Scenario tree reduction for optimisation under uncertainty using sensitivity analysis

Javier Silvente, Lazaros G. Papageorgiou, Vivek Dua*

Department of Chemical Engineering, Centre for Process Systems Engineering, University College London, Torrington Place, London WC1E 7JE, United Kingdom.

*corresponding author: v.dua@ucl.ac.uk

Abstract

This work addresses the optimal management of a system through a two-stage stochastic Non-Linear Programming (NLP) formulation. This approach uses a scenario-based mathematical formulation to tackle uncertain information. Accurate representation of uncertainty usually involves increased number of scenarios, which may result in large-scale optimisation models. Thus, the proposed formulation aims to reduce the number of scenarios through a sensitivity analysis approach. The proposed model investigates the use of scenario reduction techniques to reduce computational requirements while maintaining good quality of the final optimal solution.

Keywords: Uncertainty, NLP, Scenario reduction, sensitivity analysis, optimisation

1. Introduction

Generally, processes are dynamic, and therefore, different kinds of unexpected events may occur quite frequently. These disturbances may modify the operating conditions, such as the nominal schedule of a production plant or the planning requirements of a supply chain. The incorporation of uncertainty can be critical to ensure the generation of feasible solutions of good quality and practical interest in the decision making.

Thus, some approaches are based on the consideration of a finite number of scenarios representing the uncertainty distribution by discretising the probability of those scenarios, such as the stochastic programming. In this kind of formulation, the obtained solution is generated by satisfying all the considered scenarios. This is an advantage of this formulation since the results are feasible for all the selected scenarios. On the other hand, the obtained solutions may result in being too conservative, because the model must consider all the possibilities. This technique has been used to consider different sources of uncertainty, such as uncertainty in demand, supply of raw materials and processing times (Shapiro *et al.*, 2013), and applied to supply chain management (Schildbach and Morari, 2016) and energy systems (Silvente *et al.*, 2018), among other examples.

Note that the size of the mathematical formulation increases when large number of scenarios are considered. This may result in intractable problems due to the limitation of the computational resources or in large computational times to reach a solution. Hence, an approximation to reduce the large number of scenarios is to find a subset of scenarios that represents the original uncertainty distribution. Thus, scenario reduction techniques are based on selecting representative scenarios and re-assign new probabilities to those scenarios that better represents the original large number of scenarios, in order to reduce the computational effort while maintaining the quality of the solutions as much as possible.

Despite the importance of this topic, it has received limited attention in the literature. Note that scenario reduction techniques can be applied to different areas of interest, such as supply chains (Paulo *et al.,* 2017)*,* electrical markets (Dupačová *et al.*, 2003) hydro-thermal power systems (de Oliveira *et al.*, 2010) and maintenance of units (Qian and Tang, 2017).

In the area of scenario reduction, different heuristic algorithms have been developed. For example*,* Dupačová *et al.* (2003) developed a forward reduction and backward selection methodology to decrease the size of the scenario tree. This avoids solving the combinational optimisation problem. Posteriorly, Heitsch and Römisch (2003) extended both forward and backward methodologies in order to improve the scenario reduction performance. Also, Heitsch and Römisch (2007) improved the previous scenario reduction procedures by considering Fortet-Mourier metrics as an alternative of their upper bounds. This work was extended by relying some on stability performance (Heitsch and Römisch, 2009).

One of the scenario reduction methods is the transportation distance-based scenario reduction, which aims at minimising the probabilistic distance between the original and the reduced selected input scenario distribution (Karuppiah *et al.*, 2010). Moreover, Li and Floudas (2014) developed a Mixed Integer Linear Programming (MILP) for reducing the number of scenarios, by minimising the transportation distance in order to obtain a selected number of scenarios that represents the original scenarios. This methodology also takes into account not only the input performance but also the output performance. The methodology developed by Li and Floudas (2014) is limited by the dimensions of the problem. This method can be applied only to reasonably small formulations, approximately up to 5,000 scenarios. More recently, Li and Floudas (2016) proposed a Linear Programming (LP) scenario reduction method, to tackle the reduction from a very large net of scenarios, where the transportation distance between the original scenario and the reduced scenario subset constitutes the measure to update the scenario selection. Moreover, Li and Li (2016) proposed a linear programming-based scenario reduction, applied to chance constrained optimisation problems. Also, Kovacevic and Pichler (2015) developed a scenario tree reduction technique for linear discrete time formulations for stochastic problems, reducing the difference between an original scenario tree and a reduced one.

Other works are based on clustering techniques for scenario tree reduction. For example, Latorre *et al.* (2007) proposed a formulation considering two phases. In the first phase, the number of scenarios was reduced. In the second phase, clustering was performed to reach the desirable number of scenarios, by considering the most representative ones. Posteriorly, Beraldi and Bruni (2014) presented a clustering approach addressing the scenario tree reduction for multi-stage stochastic formulations. More recently, Chen and Yan (2018) presented a scenario tree reduction formulation throughout clustering tree nodes considering different time periods. The main idea is to cluster nodes with the same parent node into a smaller number of nodes. The proposed algorithm aims at minimising the distance between the initial scenario tree and the reduced tree.

Furthermore, a tool addressing the scenario reduction in stochastic formulations is also available in GAMS. This tool is known as SCENRED (GAMS/SCENRED Documentation), which was extended to SCENRED2 (GAMS/SCENRED2 Documentation). The aim of these algorithms is to provide a reduced scenario subset and to designate new probabilities to the conserved scenarios that comprise the new reduced scenario tree. The reduction algorithm considers a probability distance of the initial and the reduced probability metric, taking into account both scenario probabilities and distances of scenario values. Therefore, removal takes place if scenarios are in neighboured or have minor probabilities. The SCENRED tool was recently applied by Silvente *et al.* (2017) to optimally manage a microgrid, in order to reduce the uncertainty in the heat demand over different time periods.

This work is focused on the reduction of scenarios by evaluating the sensitivity of the uncertainty. While the earlier described works take into account the difference between the original and the new reduced scenario tree, the novel proposed methodology takes into account the outcomes of the model by considering the sensitivity of the uncertainty. This novel approach can be applied to linear and non-linear models. However, this methodology is tested in non-linear formulations, since authors consider that the reduction of scenarios in linear models has received adequate attention, but not the non-linear models. The proposed formulation is limited to a single uncertain parameter. Multiple sources of uncertainty may require further research.

The aim of this methodology is to exploit the new scenario reduction technique through a sensitivity approach to reduce the size of the original scenario tree, in order to obtain a good and practical solution while reducing the computational effort to reach the optimal solution. The methodology is presented through two illustrative Non-Linear Programming (NLP)-based examples, in order to analyse how the optimal solution is affected by the reduction of scenarios and how the computational time decreases. The novel proposed methodology, is going to be referred as RedOpt, denoting optimising by reducing nodes.

This novel methodology proposes to remove nodes according to the analysis and evaluation of the sensitivity associated with the problem under study. Accordingly, nodes with small sensitivity are removed, while nodes with bigger sensitivity are conserved in the scenario tree. This iterative procedure has been applied to remove nodes sequentially with low sensitivity, reducing the full scenario tree. In our proposed examples (see section 3), the original problem is reduced, for example, considering only the first time periods. Then, this problem is extended by introducing more nodes (i.e., nodes corresponding to the following time periods) once some previous nodes have been removed after analysis their sensitivity. Note that the candidate nodes to be removed are those with low sensitivity. So, the sensitivity of the children nodes will also be low. By removing these nodes with low sensitivity, the computational time to perform this analysis is reduced compared with other analysis considering the full stochastic formulation.

This paper is organised as follows. Section 1 has described the main recent advances in the area of scenario reduction. The algorithm to reduce the scenarios by applying the RedOpt approach based on a sensitivity analysis is described in Section 2. Multiple scenarios and the scenario reduction procedure are considered. Thus, a two-stage stochastic NLP formulation is proposed. The effectiveness of the scenario reduction is evaluated through two examples in Section 3. The obtained results are displayed and discussed in Section 4. Finally, concluding remarks are presented in Section 5.

