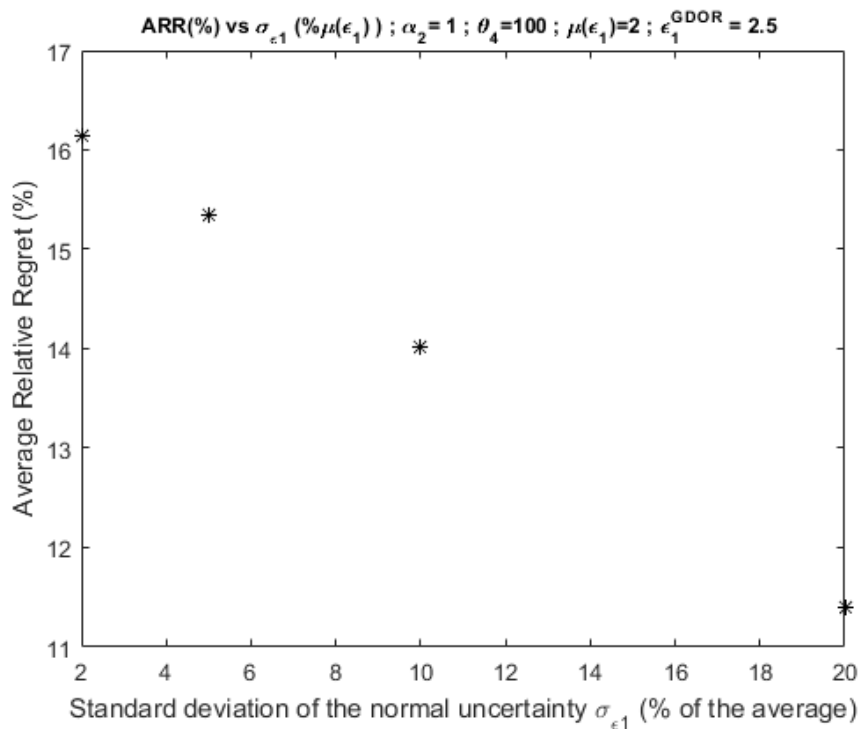
**Fig. 9.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\varepsilon_1} = 2, 5, 10, 20$  % of  $\mu(\varepsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 2.1$ .



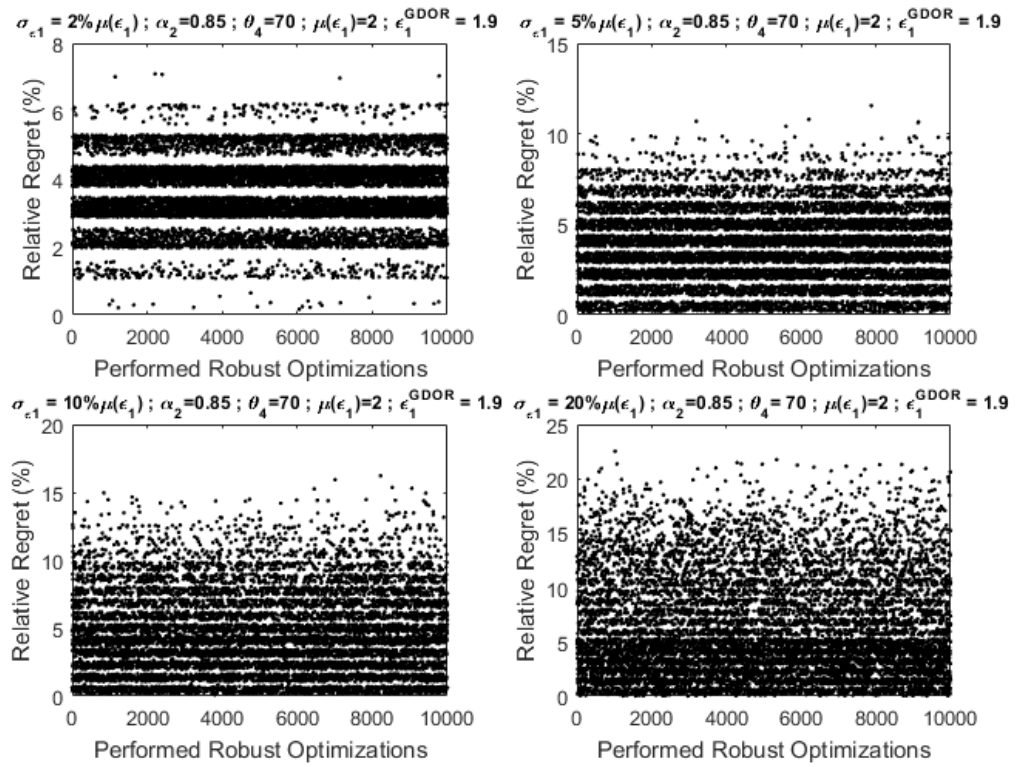
**Fig. 10.** ARR (%) vs  $\sigma_{\varepsilon_1} (\% \mu(\varepsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 2.1$ .



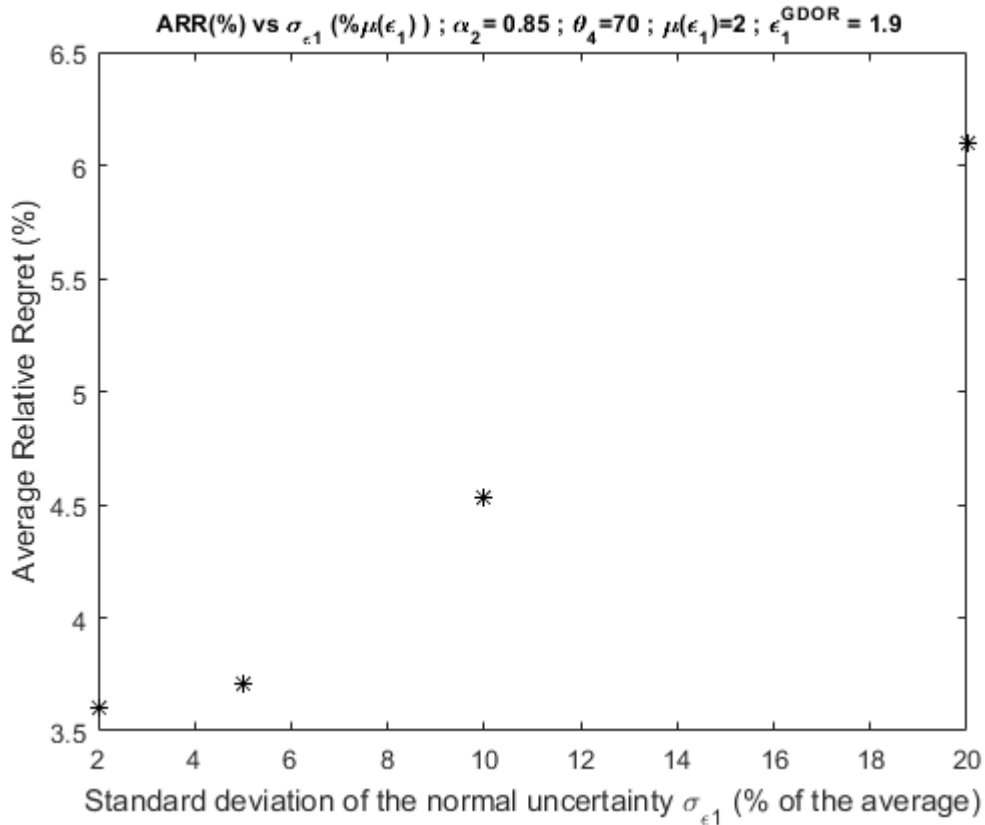
**Fig. 11.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20$  % of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 2.5$ .



