Efficient robust optimization and reliability analysis using Robustimizer

O. Nejadseyfi [a,*]

a) Corresponding Author, Engineering Technology Department, University of Twente, o.nejadseyfi@gmail.com

1. Introduction

Robtimizer is a software tool to perform optimization under the influence of uncertainties. Consider a model that represents a process. The model has some input variables and by changing those inputs the output changes. By adjusting the input variables the output changes. If some of those inputs are affected by uncertainties or if they are out of control, the uncertainties will be present in the output.

2. A Simple Example

A simple example is presented to explain robust optimization. Assume a basketball player is practicing a free throw outdoors. The process of basketball free throw can be modeled using the physics of projectile motion. The input of a simple model of projectile motion is the initial velocity and throw angle and the output is the final coordinates of the basketball and the angle of approach toward hoop as shown in Figure 1. A criterion can be used to evaluate the success of throw e.g. as a function of the distance of the center of the ball from the center of the hoop and the angle of approach toward the hoop.

This model can be used to find optimum initial velocity and throw angle that will guaranty a successful free throw. As long as the velocity and throw angle remain the same, the basketball will land on the predicted final coordinate with the predicted angle of approach since this problem is deterministic (Figure 1-a). However, if an uncertain input exists in the process, e.g. a front wind with a changing magnitude, then final coordinates using the optimum velocity and throw angle will not be the same in each try (Figure 1-b). Another noise variable in this process can be the height that the basketball is released. In this case, it is necessary to include these uncertain inputs in the model and consequently the output will be uncertain. The uncertain input and output can be handled statistically e.g. using a normal probability distribution.