2. RedOpt: Sensitivity analysis approach

Sensitivity methods play a significant role in uncertainty analysis, and therefore, in the decision making, including linear problems (Gal, 1975) and non-linear problems (Fiacco, 1990). Local sensitivity analysis is focused on the study related to how variations in the output of a model can be apportioned and how the given model depends upon the input information. Thus, marginal values associated with a variable can be defined as the variation in the value of an objective function if the aforementioned variable changes its value by one unit. Mathematically, the marginal value can be defined as follows, where Z denotes the objective function, and θ the uncertainty.

$$Marginal\ Value = \frac{dZ}{d\theta}$$

The study of the sensitivity of the model allows creating local maps, used to obtain the objective function as a function of uncertain parameters, considering the local response of the system to variations in the inputs, as well as to reduce the number of scenarios to be considered, while maintaining the quality of the solution and reducing the computational effort.

Consider θ as an uncertain parameter of the mathematical formulation. Thus, a general parametric NLP minimisation (or maximisation) problem can be defined as follows (Dua and Pistikopoulos, 1998; Pistikopoulos *et al.*, 2012):

$$\begin{gathered} Z(\theta) = \min_x f(x,\theta) \\ subject\ to \\ h(x,\theta) = 0 \\ g(x,\theta) \leq 0 \\ x \in \Re^n \\ \theta \in \Re^m \end{gathered} \tag{1}$$

According to this general formulation, θ is a vector of parameters and x is a vector of the decision variables belonging to convex sets Θ and X respectively, and $f(x,\theta)$, $g(x,\theta)$ and $h(x,\theta)$ denote nonlinear functions. Although this general formulation, our study is going to be focused only on a single uncertain parameter.

In addition, the uncertain parameters may be discretised into a finite number of scenarios, thus resulting in a multi-stochastic parametric formulation. A general two-stage stochastic parametric NLP minimisation can be defined as shown in expression (2) (Karuppiah et al., 2010).

$$\begin{gathered} Z(\theta) = \min_x \left[f^0(d) + \sum_s p_s f_s(x_s,\theta_s) \right] \\ subject\ to \\ h_s(d,x_s,\theta_s) = 0 \\ g_s(d,x_s,\theta_s) \leq 0 \\ d \in D \\ x \in X \subseteq \Re^n \\ \theta \in \Theta \subseteq \Re^m \end{gathered} \tag{2}$$

Note that θ is a vector of parameters and x is a vector of the decision variables belonging to convex sets Θ and X respectively, and $f^0(d)$, $f_s(x_s,\theta_s)$, $g_s(d,x_s,\theta_s)$ and $h_s(d,x_s,\theta_s)$ denote nonlinear functions. Notice that $g_s(d,x_s,\theta_s)$ and $h_s(d,x_s,\theta_s)$ include both first and second stage constraints. Also, $s \in S$ denotes a single scenario in the stochastic formulation. Furthermore, d represents the set of the first stage decision variables and x_s the second stage variables. Also, p_s indicates the probability associated with scenario $s \in S$.

Notice that equation (1) represents a general parametric optimisation formulation, depending on x and θ. On the other hand, equation (2) denotes a two-stage parametric problem. So, in other words, equation (2) extends the equation presented in (1) by using a stochastic approach including multiple scenarios. Also, notice that as a difference of equation (2), there is no d in equation (1), since it is a single-stage formulation. While $f(x,\theta)$ is used in equation (1), $f^0(d)$ is used in equation (2), where d denotes the first-stage decisions, and $f_s(x_s,\theta_s)$ contains only second stage constraints.

The simultaneous consideration of all uncertain sources defines the scenario tree. As illustrative example, Figure 1 shows the scenario tree associated with one source of uncertainty that affects different periods of time, considering three uncertain levels.

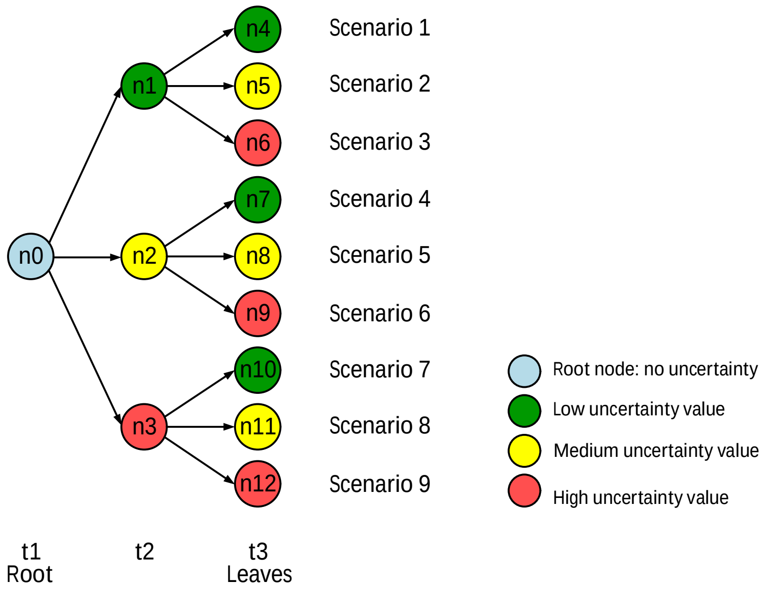

Figure 1. Generic scenario tree representation.

This generic scenario tree is built considering three levels for the uncertain parameter. In this figure, three time intervals are taken into account. The information in the first time interval is considered as certain. This corresponds to the root of the scenario tree. This node $n \in N$ is the parent of nodes $pn\epsilon N$ in the second time interval. These nodes are known as children nodes. Nodes corresponding to last time interval are known as leaves. Note that different levels and different time intervals can be considered.

Thus, each node is assigned to each time period, defining the subset $n\epsilon TN_t$, denoting the nodes that are active at each time interval. Figure 1 may help to explain this subset. For example, the node $n0$ is active only at time $t1$; nodes $n1$, $n2$ and $n3$ are active only at time $t2$; and nodes $n4$ to $n12$ are active only at time $t3$.

Also, the corresponding scenario tree $tree_{n,pn}$ is defined, where n denotes a child node and pn denotes a parent node. The different uncertain levels $l \epsilon L$ define the uncertain level. Thus, a set node $l \epsilon LN_n$ is considered to assign each uncertain level l to each node n. Once again, Figure 1 helps to a better description of this subset. In this figure, three uncertain levels have been considered. In particular, low, medium and high values of the uncertain parameter have been taken into account. This corresponds to $l1$, $l2$ and $l3$, respectively. So, in Figure 1, nodes $n1$, $n4$, $n7$ and $n10$ have associated low value in the uncertain parameter. Thus, nodes $n1$, $n4$, $n7$ and $n10$ are active only for the uncertain level $l1$. In the same way, nodes $n2$, $n5$, $n8$ and $n11$ are active only for the uncertain level $l2$, and nodes $n3$, $n6$, $n9$ and $n12$ are active only for the uncertain level $l3$.