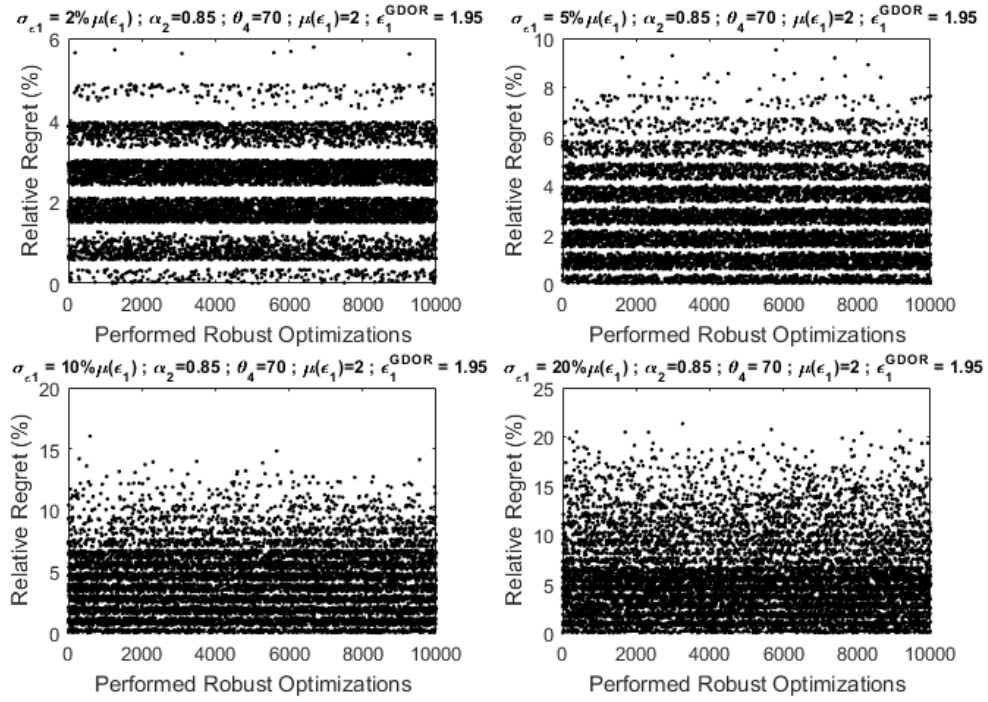
**Fig. 12.** ARR (%) vs  $\sigma_{\epsilon_1} (\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 2.5$ .



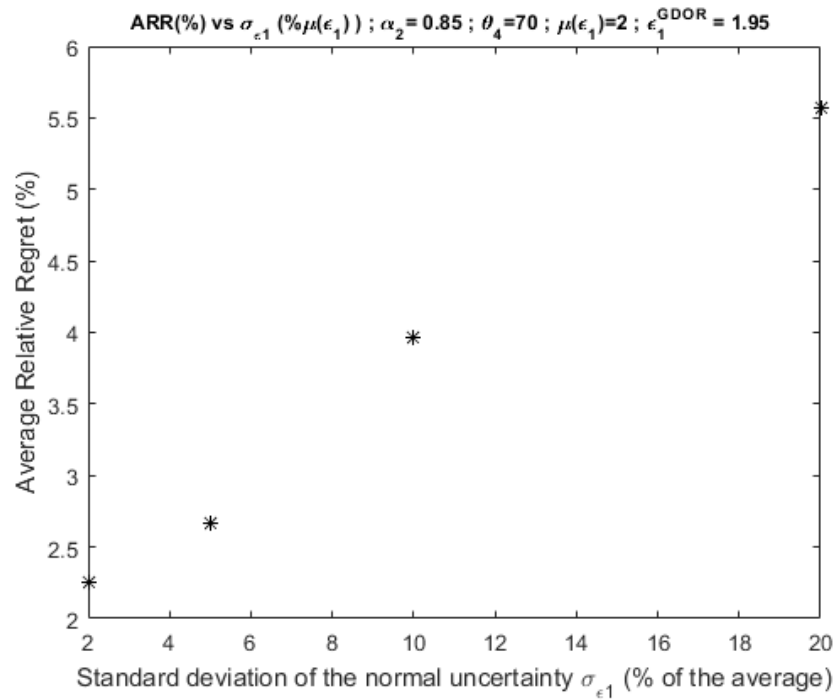
**Fig. 13.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20$  % of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 1.9$ .



**Fig. 14.** ARR (%) vs  $\sigma_{\epsilon_1} (\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 1.9$ .

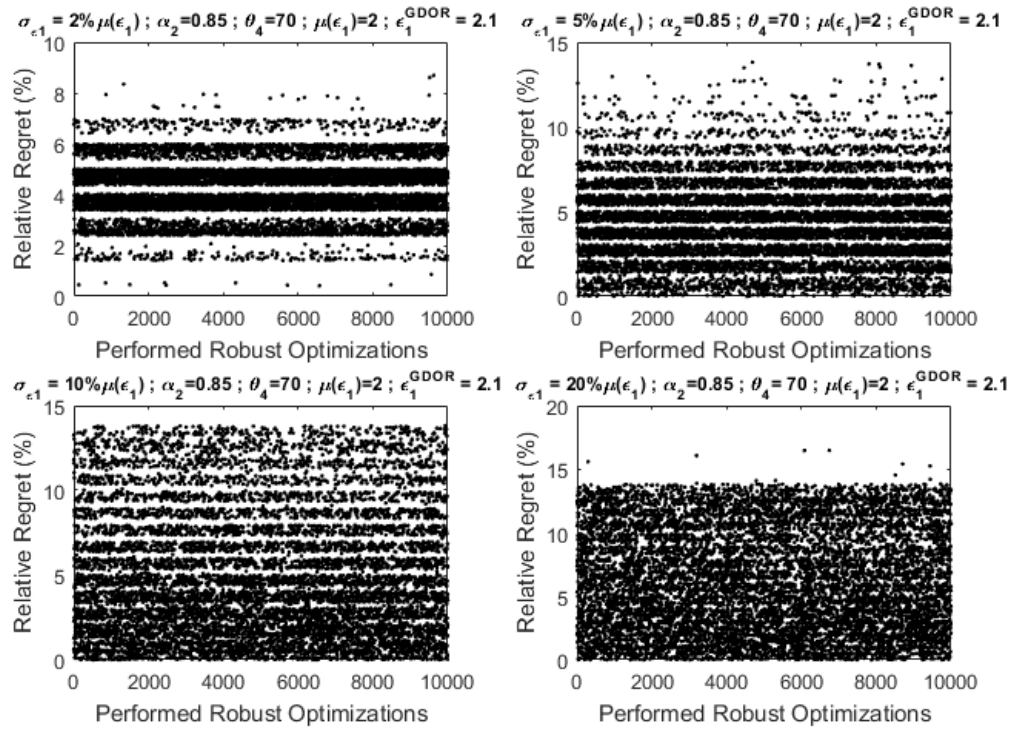


**Fig. 15.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20$  % of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 1.95$ .