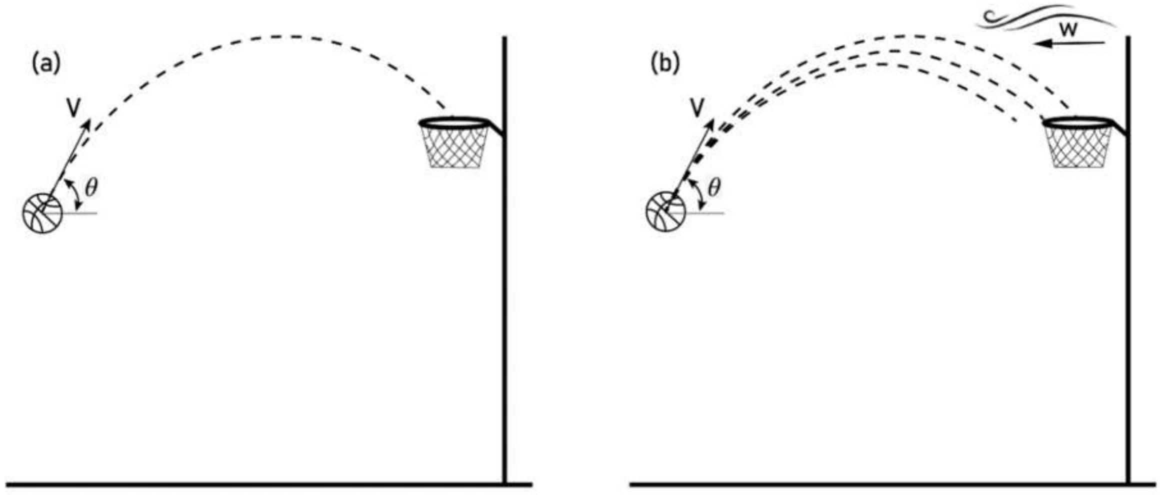

Figure 1: Basketball free throw

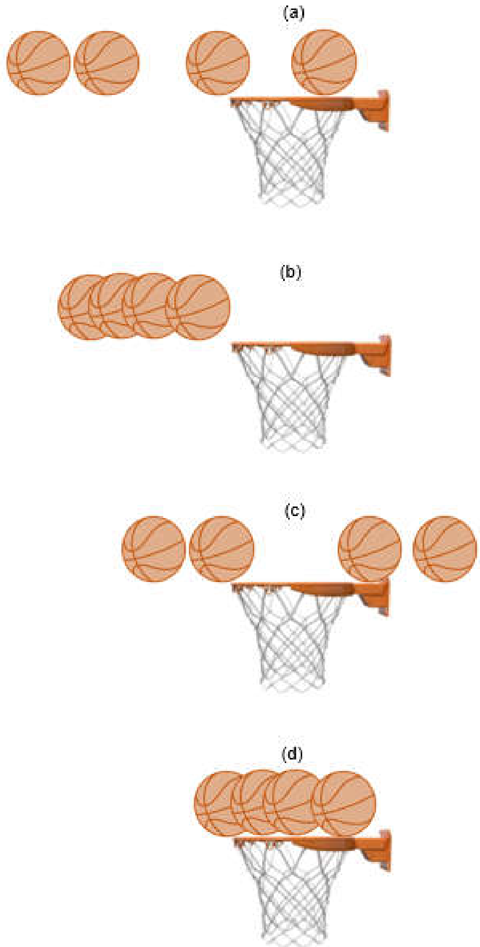

Figure 2: Top view of the hoop (a) mean off target and large scatter (b) mean off target and small scatter (c) mean on target and large scatter (d) mean on target and small scatter

Robust optimization can be used in this case to obtain an initial velocity and a throw angle that increases the success rate while the unwanted inputs exist. The variables that can be adjusted in the process are called design variables. The variables that are out of control or hard to control are called noise variables. In this example, the initial velocity and the throw angle are design variables and wind magnitude and throw height are noise variables.

To perform robust optimization, a proper definition of robustness is required and it can be defined in many different ways. A common approach in robust optimization is to reduce the scatter of output and set the mean on the target. To show why this objective is a good measure for robustness we refer to the basketball free throw example again. Figure 2 schematically shows the top views of the final coordinates of few basketball throws including uncertainties. In Figure 2-a not only the mean is off target but also a large scatter is observed. In Figure 2-b although the scatter is reduced, the mean is still off target. Figure 2-c shows a case in which the mean is on target while scatter is large. Figure 2-d is a desirable situation in which mean is on target and scatter is small. In a desirable situation the throw is both accurate and precise.

In Section 3 the features included in Robtimizer software is introduced.

3. Robustimizer

There are seven tabs as shown in Figure 3 which include the steps to formulate and perform robust optimization. Each tab is explained in detail in the following sections.

3.1. Sensitivity Analysis

Before starting the optimization procedure, the effects of design variables and noise variables on each response can be analysed to reduce the number of inputs. Since models can have many inputs, it is recommended to perform sensitivity analysis to determine the importance of the model inputs and to decide which variables to account for in the optimization [1-3]. This tab includes three other tabs which are the steps towards analyzing sensitivities:

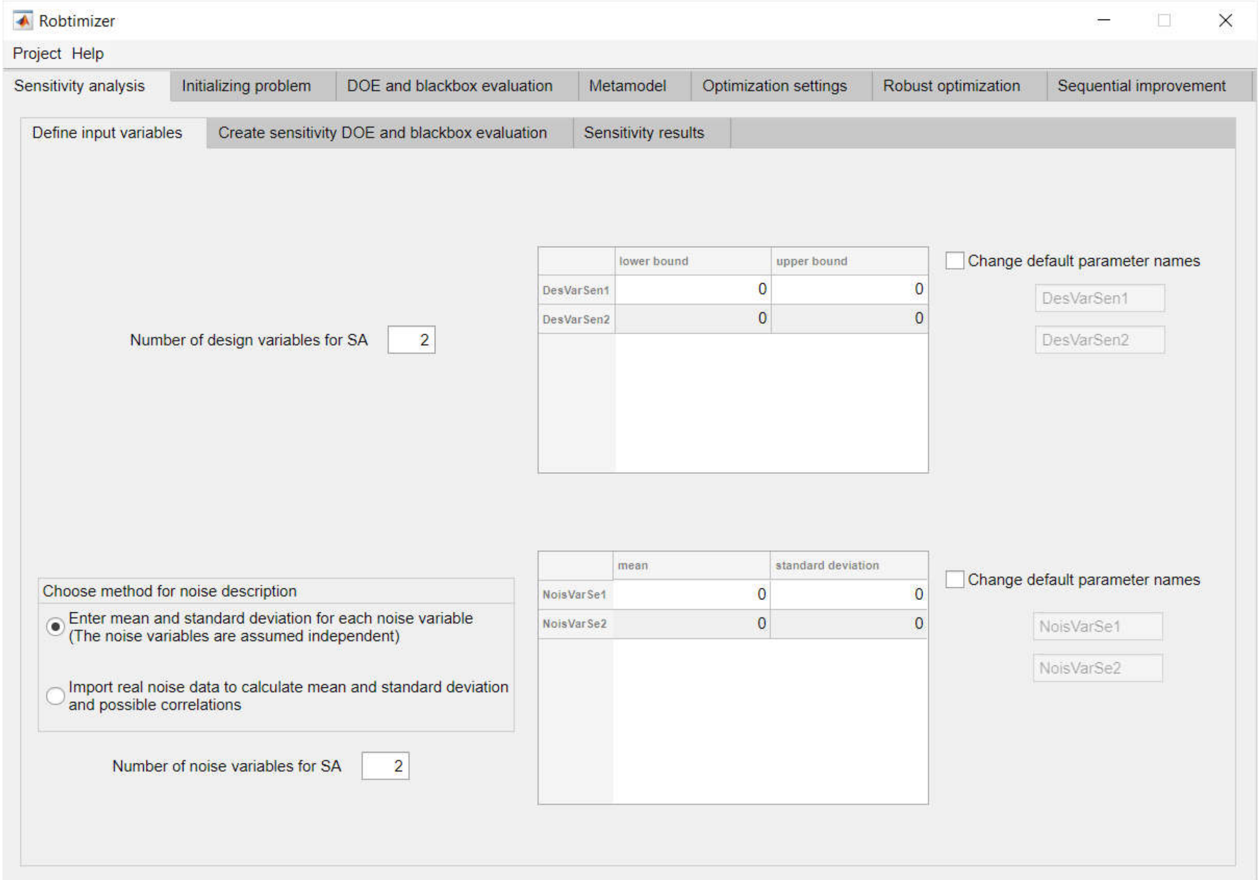

Figure 3: Robustimizer GUI

3.2. Initializing problem

After sensitivity analysis and deciding which variables to consider in the problem, the user defines the main design and noise variables. In this tab the user enters the number of design variables, upper and lower bounds for design variables, and the statistics of noise variables. There are two methods to define noise variables. If there are no measurements or available noise data the first option in choose method for noise description must be selected. After entering the number of noise variables, mean and standard deviation for each variables must be provided. If the second option in choose method for noise description is selected, the user must provide a txt file to calculate the statistics of noise variables and possible correlations. The data for each noise variable must be included in one columns and the columns must be tab separated for multiple noise. If this option is selected, the textbox number of noise variables for SA and noise data table will be disabled and the number of noise variables will be assigned as the number of columns in the file. In addition, the statistics of the noise variables will be evaluated from the provided data.

3.3. DOE and Blackbox Evaluation

In this tab, the user creates a DOE to evaluate the blackbox function. By default Robtimizer creates a Latin hypercube sample (LHS) after 1000 iterations using maximized minimum distance criterion. The user can enter the number of DOE points and decide whether or not to combine LHS with factorial design to improve DOE. By clicking on *Create DOE* button a DOE is created at which the response must be evaluated. This DOE will be shown in the respective table and it can be saved in a txt file with tab-separated columns and it can be loaded later. The user can also provide a custom-made DOE made outside Robtimizer. Importing DOE is not possible when choosing the second option of *choose method for noise description* as DOE must include the possible correlations.