Therefore, the number of nodes and scenarios (i.e., leaf nodes) grows exponentially when more time periods are considered. Table 1 displays the number of nodes and scenarios for different time periods when the same structure as shown in Figure 1 is considered:

Table 1. Number of nodes and scenarios for different time periods, for one uncertain parameter with three levels of uncertainty

Number of time intervals	Number of nodes	Number of scenarios
1	1	1
2	4	3
3	13	9
4	40	27
5	121	81
6	364	243
7	1,093	729
8	3,280	2,187
9	9,841	6,561
10	29,524	19,683
11	88,573	59,049
12	265,720	177,147
13	797,161	531,441
14	2,391,484	1,594,323

The following expressions aim to determine the probability of each node at each time period. This considers the probability at each uncertain level, time period and the parent nodes. So, each uncertain level l at each time interval t has associated a probability $probL_{l,t}$. Thus, equation (3) determines the probability of a node without considering the probability of previous nodes in the scenario tree, $stpdat_{n,t}$ at each time $t \epsilon TH$. The probability $prob_{n,t}$ of node n at time t is given by the probability in the previous nodes, as calculated in equation (4). Furthermore, equation (5) is used to introduce the uncertain parameter $\theta_{n,t}$. Particularly, the variation of a nominal parameter $Var_{l,t}$ at each uncertain level l at each time t is assigned to each node n at each time t. This term will be used to affect the value of a deterministic parameter. Further information can be found in the Examples section. For instance, in equation (6), the demand Dem_t is affected by the variability $\theta_{n,t}$.

$$stpdat_{n,t} = probL_{l,t} \qquad \forall t \epsilon TH, n \epsilon N, n \epsilon TN_t, pn \epsilon TN_{t-1}, n \epsilon tree_{n,pn}, l \epsilon L, l \epsilon LN_n \tag{3}$$

$$prob_{n,t} = stpdat_{n,t} \cdot prob_{pn,t-1} \qquad \forall t \epsilon TH, t > 1, n \epsilon TN_t, pn \epsilon TN_{t-1}, n \epsilon tree_{n,pn} \tag{4}$$

$$\theta_{n,t} = Var_{l,t} \qquad \forall t\epsilon TH, n\epsilon N, n\epsilon TN_t, pn\epsilon TN_{pn,t-1}, n\epsilon tree_{n,pn}, l\epsilon L, l\epsilon LN_n \qquad (5)$$

After reducing the original scenario tree, the probabilities will change. The probabilities in the new scenario tree are defined by $sprob_{n,t}$. Initially, the values of $prob_{n,t}$ and $sprob_{n,t}$ are equivalent.

2.1. *Strategies for the removal of nodes*

There are different strategies for the removal of nodes. In the next figure, we assume that the root plus the first time interval has been solved (nodes $n0$, $n1$, $n2$ and $n3$). After the evaluation of the sensitivity analysis, let us assume that the lower sensitivity corresponds to node $n1$. Figure 2 highlights (blue dotted line) node $n1$ and children nodes of $n1$ ($n4$, $n5$, $n6$).

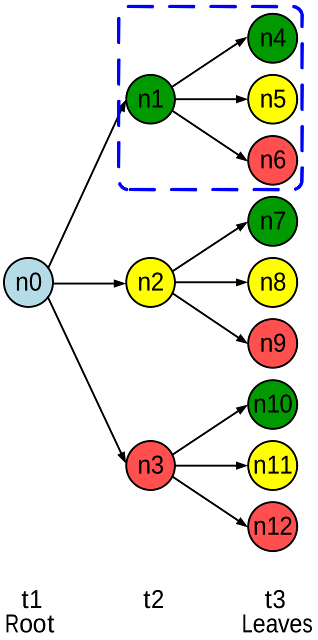

Figure 2. Scenario tree representation, where node $n1$ has less sensitivity.

One strategy is to remove the node with lower sensitivity, as well as the children nodes associated with this parent node. In our case, this corresponds to remove the parent node $n1$ and his children $n4$, $n5$ and $n6$. The probability to nodes $n2$ and $n3$ have to be re-assigned. The total number of scenarios is reduced from 9 to 6. Figure 3 details the new scenario tree. Re-assignment of probabilities is required, due to this removal. Following the example given by Figure 3, if node $n1$ is removed, then nodes $n2$ and $n3$ are conserved in the scenario tree. As a result, the re-assigned probability of node $n2$ corresponds to the original probability for node $n2$ divided by the summation of the original probabilities of nodes $n2$ and $n3$, which are the conserved nodes. The same procedure is applied for node $n3$.

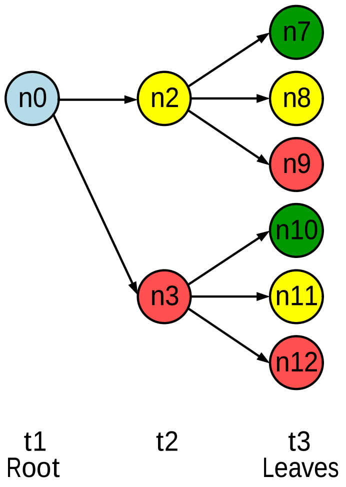

Figure 3. Scenario tree representation when node $n1$ is removed.

The same figure is obtained if the behaviour of nodes $n1$ and $n2$ is the same. This is given if both are in the same critical region (i.e., both nodes have the same sensitivity). Then, the probabilities of $n1$ are aggregated to node $n2$. Since this point, n1 is removed, and only n2 is taken into account, aggregating the probability of $n1$ to $n2$. This strategy is called nodes aggregation. Note that the probability of the removed node is added to the conserved node with the same behaviour. Considering again the figure 3 of the manuscript, node $n1$ is removed and node $n2$ has the same behaviour than node n1 (i.e., same sensitivity). Thus, the new probability of node $n2$ will be the summation of the original probabilities of nodes $n1$ and $n2$. The probability of node $n3$ is not affected.

Another strategy is to consider all nodes ($n1$, $n2$ and $n3$), without removing nodes at this point. However, no branching is considered from the node with lower sensitivity, $n1$, applying the cluster strategy. This is useful when a node has a low probability, but we want to keep this solution because the value of the solution may be highly affected. Re-assignment of probabilities is not required since there is no removal from parent nodes. For example, this situation describes a node with a remote probability but with high impact in the objective function. Figure 4 details the new scenario tree, in which $n1$, $n2$ and $n3$ have been conserved. However, no branching was considered after node $n1$ in its branch. In other words, only the intermediate solutions were considered after this node, so nodes $n4$ and $n6$ were removed, conserving only node $n5$. Notice that for this strategy no re-assignment is required. Considering figure 4, the probability of nodes $n1$, $n2$ and $n3$ are not modified. However, the probability of children nodes of node $n1$ (the node in which the cluster strategy is applied) is the same than the probability of the parent node. For example, the probability of node $n5$ will be the same than the probability for $n1$.

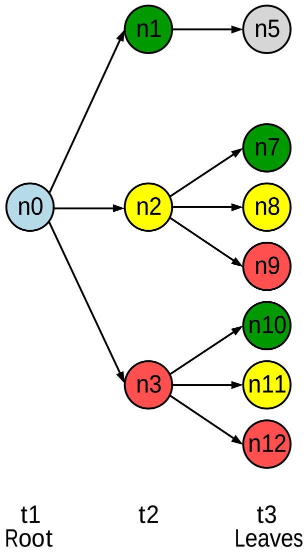

Figure 4. Scenario tree representation when clustering to node $n1$ is applied.

2.2. *Metrics for the removal of nodes*

Different metrics can be applied as criteria for the removal of nodes. One is the relative difference between the sensitivity of one node in comparison with other nodes. This is given by the division of the sensitivity of a node $\theta M_{n,t}$ and the sensitivity of other nodes $\theta M_{n',t}$. According to equation (6), if the value of this division is lower than a pre-established number $Max\theta M$, the rejection can take place, and not otherwise. This ratio should be calculated with absolute value, to consider uncertainty that both increases and decreases the value of the objective function.

$$\left|\frac{\theta M_{n,t}}{\theta M_{n',t}}\right| < Max\theta M \qquad \forall t\epsilon TH, n\epsilon N, n'\epsilon N, n\epsilon TN_t, n'\epsilon TN_t, n' \neq n \qquad (6)$$

Notice that the value of the marginal value takes into account the probability associated with each node. So, we have also to evaluate the value of the marginal value without considering the probability. Then, $\theta P_{n,t}$ parameter is defined in equation (7). A second criterion is the relative difference between the sensitivity of one node in comparison with other nodes, without considering the probability. This is given by the division of the sensitivity of a node $\theta P_{n,t}$ and the sensitivity of other nodes $\theta P_{n',t}$, non-taking into account the probability. According to equation (8), if the value of this division is lower than a pre-established number $Max\theta P$, the rejection can take place, and not otherwise.

$$\theta P_{n,t} = \frac{\theta M_{n,t}}{prob_{n,t}} \qquad \forall t\epsilon TH, n\epsilon N, n\epsilon TN_t \qquad (7)$$

$$\left|\frac{\theta P_{n,t}}{\theta P_{n',t}}\right| < Max\theta P \qquad \forall t\epsilon TH, n\epsilon N, n'\epsilon N, n\epsilon TN_t, n'\epsilon TN_t, n' \neq n \qquad (8)$$

It is worthy to mention that $Max\theta M$ and $Max\theta P$ are used to determine the conserved nodes and the nodes to be removed as well. The number of conserved scenarios depends on the pre-established $Max\theta M$ and $Max\theta P$ parameters, which values are between 0 and 1. Zero means that all scenarios can be removed, while if the value is 1 no removal takes place.