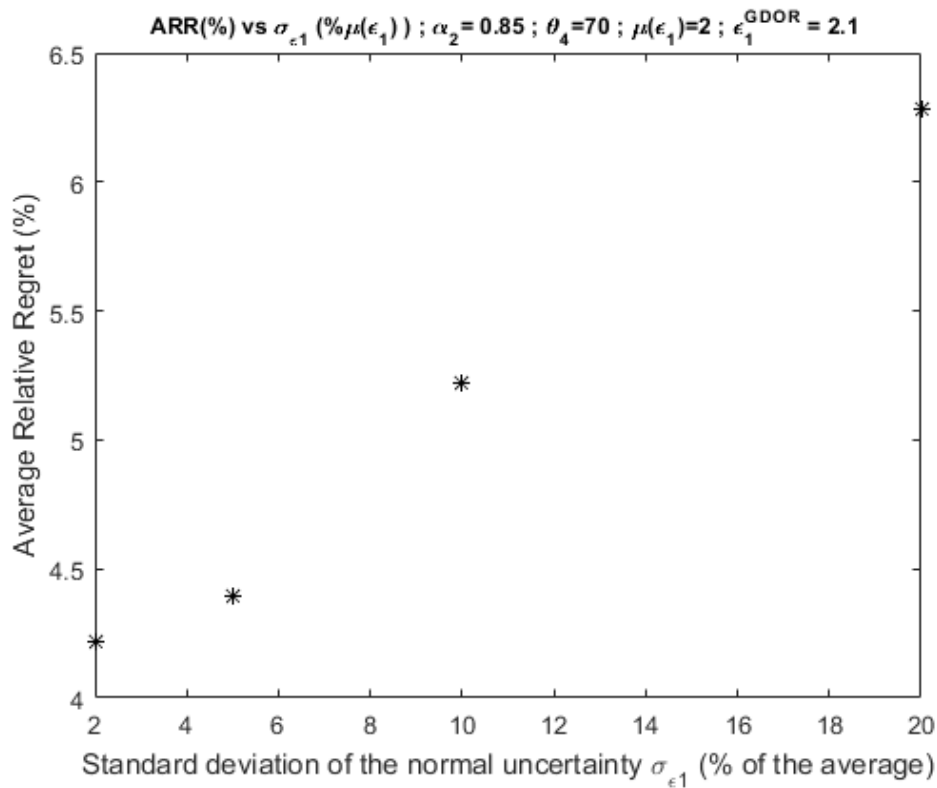


**Fig. 16.** ARR (%) vs  $\sigma_{\epsilon_1} (\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 1.95$ .

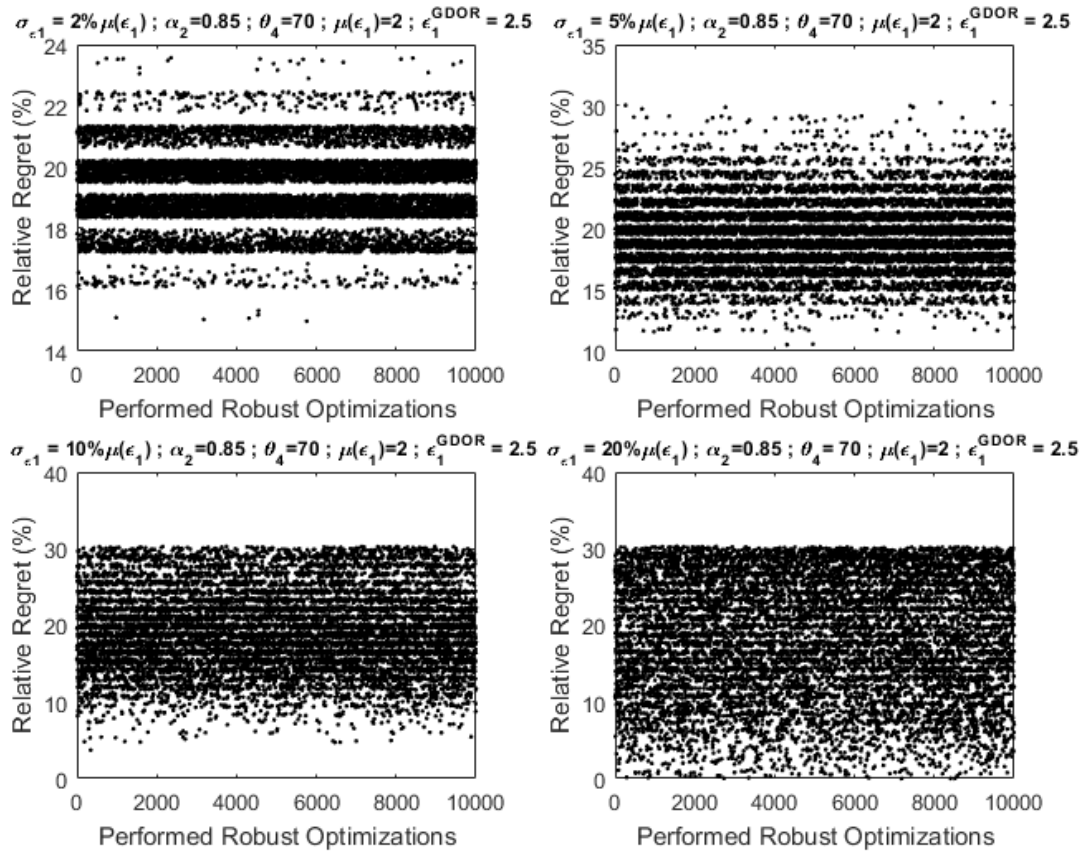




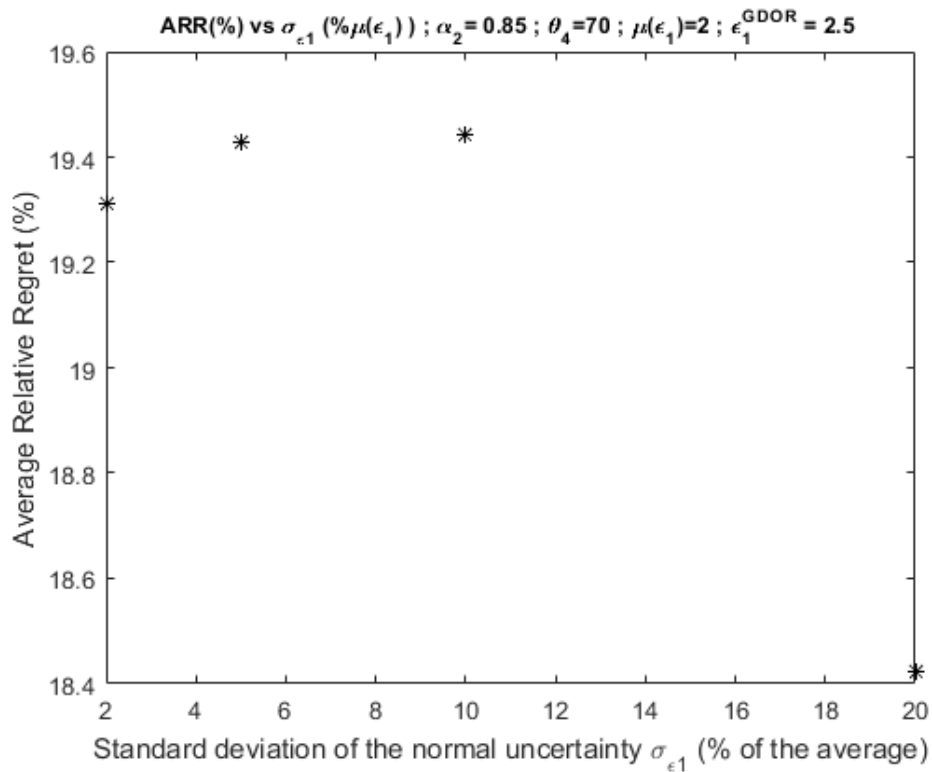
**Fig. 17.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20$  % of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 2.1$ .



**Fig. 18.** ARR (%) vs  $\sigma_{\epsilon_1}(\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 2.1$ .



**Fig. 19.** Graphs of the relative regret (%) vs performed ROs for  $\sigma_{\epsilon_1} = 2, 5, 10, 20$  % of  $\mu(\epsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 2.5$ .