3.4. Metamodel

Use this tab to create, validate and visualize the metamodel. Under *Fit metamodel* panel you can choose the type of metamodel and fit a metamodel. *Cross validation* panel can be used to validate the metamodel. Plotting and visualizing the metamodel is also part of this tab. It must be noted that plotting is performed only in 3 dimensions. One input on X axis and another input on Y axis. The response value will be shown in Z axis. The user can change the variables and responses and update the plot. If more than 2 input are present, the values for other variables will be set in their mean value and the user can change these values using the sliders in front of the name of each variable. Modification of X-axis variable and Y-axis Variable is possible using the drop-down menus.

3.5. Optimization Settings

Default optimization settings can be used to proceed with robust optimization. However, the user has more control on the methods of optimization in this tab shown in Figure 10. Optimization method, noise propagation method, objective function definition and constraints definition can be adjusted in this tab and the relevant edit fields will be enabled or disabled. The objective function to minimize can be chosen in different ways as listed under Objective function for main response panel. If the user is willing to set the mean on target while minimizing the standard deviation, the target value must be set. For a non-normally-distributed main response it is possible to improve the predictions by considering the skewness of the output [4, 5].

If there are constraints, the user can define a target value for the constraint and the type of the constraint for each constraint separately. For a non-normally-distributed constraint response it is possible to improve the predictions by considering the skewness of the response [6].

3.6. Robust Optimization

After performing all the previous steps and defining the problem the user can proceed with robust optimization. By pressing the Perform optimization button, the optimization procedure starts and when the results are generated they will be shown in the text box.

To visualize the scatter of each response at the predicted optimum the user can perform a Monte Carlo analysis at the optimum under the Visualize scatter on optimum panel. The robust optimum obtained using the existing metamodel. The user can improve the metamodel to obtain a more accurate optimum. This can be performed in the next tab.

3.7. Sequential Improvement

In this tab the user can choose a method to add an infill point to the current DOE and perform the optimization again. The default method is Jones criterion [2] with a weighting factor of 0.5. Adding a new infill points can be done either manually or automatically. If the user has defined a script to automatically obtain the results, it is recommended to use the automatic option. In that case by entering the number of sequential improvement steps Robtimizer will automatically find a best infill point, add it to the existing DOE, run the script, fit a new metamodel, find the robust optimum and repeat this procedure. The best infill point and the optimization results will be shown in the text box in the panel for each trial.

For the manual case, the user will obtain a recommended infill point [7] and add it to the existing DOE using the buttons in *Manual infill and optimization* panel. Subsequently the user returns to *DOE and blackbox evaluation* tab to import the results of the blackbox function evaluation using *Read results file* in that tab. It must be noted that the results of all DOE points must be imported. After that the user continues with fitting a new metamodel and finding the robust optimum. The user repeats this cycle as many times as required. The stop criteria can be determined based on the value of expected improvement criterion or change in the optimum design or objective function.

4. References

1. Nejadseyfi, O., H. Geijselaers, and T. van den Boogaard, *Robust optimization based on analytical evaluation of uncertainty propagation.* Engineering Optimization, 2018.
2. Nejadseyfi, O., H.J.M. Geijselaers, and A.H. van den Boogaard. *Efficient calculation of uncertainty propagation with an application in robust optimization of forming processes*. 2017. AIP Publishing LLC.
3. Nejadseyfi, O., H.J.M. Geijselaers, and A.H. van den Boogaard. *Tailoring material scatter for metal forming processes based on inverse robust optimization*. 2019. AIP Publishing LLC.
4. Nejadseyfi, O., *Robust Optimization and Tailoring of Scatter in Metal Forming Processes*. 2019: Gildeprint.
5. Nejadseyfi, O., et al., *Accounting for non-normal distribution of input variables and their correlations in robust optimization.* Optimization and Engineering, 2021: p. 1573-2924.
6. Nejadseyfi, O., H.J.M. Geijselaers, and A.H. van den Boogaard, *Evaluation and assessment of non-normal output during robust optimization.* Structural and multidisciplinary optimization, 2019. **59**(6): p. 1615-1488.
7. Nejadseyfi, O., et al., *From specified product tolerance to acceptable material and process scatter: an inverse robust optimization approach.* International journal of material forming, 2020. **13**(3): p. 1960-6214.