These two parameters ($Max\theta M$ and $Max\theta P$) are initially defined by the user. Notice that the effect of these parameters can be compared with the effect of the tolerance set by the user when using a typical/commercial numerical solver, where higher values offer lower accuracy but faster convergence and vice versa. In this particular case, large values of $Max\theta M$ and $Max\theta P$ (i.e., close to one) involve better quality solution since less scenarios are candidates to be removed, but this requires more computational time to reach the optimal solution. On the other side, lower values (i.e., close to zero) imply lower effort to solve the NLP problem, because less scenarios are conserved in the final scenario tree. In this manuscript, only pre-established values of $Max\theta M$ and $Max\theta P$ have been considered, since the objective of this work aims to present a novel formulation to reduce the number of scenarios in a non-linear formulation.

2.3. RedOpt algorithm

The iterative procedure to solve the RedOpt approach is detailed next:

1. Formulate the mathematical model, including $\theta_{n,t}$ as uncertain parameter. Find two mathematical models in this manuscript, in section 3. The first model comprises equations (9)-(15) whereas the second embraces equations (16)-(27). Define the nodes and the scenario tree. Apply different values in each node for the uncertain parameter.
2. The mathematical model should be solved iteratively. Firstly, consider time intervals corresponding to the root plus one time interval. Considering the scenario tree in the previous figure, this corresponds to time periods $t1$ and $t2$.
3. Solve the mathematical model and evaluate the sensitivity (i.e., marginal values) of variable $\theta_{n,t}$. Evaluate the sensitivity of the model:
   a) If the sensitivity of one node $\theta M_{n,t}$ is too low when compared with other nodes, $\left|\frac{\theta M_{n,t}}{\theta M_{n',t}}\right| < Max\theta M$:
      i) Nodes removal: If the sensitivity without considering probability $\theta P_{n,t}$ is still low, delete this node and the children nodes. Also, re-assignment of probabilities for the conserved nodes is required (see Figure 3).
      ii) Cluster of nodes: If the sensitivity without probability $\theta P_{n,t}$ is not low, this means that this corresponds to a probable remote node. However, this node has to be considered, because although its probability is too low, the objective function is highly affected by this node. After this point, no branching is considered for this node. Re-assignment of probabilities is not required since there is no removal of nodes in the iteration (See Figure 4).
      iii) However, if the sensitivity without probability for two nodes has the same value, go to step (b-iii).
   b) If the sensitivity of one node $\theta M_{n,t}$ is not low when compared with other nodes, then:

iii) Nodes aggregation: If two nodes have associated the same sensitivity without probability, this means that these two nodes are located in the same critical region. Thus, one of them can be removed, because both nodes have the same behaviour. Consequently, the node with lower probability is removed. There is no re-assignment of probabilities, but the probability of the removed nodes should be assigned to the conserved node with the same sensitivity without probability.

iv) If the considered nodes have different values for the sensitivity without probability, all of them should be considered. Thus, there is no removal of nodes.

4. Add another time interval to the current selection and repeat the procedure until the last optimisation period, which corresponds to the leaf nodes. If there are no more time intervals, stop.

Figure 5 displays a graphical algorithm for a better understanding:

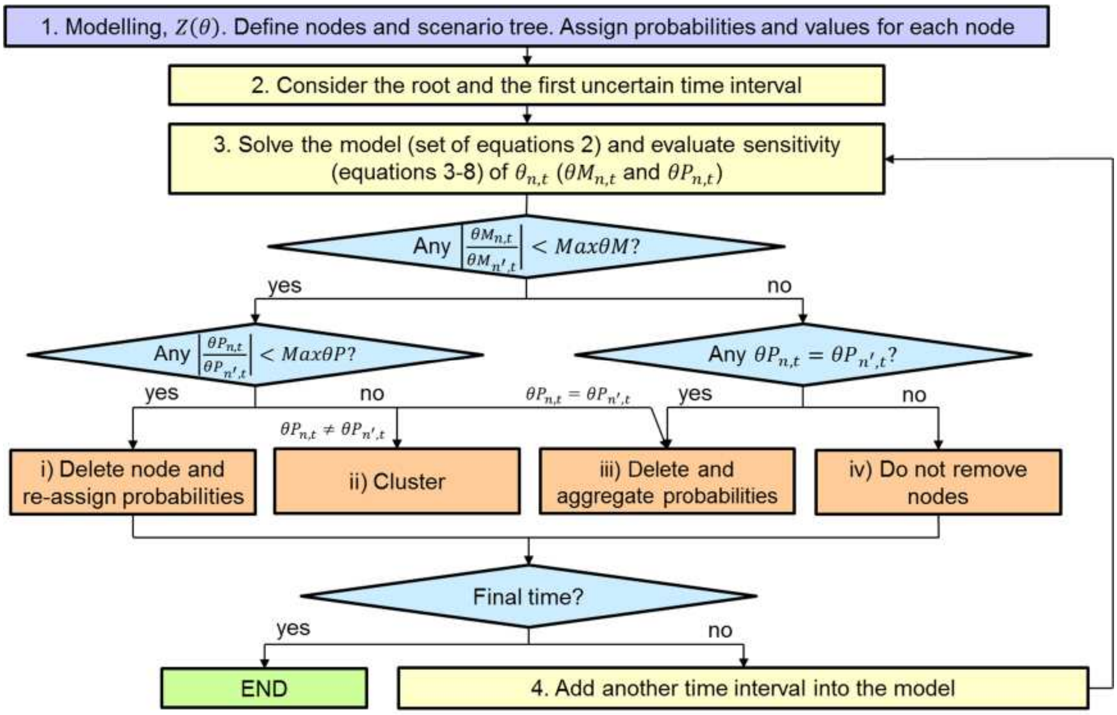
1. Modelling, $Z(\theta)$. Define nodes and scenario tree. Assign probabilities and values for each node
2. Consider the root and the first uncertain time interval
3. Solve the model (set of equations 2) and evaluate sensitivity (equations 3-8) of $\theta_{n,t}$ ($\theta M_{n,t}$ and $\theta P_{n,t}$)
Any $\left|\frac{\theta M_{n,t}}{\theta M_{n',t}}\right| < Max\theta M?$
yes
no
Any $\left|\frac{\theta P_{n,t}}{\theta P_{n',t}}\right| < Max\theta P?$
yes
no
$\theta P_{n,t} = \theta P_{n',t}$
yes
Any $\theta P_{n,t} = \theta P_{n',t}?$
no
$\theta P_{n,t} \neq \theta P_{n',t}$
i) Delete node and re-assign probabilities
ii) Cluster
iii) Delete and aggregate probabilities
iv) Do not remove nodes
Final time?
yes
no
END
4. Add another time interval into the model

Figure 5. Generic algorithm for the RedOpt approach.

3. Examples

Two motivating examples are proposed to analyse how the reduction of scenarios affects the optimal solution. Thus, two examples of the GAMS library have been selected.