**Fig. 20.** ARR (%) vs  $\sigma_{\epsilon_1}(\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 0.85, \theta_4 = 70, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 70, \epsilon_1 = 2.5$ .



The results from the graphs in Figs. 5 to 12, and shown in Table 3, indicate the best values for all fine-tuned standard deviations, for each GDOR situation, for the price of type-1 part ( $\varepsilon_1$ ), when  $\alpha_2 = 1$  and  $\theta_4 = 100$ , so that the optimal sales planning under uncertainty can achieve the best RO, i.e., get closer to GDOR. Table 4 shows the best values for all fine-tuned standard deviations when  $\alpha_2 = 0.85$  and  $\theta_4 = 70$ . The criterion for choosing each best value for a standard deviation is the one with the lowest ARR as established in algorithm's step vi).

**Table 3**

Best standard deviation values  $\sigma_{\varepsilon_1}$  (final fine-tuning values) for each GDOR situation of all instances of the first case.

Situation of GDOR	$\mu(\varepsilon_1)_{LRR}$	Tuned $\sigma_{\varepsilon_1}$
$\varepsilon_1 = 1.9$	2	5% $\mu(\varepsilon_1)_{RRT}$ (Fig. 6)
$\varepsilon_1 = 1.95$	2	2% $\mu(\varepsilon_1)_{RRT}$ (Fig. 8)
$\varepsilon_1 = 2.1$	2	10% $\mu(\varepsilon_1)_{RRT}$ (Fig. 10)
$\varepsilon_1 = 2.5$	2	20% $\mu(\varepsilon_1)_{RRT}$ (Fig. 12)

**Table 4**

Best standard deviation values  $\sigma_{\varepsilon_1}$  (final fine-tuning values) for each GDOR situation of all instances of the second case.

Situation of GDOR	$\mu(\varepsilon_1)_{LRR}$	Tuned $\sigma_{\varepsilon_1}$
$\varepsilon_1 = 1.9$	2	2% $\mu(\varepsilon_1)_{RRT}$ (Fig. 14)
$\varepsilon_1 = 1.95$	2	2% $\mu(\varepsilon_1)_{RRT}$ (Fig. 16)
$\varepsilon_1 = 2.1$	2	2% $\mu(\varepsilon_1)_{RRT}$ (Fig. 18)
$\varepsilon_1 = 2.5$	2	20% $\mu(\varepsilon_1)_{RRT}$ (Fig. 20)

The tendency of each bell curve to generate more values in certain ranges inherent to each standard deviation can be seen in the graphs of Relative Regret (%) vs Performed ROs in Figs. 5 to 20. They show a higher frequency of single points generated in these bands (due to the greater area under the normal curve), and therefore a higher frequency of solutions obtained that are linked to these bands. As shown in these figures, the ranges and frequencies change for each different standard deviation value and each set of specifications of input parameters.

Tables 4 and 5 show that the framework methodology can also lead to results of fine-tuning for the standard deviation that prove that it is not necessary to have a risk aversion to obtain the best results for models of regret in RO when the GDOR and RRT philosophy is embraced.

Two other fine-tunings were performed for the example problem when  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$  for RRT, and  $\theta_4 = 100, \varepsilon_1 = 1.9$  for GDOR (same conditions as in Fig. 6). Figs. 21 to 24 show that the large subjective number mentioned in Step iv) does not necessarily need to be very large as the amount of optimizations performed in the fine-tuning shown by Figs. 5 to 20, when the problem is linear and the OF is monotonic. The results of the fine-tuning and the specifications are shown in Table 5. The number of optimizations performed for each standard deviation value assumed in the fine-tuning of Figs. 5 to 20 was 10,000, while that of the fine-tuning of Figs. 21 and 22 was 100 and that in the fine-tuning of the Figs. 23 and 24 was 50. As shown in Figs. 6, 22 and 24, the value of the fine-tuned standard deviation is the same

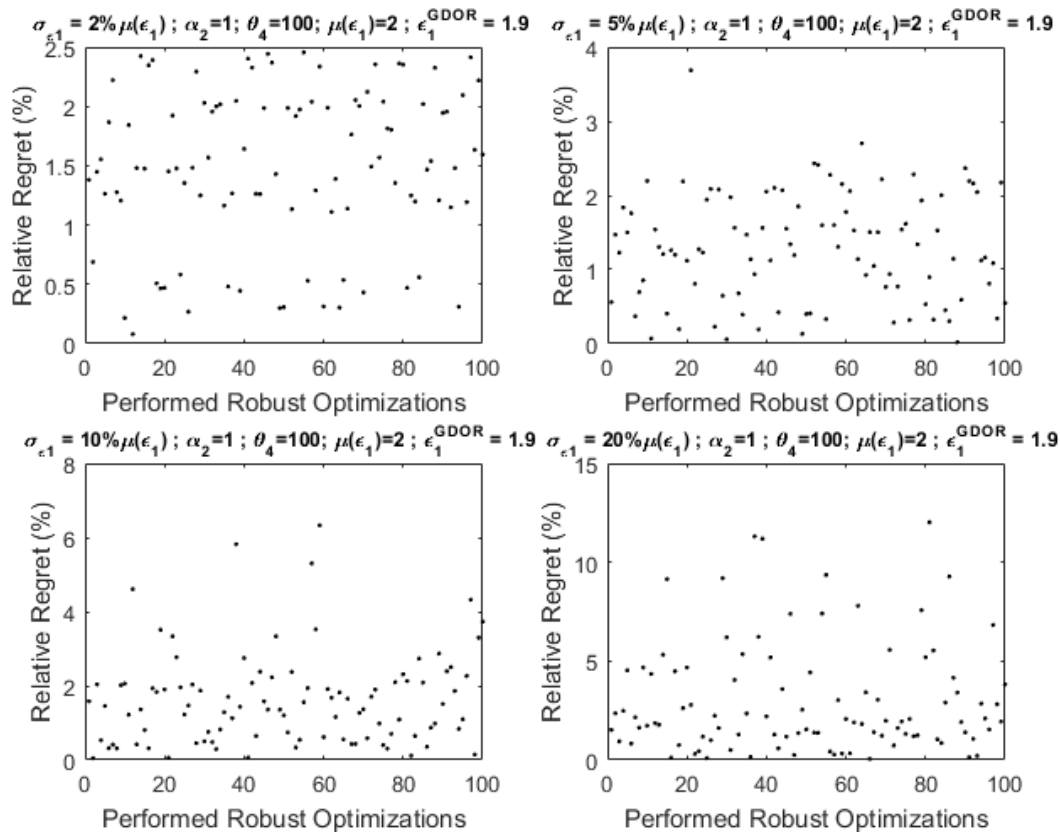
(5%  $\mu(\varepsilon_1)_{RRT}$ ), the values of  $ARR^\sigma$  (%) are very similar, and the trends are the same. Step  $\nu$ ) favors that the large subjective number of optimizations performed does not need to be exaggeratedly large, as the decision making for choosing the fine-tuned standard deviation is based on an average, i.e., by calculating the  $ARR^\sigma$  (%) according to Eq. 7, to which another physical meaning of  $ARR^\sigma$  (%) is the average distance of the solutions from the realizations in relation to the ideal solution. However, if the OF were not monotonic, this average distance would not follow an order relation, making it necessary to perform several further optimizations to obtain a statistically consistent fine-tuning (statistical confidence).

**Table 5**

Fine-tuning results for the case  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$  for RRT, and  $\theta_4 = 100, \varepsilon_1 = 1.9$  for GDOR (conditions of the Fig. 6) for different realizations.