3.1. Optimal pricing and extraction for the OPEC

The first example is an NLP that tries to manage the optimal pricing and extraction for the Organisation of the Petroleum Exporting Countries (OPEC), (Pindyck, 1978). This deterministic problem (GAMS library, Optimal Pricing and Extraction for OPEC problem) has been transformed into a stochastic model, considering different scenarios. Notice that uncertainty has been taken into account through the introduction of $\theta_{n,t}$, which is the variation from the nominal conditions. In this example, this variation affects the overall demand. Consequently, equation (9) includes this term $\theta_{n,t}$. Further information of this mathematical model can be found in Pindyck, (1978). Equation (9) determines the total demand $TD_{n,t}$. The non-linear equation (10) establishes the supply of oil by non-OPEC countries, $S_{n,t}$. Constraint (11) is the accounting equation to determine the cumulative supply by non-OPEC countries, $CS_{n,t}$. Equation (12) is used to determine the demand equation for OPEC countries, $D_{n,t}$. Constraint (13) establishes the balance equation to determine the OPEC reserves, $R_{n,t}$. Equation (14) determines the yearly objective function, which are the revenues $Rev_{n,t}$. The objective of this problem is to maximise the profit Z, given by equation (15).

Thus, the stochastic model is defined as follows in equations (9)-(15).

$$TD_{n,t} = 0.87 \cdot TD_{pn,t-1} - 0.13 \cdot P_{n,t} + Dem_t + \theta_{n,t} \qquad \forall t\epsilon TH, n\epsilon SN, pn\epsilon SN, n\epsilon TN_t, pn\epsilon TN_{t-1}, n\epsilon tree_{n,pn} \tag{9}$$

$$S_{n,t} = 0.75 \cdot S_{pn,t-1} + \left(1.1 + 0.1 \cdot P_{n,t}\right) \cdot 1.02^{-CS_{n,t}/7} \qquad \forall t\epsilon TH, n\epsilon SN, pn\epsilon SN, n\epsilon TN_t, pn\epsilon TN_{t-1}, n\epsilon tree_{n,pn} \tag{10}$$

$$CS_{n,t} = CS_{pn,t-1} + S_{n,t} \qquad \forall t\epsilon TH, n\epsilon SN, pn\epsilon SN, n\epsilon TN_t, pn\epsilon TN_{t-1}, n\epsilon tree_{n,pn} \tag{11}$$

$$D_{n,t} = TD_{n,t} - S_{n,t} \qquad \forall t\epsilon TH, n\epsilon TN_t, n\epsilon SN \tag{12}$$

$$R_{n,t} = R_{pn,t-1} - D_{n,t} \qquad \forall t\epsilon TH, n\epsilon SN, pn\epsilon SN, n\epsilon TN_t, pn\epsilon TN_{t-1}, n\epsilon tree_{n,pn} \tag{13}$$

$$Rev_{n,t} = D_{n,t} \cdot \left(P_{n,t} - \frac{250}{R_{n,t}}\right) - D_{n,t} \qquad \forall t\epsilon TH, n\epsilon TN_t, n\epsilon SN \tag{14}$$

$$Z = \sum_{\substack{n \\ n\epsilon TN_t \\ n\epsilon SN}} \sum_{\substack{t \\ t\epsilon TH}} sprob_{n,t} \cdot Rev_{n,t} \cdot 1.05^{1-T_t} \tag{15}$$

Next, all data used to solve this mathematical model is displayed. Table 2 shows the deterministic demand at each time period. Uncertainty in the demand is managed by considering three different demand levels at each time period. These three levels correspond to low, medium and high demand levels. The probability associated with each demand level $probL_{l,t}$ and the variations from the nominal demand $Var_{l,t}$ and the demand for each demand are displayed in Table 2.

Table 2. Data for example 1.

Time	Deterministic demand, Dem_t	$l1$ = Low demand		$l2$ = Medium demand		$l3$ =High demand	
		$probL_{l,t}$	$Var_{l,t}$	$probL_{l,t}$	$Var_{l,t}$	$probL_{l,t}$	$Var_{l,t}$
$t1$	3.300	0		100	0	0	
$t2$	3.334	11	-0.3334	45	0	44	0.3334
$t3$	3.370	16	-0.3370	50	0	34	0.3370
$t4$	3.405	29	-0.3405	15	0	56	0.3405
$t5$	3.441	36	-0.3441	36	0	28	0.3441
$t6$	3.478	43	-0.3478	19	0	38	0.3478
$t7$	3.515	29	-0.3478	39	0	32	0.3478
$t8$	3.553	19	-0.3515	53	0	28	0.3515
$t9$	3.591	26	-0.3553	49	0	25	0.3553
$t10$	3.630	30	-0.3591	39	0	31	0.3591

$t11$	3.669	40	-0.3630	38	0	22	0.3630
$t12$	3.709	21	-0.3669	46	0	33	0.3669
$t13$	3.750	37	-0.3709	28	0	35	0.3709

Furthermore, Table 3 displays the initial values for some variables, describing the initial status of the case under study. This corresponds to the value of the variables for the first node and the first time period.

Table 3. Initial values for example 1.

Variable	Initial value
$TD_{n=0,t=1}$	18.0
$S_{n=0,t=1}$	6.5
$R_{n=0,t=1}$	500
$CS_{n=0,t=1}$	0.0

The values of $Max\theta M$ and $Max\theta P$ have been pre-established to determine and analyse the conserved nodes constituting the final scenario tree. The selected values are 0.9 and 0.6 for $Max\theta M$ and $Max\theta P$, respectively. The nomenclature used to formulate this mathematical model is shown below:

Parameter

Dem_t demand at time t

T_t time period t

Continuous and positive variables

$CS_{n,t}$ cumulative supply by non-OPEC countries in node n at time t

$D_{n,t}$ demand for OPEC-oil in node n at time t

$P_{n,t}$ world price of oil in node n at time t

$R_{n,t}$ OPEC reserves in node n at time t

$Rev_{n,t}$ revenues in node n at time t

$S_{n,t}$ supply of oil by non-OPEC countries in node n at time t

$Td_{n,t}$ total demand for oil in node n at time t

Variables

Z profit (objective function)

3.2. Household optimisation problem

The second example is also an NLP that tries to manage the household optimisation (Fair, 1984). This deterministic problem (GAMS library, Household optimisation problem) has been transformed into a stochastic model, considering different uncertain scenarios. Note that uncertainty has been considered by implementing $\theta_{n,t}$, which is the variation from the nominal conditions. In this example, this variation affects the price. Thereby, equations (21) and (25) include this term $\theta_{n,t}$. Further information of this mathematical model can be found in Fair, (1984). The stochastic model is defined as follows in equations (16)-(27). Constraint (16) is applied to determine the utility function factor, $Ufact_t$. Equation (17) delimits the maximum value of the labour supplied for node n at time t, $F_{n,t}$. The non-linear equation (18) determines the utility at each period, $U_{n,t}$. Constraints (19) and (20) calculate the incomes before and after taxes, $Y_{n,t}$ and $Tax_{n,t}$, respectively. Equation (21) and (22) constitutes savings and the budget constraint, respectively. The non-linear equation (23) is used to determine the time spent on money holdings, $B_{n,t}$. Constraint (24) constitutes the terminal condition for assets, $A_{n,t}$. Equations (25) and (26) are the domain constraints on time money and utility, respectively. The objective is to maximise objective Z, given by equation (27).