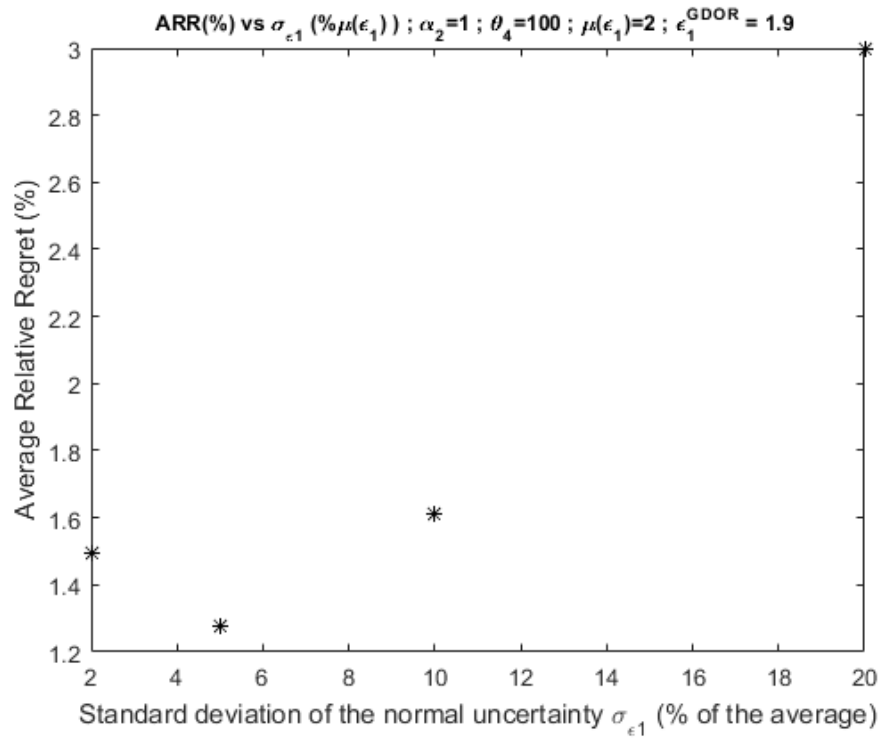
Quantity of performed optimizations <sup>‡</sup>	Tuned value of $\sigma_{\varepsilon_1}$
10000	5% $\mu(\varepsilon_1)_{RRT}$ (Fig. 6)
100	5% $\mu(\varepsilon_1)_{RRT}$ (Fig. 22)
50	5% $\mu(\varepsilon_1)_{RRT}$ (Fig. 24)

<sup>‡</sup>: Just like the other fine-tunings, this one was performed several times for each of these quantities, and the result remains unchanged. All the algorithms implemented in MATLAB® are given as supplementary materials.

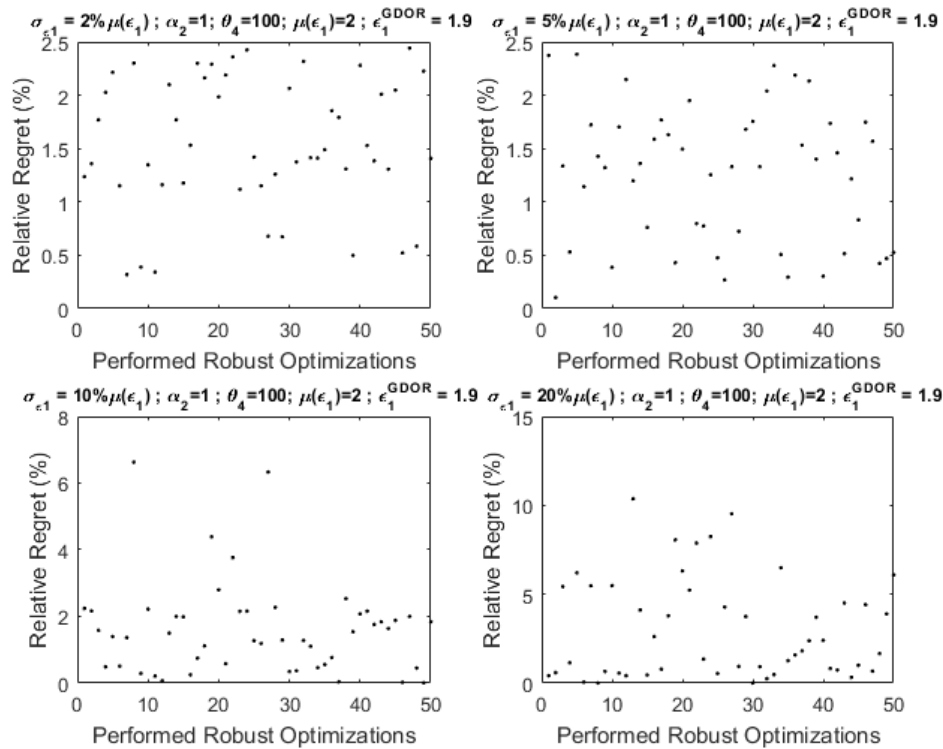


**Fig. 21.** Graphs of the Relative Regret (%) vs Performed ROs for  $\sigma_{\varepsilon_1} = 2, 5, 10, 20\%$  of  $\mu(\varepsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 1.9$ .

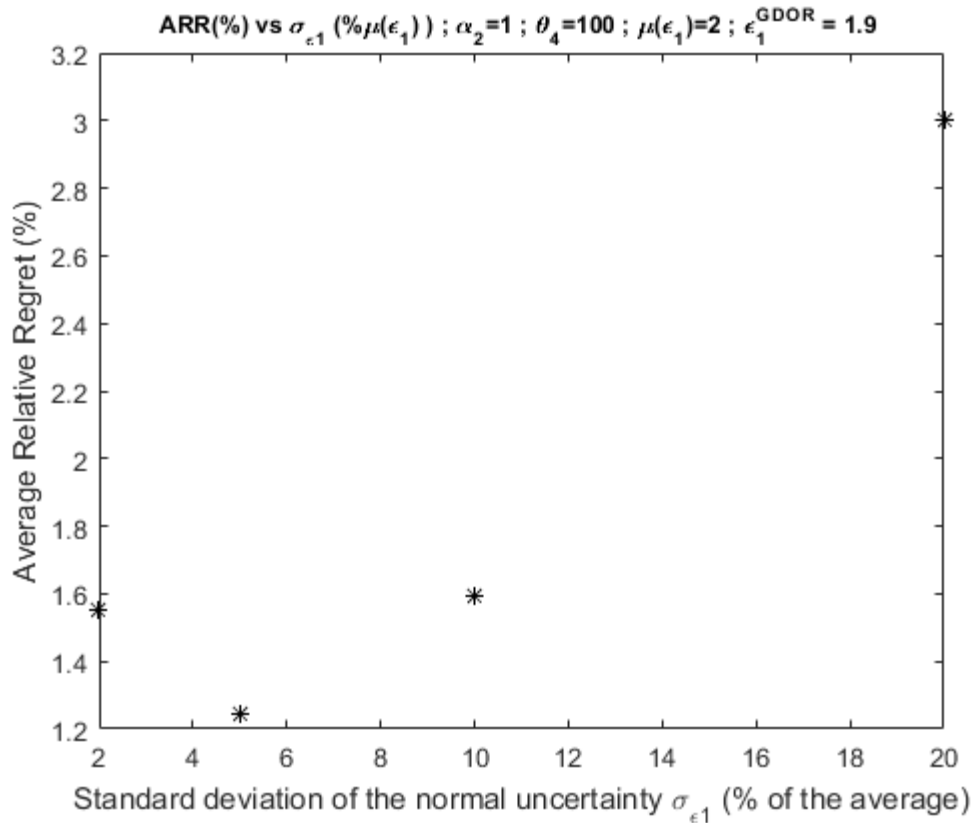
$$N^{BIG} = 100.$$



**Fig. 22.** ARR (%) vs  $\sigma_{\varepsilon_1}(\% \mu(\varepsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 1.9$ .  $N^{BIG} = 100$ .