$$Ufact_t = \lambda^{T_t - 1} \qquad \forall t\epsilon TH \tag{16}$$

$$F_{n,t} \leq F_t^{max} \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{17}$$

$$U_{n,t} = \left(\alpha \cdot C_{n,t}{}^{-\beta} + (1-\alpha) \cdot \left(TH - F_{n,t} - B_{n,t}\right)^{-\beta}\right)^{\frac{-0.01}{\beta}} \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{18}$$

$$Y_{n,t} = \omega \cdot F_{n,t} + \rho \cdot A_{n,t} \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{19}$$

$$Tax_{n,t} = \delta \cdot Y_{n,t} \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{20}$$

$$S_{n,t} = Y_{n,t} - Tax_{n,t} - \left(Price_t + \theta_{n,t}\right) \cdot C_{n,t} \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{21}$$

$$S_{n,t} = A_{n,t} - A_{pn,t-1} + M_{n,t} - M_{pn,t-1} \qquad \forall t\epsilon TH, n\epsilon TN_t, n\epsilon tree_{n,pn}, pn\epsilon TN_{t-1} \tag{22}$$

$$B_{n,t} \cdot \left(M_{n,t} - \gamma_1 \cdot Pg \cdot C_{n,t}\right) = \gamma_2 \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{23}$$

$$A_{n,t} + M_{n,t} = AM \qquad \forall t\epsilon TH, t = T, n\epsilon TN_t \tag{24}$$

$$1.01 \cdot \gamma_1 \cdot \left(Price_t + \theta_{n,t}\right) \cdot C_{n,t} \leq M_{n,t} \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{25}$$

$$F_{n,t} + B_{n,t} \leq 0.9 \cdot TH \qquad \forall t\epsilon TH, n\epsilon TN_t \tag{26}$$

$$Z = \prod_{\substack{t \\ t\epsilon TH}} \sum_{\substack{n \\ n\epsilon TN_t}} sprob_{n,t} \cdot U_{n,t} \cdot Ufact_t \tag{27}$$

Data used to solve this mathematical model is shown in Table 4, including the deterministic price at each time period, $Price_t$, and the uncertain parameters to be managed, such as the probability of each price level $probL_{l,t}$ and the variations from the nominal price $Var_{l,t}$ for the three considered uncertain levels (low, medium and high price levels).

Table 4. Data for example 2.

Time	Price, $Price_t$	$l1$ = Low price		$l2$ = Medium price		$l3$ =High price	
		$probL_{l,t}$	$Var_{l,t}$	$probL_{l,t}$	$Var_{l,t}$	$probL_{l,t}$	$Var_{l,t}$
$t1$	1.0	0		100	0	0	
$t2$	1.0	11	-0.1	45	0	44	0.1
$t3$	1.0	16	-0.1	50	0	34	0.1
$t4$	1.0	29	-0.1	15	0	56	0.1
$t5$	1.0	36	-0.1	36	0	28	0.1
$t6$	1.0	43	-0.1	19	0	38	0.1
$t7$	1.0	29	-0.1	39	0	32	0.1
$t8$	1.0	19	-0.1	53	0	28	0.1
$t9$	1.0	26	-0.1	49	0	25	0.1
$t10$	1.0	30	-0.1	39	0	31	0.1
$t11$	1.0	40	-0.1	38	0	22	0.1

Also, Table 5 displays the input parameters used to solve the presented mathematical formulation:

Table 5. Input parameter values for example 2.

Parameter	Value
AM	1100
F_t^{max}	400
TH	1004.72
α	0.5
β	-0.5
γ_1	0.255905
γ_2	1.0
δ	0.2
λ	0.944
ρ	0.07
ω	1.0

Finally, the values of $Max\theta M$ and $Max\theta P$ have been pre-established to determine and analyse the conserved nodes constituting the final scenario tree. The selected values are 0.5 and 0.7 for $Max\theta M$ and $Max\theta P$, respectively.

The nomenclature used to formulate this mathematical model is shown below:

Parameter

AM	terminal assets target
F_t^{max}	maximum labour available
$Price_t$	price of goods at time period t
TH	total number of hours in the period
T_t	time period t
$Ufact_t$	utility function factor
α	distribution coefficient
β	elasticity coefficient
γ_1	coefficient 1 in money holding function
γ_2	coefficient 2 in money holding function
δ	income tax rate
λ	discount rate
ρ	one period interest rate
ω	wage rate

Continuous and positive variables

$A_{n,t}$	assets in node n at time t
$B_{n,t}$	time spent on money holdings in node n at time t
$C_{n,t}$	consumption in node n at time t
$F_{n,t}$	labour supplied in node n at time t
$M_{n,t}$	revenues in node n at time t
$S_{n,t}$	saving in node n at time t
$Tax_{n,t}$	net taxes paid in node n at time t
$U_{n,t}$	utilities in node n at time t
$Y_{n,t}$	incomes in node n at time t

Variables

Z	objective variable

4. Results and discussion

The resulting NLP models have been implemented in GAMS 24.7 and solved using CONOPT 3.17. Note that GAMS allows the user to evaluate the sensitivity of a model by evaluating the marginal value of a variable, such as $\theta_{n,t}$.

This section presents the results of the following situations:

- The full stochastic case (non-scenarios techniques applied).
- The stochastic model applying RedOpt, to reduce the number of scenarios.
- The stochastic case applying SCENRED/GAMS.

The proposed examples have been solved considering different time intervals. According to the considered time intervals, the number of scenarios will vary. In other words, the more time intervals, the more scenarios (see Table 1). Therefore, the computational time to reach the optimal solution increases when more time intervals (more scenarios) are taken into account. Thus, different time horizons have been considered to evaluate how the reduction of scenarios considering different time intervals affects the value of the objective function. Table 6 displays the optimal results, computation effort and the number of conserved scenarios for example 1, and these results have been compared with the full stochastic problem. Furthermore, Table 7 includes the model statistics for this example for the different time horizons. As expected, the number of equations and variables when the RedOpt approach (i.e., sensitivity analysis approach) is applied decreases. Furthermore, the model was solved also considering 13 time intervals. However, the model could not be solved due to lack of memory of the computer.

According to the observed results, the quality of the solution has not been highly affected (<2%) by the RedOpt approach for reducing the number of scenarios. This technique has reduced the number of scenarios by approximately 50%. Particularly, cluster technique has been applied for the first two iterations (i.e., time periods $t2$ and $t3$). This is because the sensitivity $\theta M_{n,t}$ of two nodes, $n1$ and $n10$, has associated a low value. However, the objective function may be highly affected, since the sensitivity without considering probability, $\theta P_{n,t}$, has not associated a reduced value compared with the remaining nodes. For the rest of iterations, the removal of nodes is applied, re-assigning probabilities to the remaining nodes. This is because there are nodes with low sensitivity, and the removal can take place without a big impact on the optimal value.

Furthermore, although the RedOpt approach offers a good quality solution, the computational time is bigger for t = 10 and t = 11 when compared with the full stochastic model. This is explained because, for reduced time intervals, the computational time required for solving all iterations is bigger than for solving the full stochastic problem. In other words, this was due to the time spent on analysing the sensitivity and posteriorly removing/clustering nodes at each iteration. Another explanation is that the model is not highly non-linear, what means that the model can be solved faster than highly non-linear models. The second example is highly non-linear, and for all time intervals, the RedOpt approach is faster. This explains why for reduced time intervals, the CPU time required for solving the model is higher when iterating instead of solving the full NLP. Note that no solution was found in 100,000 CPU seconds for both the full stochastic case and for RedOpt when more than 13 time intervals are considered, due to the size of the NLP problem.

Table 6. Optimal solution, computational time and number of conserved scenarios for the example 1 considering different time intervals. The symbol ** denotes that no solution was found.

Time horizon	Approach	Optimal solution (m.u.)		Computational time (CPU, s)		Conserved scenarios	
	Stochastic	743.6		55.6		19,683	
t = 10	RedOpt	743.9	(+0.4%)	255.8		9,841	(49.99%)
	SCENRED	721.3	(-3.9%)	10,000		9,841	(49.99%)
	Stochastic	803.4		650.7		59,049	
t = 11	RedOpt	818.4	(+1.8%)	1,274.6		29,527	(50.00%)
	SCENRED	**		**		**	
	Stochastic	864.7		8,814.7		177,147	
t = 12	RedOpt	849.4	(-1.8%)	5,214.2	(-40.8%)	88,576	(50.01%)
	SCENRED	**		**		**	
	Stochastic	923.5		59,661.3		531,441	
t = 13	RedOpt	906.8	(-1.8%)	36,317.4	(-39.1%)	265,723	(50.00%)
	SCENRED	**		**		**	

Table 7. Model statistics for the example 1.