**Fig. 23.** Graphs of the Relative Regret (%) vs Performed ROs for  $\sigma_{\varepsilon_1} = 2, 5, 10, 20$  % of  $\mu(\varepsilon_1)$ , where the recent reference timeline (RRT) for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 1.9$ .  $N^{BIG} = 50$ .



**Fig. 24.** ARR (%) vs  $\sigma_{\epsilon_1}(\% \mu(\epsilon_1))$ , where the recent reference timeline (RRT) for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 1.9, N^{BIG} = 50$ .

Generation of random numbers by Monte Carlo simulation provides two properties for the standard deviation of a normal uncertainty: i) range and ii) population concentration. These properties are designed by the shape of the normal curve. That is why few optimizations are needed to fine-tune the model, as the RR (%) will have exclusive ranges and exclusive high concentrations in monotonic objective functions in the feasible region, for different values of standard deviation. Other than that, the less horizontal the OF curve is (the less smooth the function in the feasible region), the less optimizations will be necessary because the highest concentrations of RR population (%) will have more distinct and separate ranges from each other; therefore, the greater is the difference between values of  $ARR^\sigma$  (e.g., Figs. 15 and 20). In addition, the greater the difference in value between the  $ARR^\sigma$ , the smaller is the number of  $N^{BIG}$  required because the more non-smooth is the OF.

The standard deviation value equal to 5%.  $\mu(\epsilon_1)_{RRT}$  in Figs. 6, 22, and 24 represents an improvement of approximately 114% compared to the standard deviation value equal to 20%  $\mu(\epsilon_1)_{RRT}$ , and an improvement of approximately 14% compared to the value equal to 2%  $\mu(\epsilon_1)_{RRT}$ . Table 6 shows the improvement between using the best value against the worst standard deviation in Figs. 6, 8, 10, 12, 14, 16, 18, and 20. The improvement calculation is done by subtracting, in absolute value, the ARR value of the worst standard deviation value by the ARR value of the best standard deviation, with the subsequent division by ARR of the best standard deviation value and the multiplication by 100%.

**Table 6**

The improvement between using the best value versus the worst standard deviation value in Figs. 6, 8, 10, 12, 14, 16, 18, and 20.

Figures	Best value of $\sigma_{\varepsilon_1}$	Worst value of $\sigma_{\varepsilon_1}$	Improvement (%)
6	5% $\mu(\varepsilon_1)_{RRT}$	20% $\mu(\varepsilon_1)_{RRT}$	159
8	2% $\mu(\varepsilon_1)_{RRT}$	20% $\mu(\varepsilon_1)_{RRT}$	543
10	10% $\mu(\varepsilon_1)_{RRT}$	2% $\mu(\varepsilon_1)_{RRT}$	53
12	20% $\mu(\varepsilon_1)_{RRT}$	2% $\mu(\varepsilon_1)_{RRT}$	417
14	2% $\mu(\varepsilon_1)_{RRT}$	20% $\mu(\varepsilon_1)_{RRT}$	667
16	2% $\mu(\varepsilon_1)_{RRT}$	20% $\mu(\varepsilon_1)_{RRT}$	152
18	2% $\mu(\varepsilon_1)_{RRT}$	20% $\mu(\varepsilon_1)_{RRT}$	50
20	20% $\mu(\varepsilon_1)_{RRT}$	2% $\mu(\varepsilon_1)_{RRT}$	6

Now suppose that the mathematical model of the same example was hypothetically nonlinear as Eq. (9) (note: the purpose of this fine-tuning is to show the functioning for nonlinear problems in situations that abide by the framework's axioms):

$$\max_x f(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi}) = \frac{4x_1^2}{x_2} \varepsilon_1 + x_2 \alpha_2 E\{\xi_2\} + x_1 \theta_1 + x_2 \theta_2 - \theta_3 \quad (9)$$

$$s. t. x_1 + x_2 - UB \leq 0$$

$$x_1 + x_2 - LB \geq 0$$

$$x_2 \alpha_2 E\{\xi_2\} - \theta_4 \leq 0$$

$$\frac{4x_1^2}{x_2} \varepsilon_1 - \theta_5 \leq 0$$

$$x_1 - 20 \geq 0; x_2 - 20 \geq 0; x_1, x_2 \in \mathbb{I}$$

$$0.5 \leq \varepsilon_1 \leq 3.5$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \boldsymbol{\varepsilon} = [\varepsilon_1]; \boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5]; \boldsymbol{\alpha} = [\alpha_2]; \boldsymbol{\xi} = [E\{\xi_2\}]$$

It is necessary to perform the monotonicity test to assess whether this nonlinear OF is monotonic in the feasible region of the optimization problem for each and all  $x_i$  and  $\varepsilon_j$ , so that it is not necessary to perform several optimizations during the fine-tuning by natural increase of statistical confidence. This test was not done for the linear version of this example because it was obvious that the OF was monotonic as it was linear and all terms containing  $x_i$  and  $\varepsilon_j$  were mathematically positive, and there was no division of any term by any  $x_i$  and  $\varepsilon_j$ . The partial derivatives of this nonlinear function are given by Eq. (10) to (12):

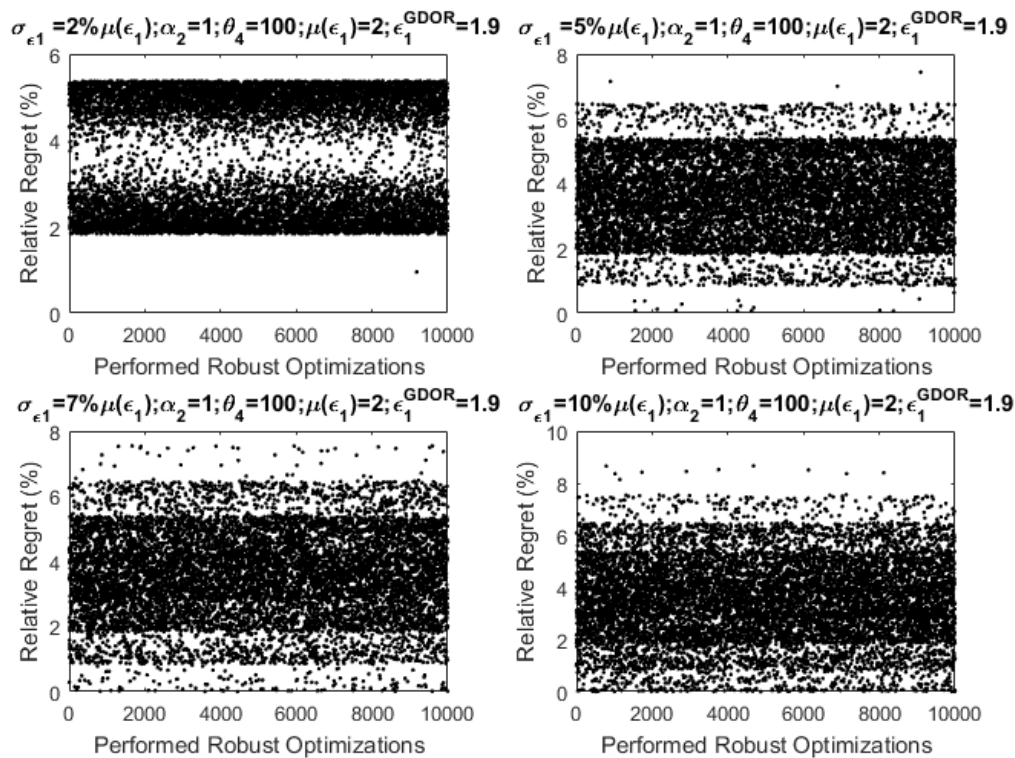
$$\frac{\partial f(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})}{\partial x_1} = \frac{8x_1 \varepsilon_1}{x_2} + \theta_1 \quad (10)$$

$$\frac{\partial f(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})}{\partial x_2} = \alpha_2 E\{\xi_2\} + \theta_2 - \frac{4\varepsilon_1 x_1^2}{x_2^2} \quad (11)$$