Time horizon	Stochastic model		RedOpt	
	Equations	Variables	Equations	Variables
t = 10	177,139	206,666	88,705	118,277
t = 11	531,433	620,009	265,867	354,493
t = 12	1,594,315	1,860,038	882,649	1,176,869
t = 13	4,782,961	5,580,125	2,391,661	3,188,885

Moreover, the tool SCENRED and SCENRED2 provided by GAMS has been applied. For 10 time intervals, the model was solved, but spending an unaffordable computational time. For more time intervals, the tool SCENRED was not able to find the optimal solution (i.e., solution not found), which is denoted by ** in Table 6. This is due to a lack of memory, since this tool is not suitable for very large number of scenarios. Thus, SCENRED and SCENRED2 techniques are computationally intractable and not able to be solved when more time intervals are considered.

Figure 6 plots the optimal results and the computational time to solve the proposed problem.

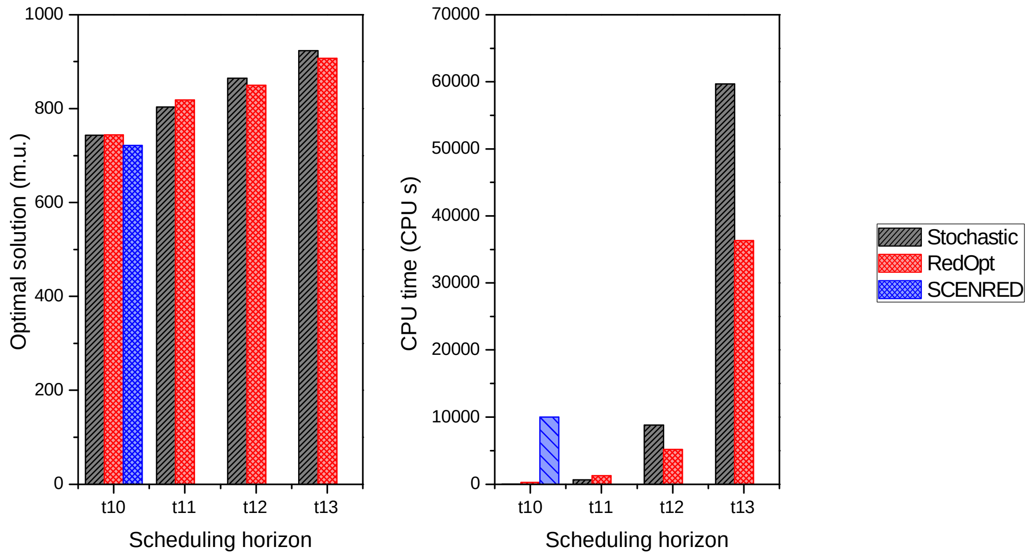

Figure 6. Results for the example 1.

Also, the RedOpt methodology to reduce the number of scenarios has been applied to the second example. Analogously, SCENRED/GAMS has been tested. As in the first example, this tool is computationally intractable and not able to solve the problem. The SCENRED tool can be customised by the user by specifying the number of conserved scenarios. In this case, for a fair comparison of the obtained results through RedOpt and SCENRED, the RedOpt was firstly used. Once the problem was solved, the number of conserved scenarios were determined. Then, this value was inputted in SCENRED as input parameters, in order to analyse the same results using the same number of conserved scenarios.

Table 8 presents the results for the second example. As in the first example, good quality results are obtained, since the value of the objective value is altered by less than 4%. However, the computational time decreases drastically for all considered time horizons. This is explained because the model is highly non-linear, and the reduction of scenarios involves a high reduction in the computational effort to reach the optimal solution.

The reduction of scenarios was carried out by clustering nodes in the second iteration (a node has low probability but affects highly the value of the solution), nodes aggregation (two nodes have the same behaviour and are merged) and nodes removal (a node has low sensitivity and can be removed).

Moreover, Table 9 shows the model statistics for this example, whereas Figure 7 displays the optimal results and the computational time to solve the second illustrative example.

Table 8. Optimal solution, computational time and number of conserved scenarios for the example 2 considering different time intervals. The symbol ** denotes that no solution was found.

Time horizon	Approach	Optimal solution (m.u.)		Computational time (CPU, s)		Conserved scenarios	
$t = 7$	Stochastic	3,691.3		137.4		729	
	RedOpt	3,643.9	(-1.3%)	25.4	(-81.5%)	366	(50.02%)
	SCENRED	3,688.1		39.0		366	(50.02%)
$t = 8$	Stochastic	11,241.1		1,125.5		2,187	

	RedOpt	11,123.7	(-1.1%)	456.0	(-59.5%)	1,095	(50.01%)
	SCENRED	11,241.9		543.2		1,095	(50.01%)
t = 9	Stochastic	32,259.4		6,279.7		6,561	
	RedOpt	31,936.6	(-1.0%)	3,330.4	(-47.0%)	3,281	(50.01%)
	SCENRED	32,257.8		3,897.4		3,281	(50.01%)
t = 10	Stochastic	87,240.6		38,906.0		19,683	
	RedOpt	86,291.3	(-1.1%)	18,822.5	(-51.6%)	9,843	(50.00%)
	SCENRED	87,088.9		22,934.9		9,843	(50.00%)
t = 11	Stochastic	213,611.7		100,057.0		59,049	
	RedOpt	221,892.5	(+3.7%)	47,131.4	(-52.9%)	29,526	(50.00%)
	SCENRED	**		**		**	

Table 9. Model statistics for the example 2.

Time horizon	Stochastic model		RedOpt	
	Equations	Variables	Equations	Variables
t = 7	9,473	9,838	4,815	5,560
t = 8	28,428	29,522	14,305	16,511
t = 9	85,291	88,572	42,749	49,332
t = 10	255,878	265,720	128,055	147,763
t = 11	767,637	797,162	383,947	443,024

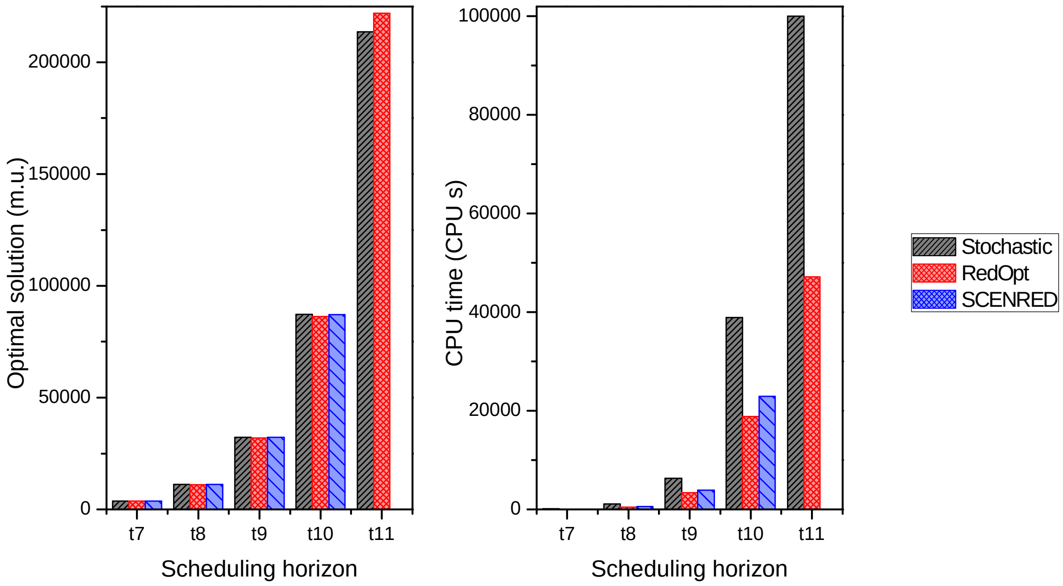

Figure 7. Results for the example 2.

5. Concluding remarks

This paper proposes a sensitivity analysis approach, called RedOpt, for the scenario reduction in two-stage stochastic NLP formulations. Accurate representation of uncertainty may require much larger number of scenarios rendering the resulting optimisation model computationally intractable. Here, the RedOpt approach for the scenario reduction has been used to deal with uncertainty.

The obtained results have demonstrated that this novel procedure can be used efficiently to reduce the number of scenarios while obtaining good quality results with reduced computational requirements. This formulation has been tested in 2 examples, reducing the computational time by 40% and 52% for examples 1 and 2 respectively, while reducing the conserved scenarios by 50% approximately. Moreover, good quality solutions have been obtained by using RedOpt, since the optimal value has been altered only by 2% and 4% for examples 1 and 2 respectively when compared with the stochastic model. Thus, RedOpt is able to reduce the computational time to reach the optimal solution while maintaining valuable quality results.

Further work will focus on applying this methodology to industrial problems, considering different sources of uncertainty, as well as to extend the current approach to Mixed Integer NLP.

Nomenclature

Indexes and sets

$n \in N$	nodes
$l \in L$	uncertain parameter level
$t \in TH$	time intervals to be considered
$pn \in N$	Subset of parent nodes pn
$n \in SN$	Subset of nodes n in the reduced scenario tree SN
$n \in TN_t$	Subset of nodes n included in the time intervals t defining the tree node mapping
$l \in LN_n$	Subset of uncertain levels l in each node n
$tree_{n,pn}$	Subset of nodes n and parent nodes pn defining the scenario tree

Parameters

$Max\theta M$	Maximum relative difference between marginal values of $\theta_{n,t}$ in node n at time t
$Max\theta P$	Maximum relative difference between marginal values of $\theta_{n,t}$ in node n at time t without considering probability
$prob_{n,t}$	Probability of node n in the original scenario tree at time interval t
$sprob_{n,t}$	Probability of node n in the reduced scenario tree at time interval t
$\theta_{n,t}$	Uncertainty parameter in node n at time t
$\theta M_{n,t}$	Marginal value of $\theta_{n,t}$ in node n at time t
$\theta P_{n,t}$	Marginal value of $\theta_{n,t}$ in node n at time t without considering probability

Acknowledgments

The authors would like to thank the financial support received from the UK Engineering and Physical Sciences Research Council (under the project EP/M028240/1).

References

Beraldi, P. & Bruni, M.E. (2014). A clustering approach for scenario tree reduction: an application to a stochastic programming portfolio optimization problem. TOP, 22, 934-949.

Chen, Z. & Yan, Z. (2018). Scenario tree reduction methods through clustering nodes. Computers & Chemical Engineering, 109, 96-111.

de Oliveira, W.L., Sagastizábal, C., Penna, D.D.J., Maceira, M.E.P. & Damázio, J.M. (2010). Optimal scenario tree reduction for stochastic streamflows in power generation planning problems. Optimization Methods and Software, 25, 917-936.

Dua, V. & Pistikopoulos, E.N. (1998). An outer-approximation algorithm for the solution of multi-parametric MINLP problems. Computers & Chemical Engineering, 22, S955-S958.

Dupačová, J., Gröwe-Kuska, N. & Römisch, W. (2003). Scenario reduction in stochastic programming: An approach using probability metrics. Mathematical Programming, Series B, 95, 493-511.

Fair, R.C. (1984). Specification, Estimation, and Analysis of Macroeconomic Models. Harvard University Press, Cambridge.

Fiacco, A.V. & Ishizuka, Y. (1990). Sensitivity and stability analysis for nonlinear programming. Annals of Operations Research, 27, 215-235.

Gal, T. (1975). Rim multiparametric linear programming. Management Science, 21, 567-575.

GAMS library, Household Optimization Problem by Fair problem, (accessed April 2018) https://www.gams.com/latest/gamslib_ml/libhtml/gamslib_hhfair.html

GAMS library, Optimal Pricing and Extraction for OPEC problem, (accessed April 2018). https://www.gams.com/latest/gamslib_ml/libhtml/gamslib_pindyck.html

GAMS/SCENRED Documentation Center, (Accessed April 2018). https://www.gams.com/latest/docs/T_SCENRED.html

GAMS/SCENRED2 Documentation Center, (Accessed April 2018). https://www.gams.com/24.8/docs/tools/scenred2/index.html

Heitsch, H. & Römisch, W. (2003). Scenario reduction algorithms in stochastic programming. Computational Optimization and Applications, 24, 187-206.

Heitsch, H. & Römisch, W. (2007). A note on scenario reduction for two-stage stochastic programs. Operations Research Letters, 35, 731-738.

Heitsch, H. & Römisch, W. (2009). Scenario tree modeling for multistage stochastic programs. Mathematical Programming, 118, 371-406.

Karuppiah, R., Martín, M. & Grossmann, I.E. (2010). A simple heuristic for reducing the number of scenarios in two-stage stochastic programming. Computers & Chemical Engineering, 34, 1246-1255.

Kovacevic, R.M. & Pichler, A. (2015). Tree approximation for discrete time stochastic processes: a process distance approach. Annals of Operations Research, 235, 395-421.

Latorre, J.M., Cerisola, S. & Ramos, A. (2007). Clustering algorithms for scenario tree generation: Application to natural hydro inflows. European Journal of Operational Research, 181, 1339-1353.

Li, Z. & Floudas, C.A. (2014). Optimal scenario reduction framework based on distance of uncertainty distribution and output performance: I. Single reduction via mixed integer linear optimization. Computers & Chemical Engineering, 70, 50-66.

Li, Z. & Floudas, C.A. (2016). Optimal scenario reduction framework based on distance of uncertainty distribution and output performance: II. Sequential reduction. Computers & Chemical Engineering, 84, 599-610.

Li, Z. & Li, Z. (2016). Linear programming-based scenario reduction using transportation distance. Computers & Chemical Engineering, 88, 50-58.

Paulo, H., Cardoso-Grilo, T., Relvas, S. & Barbosa-Póvoa, A.P. (2017). Designing integrated biorefineries supply chain: Combining stochastic programming models with scenario reduction methods. Computer Aided Chemical Engineering, 40, 901-906.

Pindyck, R.S. (1978). Gains to Producers from the Cartelization of Exhaustible Resources. The Review of Economics and Statistics, 60, 238-251.

Pistikopoulos, E.N., Dominguez, L., Panos, C., Kouramas, K. & Chinchuluun, A. (2012). Theoretical and algorithmic advances in multi-parametric programming and control. Computational Management Science, 9, 183-203.

Qian, X. & Tang, Q. (2017). Scenario reduction method based on output performance for condition-based maintenance optimization. Mechanika, 23, 743-749.

Schildbach, G. & Morari, M. (2016). Scenario-based model predictive control for multi-echelon supply chain management. *Energy Journal of Operational Research*, 252, 540-549.

Shapiro, A., Tekaya, W., da Costa, J.P. & Soares, M.P. (2013). Risk neutral and risk averse stochastic dual dynamic programming method. *European Journal of Operational Research*, 224, 375-391.

Silvente, J., Papageorgiou, L.G. & Dua, V. (2017). Optimal management of microgrids under uncertainty using scenario reduction. Computer Aided Chemical Engineering, 2017, 40, 2257-2262.

Silvente, J., Kopanos, G.M., Dua, V. & Papageorgiou. (2018). A rolling horizon approach for optimal management of microgrids under stochastic uncertainty, Chemical Engineering Research and Design, 131, 293-317.