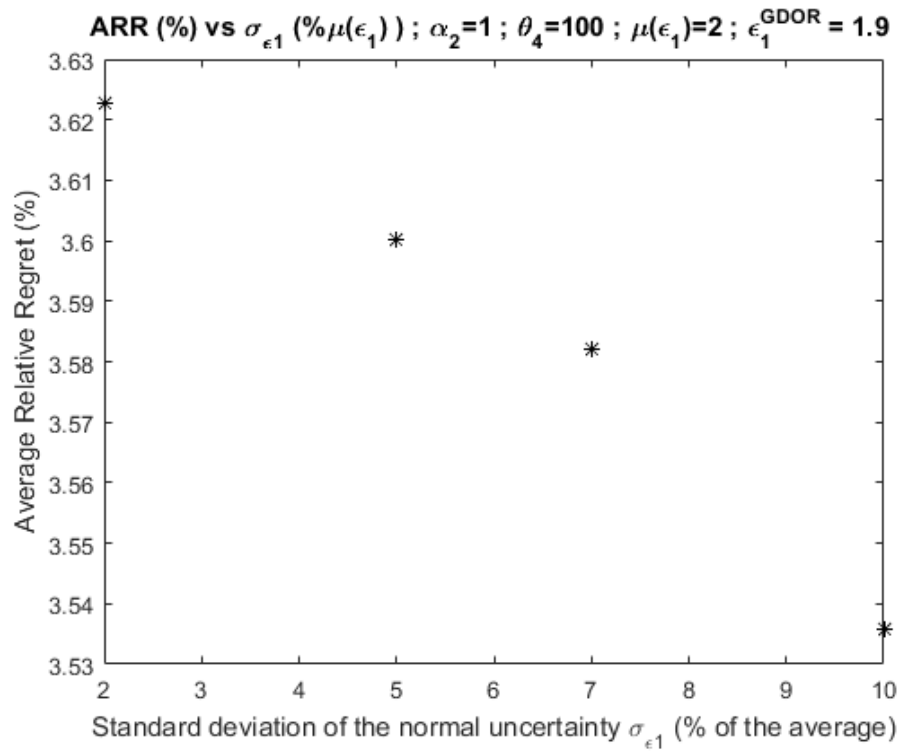
$$\frac{\partial f(\mathbf{x}, \boldsymbol{\varepsilon}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi})}{\partial \varepsilon_1} = \frac{4x_1^2}{x_2} \quad (12)$$

All terms involving the variables  $x_1$  and  $x_2$  and the uncertainty  $\varepsilon_1$  in Eq. (10) and (12) are positive, so the function is strictly increasing and consequently monotonic for  $x_1$  and  $\varepsilon_1$ . However, Eq. (11) has a negative and a positive term, so it is necessary to check whether the order relation of the function will be preserved for  $x_2$  in the feasible region of the optimization problem. There are several ways to test monotonicity in the literature, and it is up to the user how to make the test. For the operational research sector, most models are conducive to having monotonic objective functions that naturally follow the framework's axioms. An algorithm was created and executed in MATLAB® (R2019b, Mathworks, Natick, MA, USA) to check for the monotonicity of this function and can be found as supplementary material. The function also proved to be monotonic for  $x_2$  in the feasible region of the optimization problem. What this algorithm does is to check all the values of the partial derivatives through the DCFR; and to be monotonic, the sign of all partial derivatives must remain unchanged through this domain.

By fine-tuning this problem to the same fine-tuning situation as in the case of Fig. 5 but for values of 2, 5, 7, and 10%.  $\mu(\varepsilon_1)_{RRT}$  for  $\sigma_{\varepsilon_1}$  and knowing that the OF of the GDOR is the same as Eq. (9), the results are shown in Figs. 25 and 26. This fine-tuning was run several times, and the behaviors of  $\sigma_{\varepsilon_1}$  and of  $ARM^\sigma$  (from Fig. 26) remained the same every time. The fine-tuned value for  $\sigma_{\varepsilon_1}$  is 10 % .  $\mu(\varepsilon_1)_{RRT}$ .



**Fig. 25.** Relative Regret (%) vs Performed ROs for 2,5,7, 10%.  $\mu(\varepsilon_1)$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 1.9$  (same conditions as in Fig. 6).  $N^{BIG} = 10000$ .



**Fig. 26.** ARR (%) vs  $\sigma_{\epsilon_1}(\% \mu(\epsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\epsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 100, \epsilon_1 = 1.9$  (same conditions as in Fig. 6).  $N^{BIG} = 10000$ .

Due to the low variations in RR (%) in Fig. 26, it can be inferred that this OF has some plateau for some ranges of the problem variables in the feasible region of the optimization problem and it proves to be a smooth function in the DCFR (i.e., it has a plateau in the DCFR) and in the range of  $\epsilon_1$ . It entails few variations across all  $ARM^\sigma$  and requires a larger  $N^{BIG}$  than if it were linear to fine-tune with statistical confidence. The algorithm can be found in the supplementary material.

In the event that the value of  $f_{obj}^{ODGR}$  is specified to be equal to, say, 335 \$ and the fine-tuning has the same conditions as in Fig. 26, but the OF is given by Eq.(13), there would be no low variations in the RR (%) (shown in Fig.27) because this new OF given by Eq.(13) is less smooth than the one given by Eq.(9), and therefore it needs a smaller number for  $N^{BIG}$ , which in this case was equal to 200, and even if the fine-tuning is done several times, the fine-tuning result is the same. The algorithm can be found in the supplementary material. The improvement between worst value ( $\sigma_{\epsilon_1} = 2\% \mu(\epsilon_1)_{RRT}$ ) and best value ( $\sigma_{\epsilon_1} = 10\% \mu(\epsilon_1)_{RRT}$ ) of the standard deviation of the fine-tuning in Fig. 27 is approximately 7%.

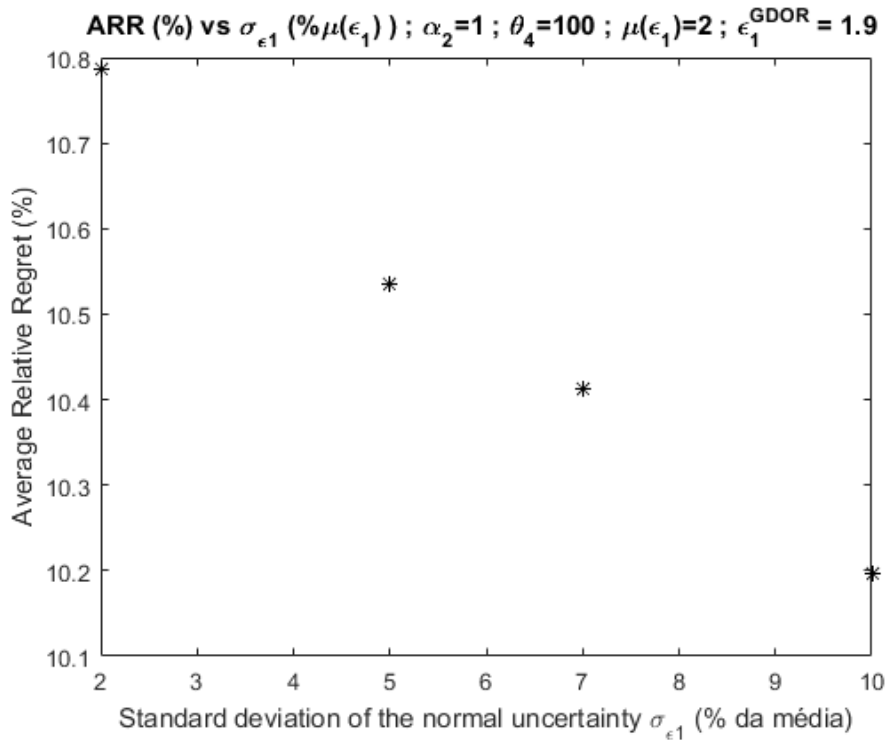
$$\begin{aligned} \max_x f(x, \epsilon, \theta, \alpha, \xi) &= \frac{x_1^2}{x_2} \epsilon_1 + x_2 \alpha_2 E\{\xi_2\} + \frac{x_1^2}{x_2} \theta_1 + \frac{x_2^2}{x_1} \theta_2 - \theta_3 \quad (13) \\ s.t. \quad x_1 + x_2 - UB &\leq 0 \\ x_1 + x_2 - LB &\geq 0 \\ x_2 \alpha_2 E\{\xi_2\} - \theta_4 &\leq 0 \\ \frac{x_1^2}{x_2} \epsilon_1 - \theta_5 &\leq 0 \end{aligned}$$



$$x_1 - 20 \geq 0; x_2 - 20 \geq 0; x_1, x_2 \in \mathbb{I}$$

$$0.5 \leq \varepsilon_1 \leq 3.5$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \varepsilon = [\varepsilon_1]; \theta = [\theta_1 \theta_2 \theta_3 \theta_4 \theta_5]; \alpha = [\alpha_2]; \xi = [E\{\xi_2\}]$$



**Fig. 27.** Graph of average relative regret (%) vs  $\sigma_{\varepsilon_1} (\% \mu(\varepsilon_1))$ , where the RRT for the RO considers  $\alpha_2 = 1, \theta_4 = 100, \mu(\varepsilon_1) = 2$ , to which the GDOR is performed with  $\theta_4 = 100, \varepsilon_1 = 1.9$  (same conditions as in Fig.26).  $N^{BIG} = 200$ .

Different models will have different natural robustness, and for each parameterization, there will also exist a given robustness linked to the model and parameterization. The standard deviation value directly influences the number of solutions that are feasible, as this obviously provides the value of a parameter that can generate feasible solutions or not.

The framework of the present work determines the standard deviation that makes the robust solution closer to ideality ( $f_{obj}^{ODGR}$ ), while naturally reducing the penalty value of the robust solution and regulating the robustness of the model, rather than just depending on the worst case scenario or limited cases. In other words, in addition to increasing the robustness of the model by increasing conservatism, this framework also reduces the penalty in the OF due to its fine-tuning strategy that aims to achieve an ideal solution through the regret model without assigning the scenario tree.

#### 4. Conclusions

RO is a field of high flexibility as to how to deal with the treatability of solutions in an optimization under uncertainty. Different frameworks have been developed, and this work reports on the designing of a

new RO framework based on original concepts with an original regret model and a stochastic and algorithmic numerical strategy to make the robust solution better approach an ideal solution. This methodology considers the uncertainties in the process, at least one of which must be normally distributed.

In addition, the framework provides the regulation of robustness and conservatism by an algorithm that adjusts the penalty factor values, which are specified out or within a loop. As this is an alternative new avenue in RO, some aspects can be studied in-depth and changed to analyze the behaviors of the fine-tune methodology by changing different aspects – e.g., by changing how the penalty factor is calculated/specified.

The results of the applied examples showed that depending on the philosophy of the RO framework used, risk aversion for choosing values close to the average for the uncertainties is not always the best option to achieve the best interest in decision making. Besides, an in-depth study can be carried out to investigate changes that can be made in the methodology for its applicability to non-monotonic objective functions in the DCFR.

The algorithm is flexible and its steps can be reformulated for further studies in order to generate new tenets of how robust solutions could also be considered in decision making. On other side, the mathematical formulation is also flexible and can also be a target of new investigations, e.g. – either the inclusion or combination of new tradeoff parameters or terms aiming at the balance of conservatism and robustness embodied with the concept of targeting an ideal.

## References

- Assavapokee, T., Realff, M. J., Ammons, J. C., & Hong, I. H. (2008). Scenario relaxation algorithm for finite scenario-based min–max regret and min–max relative regret robust optimization. *Computers & operations research*, 35(6), 2093-2102. <https://doi.org/10.1016/j.cor.2006.10.013>
- Baohua, W., & Shiwei, H. E. (2009). Robust optimization model and algorithm for logistics center location and allocation under uncertain environment. *Journal of Transportation Systems Engineering and Information Technology*, 9(2), 69-74. [https://doi.org/10.1016/S1570-6672\(08\)60056-2](https://doi.org/10.1016/S1570-6672(08)60056-2)
- Ben-Tal, A., & Nemirovski, A. (1998). Robust convex optimization. *Mathematics of operations research*, 23(4), 769-805. <https://doi.org/10.1287/moor.23.4.769>
- Ben-Tal, A., & Nemirovski, A. (1999). Robust solutions of uncertain linear programs. *Operations research letters*, 25(1), 1-13. [https://doi.org/10.1016/S0167-6377\(99\)00016-4](https://doi.org/10.1016/S0167-6377(99)00016-4)
- Ben-Tal, A., & Nemirovski, A. (2000). Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical programming*, 88(3), 411-424. <https://doi.org/10.1007/PL00011380>

Bertsimas, D., & Sim, M. (2004). The price of robustness. *Operations research*, 52(1), 35-53. <https://doi.org/10.1287/opre.1030.0065>

Chen, B., Wang, J., Wang, L., He, Y., & Wang, Z. (2014). Robust optimization for transmission expansion planning: Minimax cost vs. minimax regret. *IEEE Transactions on Power Systems*, 29(6), 3069-3077. <https://doi.org/10.1109/TPWRS.2014.2313841>

El Ghaoui, L., & Lebret, H. (1997). Robust solutions to least-squares problems with uncertain data. *SIAM Journal on matrix analysis and applications*, 18(4), 1035-1064. <https://doi.org/10.1137/S0895479896298130>

El Ghaoui, L., Oustry, F., & Lebret, H. (1998). Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9(1), 33-52. <https://doi.org/10.1137/S1052623496305717>

Jiang, R., Wang, J., Zhang, M., & Guan, Y. (2013). Two-stage minimax regret robust unit commitment. *IEEE Transactions on Power Systems*, 28(3), 2271-2282. <https://doi.org/10.1109/TPWRS.2013.2250530>

[Kouvelis, P., & Yu, G. \(1997\). \*Robust discrete optimization and its applications\* \(Vol. 14\). Springer Science & Business Media.](#)

Soyster, A. L. (1973). Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations research*, 21(5), 1154-1157. <https://doi.org/10.1287/opre.21.5.1154>

Xidonas, P., Mavrotas, G., Hassapis, C., & Zopounidis, C. (2017). Robust multiobjective portfolio optimization: A minimax regret approach. *European Journal of Operational Research*, 262(1), 299-305. <https://doi.org/10.1016/j.ejor.2017.03.041>