Optimization and Engineering manuscript No.
(will be inserted by the editor)

Iterative Scenario Based Reduction Technique for Stochastic Optimization Using Conditional Value-at-Risk

Raquel García-Bertrand · Roberto Mínguez

Received: date / Accepted: date

Abstract In the last decades, several tools for managing risks in competitive markets, such as the conditional value-at-risk, have been developed. These techniques are applied in stochastic programming models primarily based on scenarios and/or finite sampling, which in case of large-scale models increase considerably their size according to the number of scenarios, sometimes resulting in intractable problems. This shortcoming is solved in the literature using i) scenario reduction methods, and/or ii) speeding up optimization techniques. However, when reducing the number of scenarios, part of the stochastic information is lost. In this paper, an iterative scheme is proposed to get the solution of a stochastic problem representing the stochastic processes via a set of scenarios and/or finite sampling, and modeling risk via conditional value-at-risk. This iterative approach relies on the fact that the solution of a stochastic programming problem optimizing the conditional value-at risk only depends on the scenarios on the upper tail of the loss distribution. Thus, the solution of the stochastic problem is obtained by solving, within an iterative scheme, problems with a reduced number of scenarios (subproblems). This strategy results in an important reduction in the computational burden for large-scale problems, while keeping all the stochastic information embedded in the original set of scenarios. In addition, each subproblem can be solved using speeding-up optimization techniques. The proposed method is very easy to implement and, as numerical results show, the reduction in computing time can be dramatic, and more pronounced as the number of initial scenarios or samples increases.

R. García-Bertrand
Department of Electrical Engineering, Universidad de Castilla-La Mancha, Ciudad Real, Spain
Tel.: +34-926-295300 Ext.: 3889
Fax: +34-926-295361
E-mail: raquel.garcia@uclm.es

R. Mínguez
Environmental Hydraulics Institute "IH Cantabria", Universidad de Cantabria, Cantabria, Spain
Tel.: +34-942-201852
Fax: +34-942-201860
E-mail: roberto.minguez@uclm.es

Keywords Risk management · decision making · scenario reduction · stochastic programming · conditional value-at-risk

Contents

1 Introduction

Stochastic programming (Birge and Louveaux, 1997) is a framework for modeling problems that involve uncertainty. Since real world almost invariably includes uncertain parameters, the discipline of stochastic programming has grown and broadened to cover a wide range of applications. In this stochastic framework different risk measures, such as the Conditional Value-at-Risk (CVaR) (Rockafellar and Uryasev, 2000, 2002; Krokhmal et al, 2002), have been developed to lessen the danger to which the decision maker is exposed because of uncertainty.

Many problems of optimization under uncertainty involve discontinuos distributions, where discrete probabilities come out from scenario models or finite sampling of random variables. This is the type of problem considered in this paper.

The use of CVaR with discrete loss distributions based on scenarios and/or finite sampling, takes advantage of Rockafellar and Uryasev (2002) results, who proposed a scenario-based method to optimize portfolios using linear or nonsmooth programming techniques.

A key step to apply this finite sampling stochastic approach is scenario generation. There exist several approaches to generate sets of scenarios based on different principles. A survey of the most common methods is provided in Dupačová et al (2000). Sampling from historical time series or from statistical models is a popular and easy way to generate data scenarios, see for example Olsson and Soder (2004) or Eichhorn et al (2005). A method based on the target/moment-matching principle can be found in Høyland et al (2003) and Høyland and Wallace (2001). Heitsch and Römisch (2009a) and Pflug (2001) provide a method based on probability metric approximations.

However, the number of scenarios needed to represent the actual stochastic processes can be very large and general-purposes algorithms and software can: i) fail on optimizing the portfolio, ii) being too time consuming, or iii) result in intractable problems, even using the approach proposed by Rockafellar and Uryasev (2002). To solve this shortcoming, two different alternatives exist in the literature: i) scenario reduction techniques, and/or ii) speeding up optimization algorithms.

Scenario reduction techniques concentrate on decreasing the number of scenarios to control model complexity while preserving the degree of approximation. Methods for two-stage stochastic programs can be found in Dupačová et al

(2003) and Heitsch and Römisch (2003). Techniques developed in those two papers were improved in Heitsch and Römisch (2007), extended to mixed-integer models in Henrion et al (2008, 2009), and extended to multistage stochastic programs in Heitsch and Römisch (2009b) and Gröwe-Kuska et al (2003). The idea behind these methodologies is to compute the best approximation of the underlying discrete probability distribution using concepts of stability theory (Heitsch et al, 2006). A scenario reduction technique applied to electricity problems can be found in Pineda and Conejo (2010). Since the CVaR level is based on the tail of the loss distribution, we need to ensure that no information is lost in this portion of the distribution after scenario reduction. However, the main drawback of scenario reduction techniques is the inevitable loss of part of the stochastic information contained in the original set of scenarios, which may alter the CVaR level.

In addition to methods for reducing scenarios, there is another research trend focused on speeding up CVaR-optimization. Techniques based on cutting plane algorithms in a decomposition scheme have been proposed in, e.g., Künzi-Bay and Mayer (2006); Ahmed (2006); Fábián (2008); Fábián and Vesprémi (2008).

This paper proposes a simple and efficient iterative method to solve stochastic programming problems with the following properties:

1. Risk is modeled via the CVaR, where stochastic variables only affect the objective function.
2. They are based on scenarios or finite sampling.

The main characteristics of the iterative method are:

1. The loss of the stochastic information contained in scenarios is avoided.
2. They can be used with speeding-up CVaR optimization algorithms.

The proposed method achieves a significant reduction on the computational time required to solve large-scale stochastic problems, while keeping the stochastic information given by all scenarios. This is particularly advantageous for those problems whose solution may be tedious using existing solvers, or even intractable due to memory limitations.

The methodology relies on the fact that the optimal solution of a stochastic problem optimizing CVaR only depends on scenarios on the upper tail of the loss distribution, which are considered as active scenarios. Since we do not know in advance which scenarios are active at the optimal solution, an iterative approach is presented where the active set is updated iteratively until the optimal solution is attained.

Since in economic, management science and other contexts, when dealing with loss distributions, it is common to consider high confidence levels $\alpha = 0.9, \alpha = 0.95$ or $\alpha = 0.99$, a reduced number of active scenarios representing approximately 10, 5 and 1% of the probability distribution function, respectively, are required to achieve the optimal solution. This is the main reason why numerical simulations show that computational time reduction can be dramatic, and more pronounced as the number of initial scenarios or finite samplings increase.

The rest of the paper is organized as follows. Section 2 presents i) the model structure of the optimal risk-management problem considered in this paper, ii) how to model it using CVaR and the approach proposed by Rockafellar and Uryasev (2002) based on scenarios or finite sampling, and iii) an extension for getting the efficient frontier. Section 3 describes the iterative reduction technique and discusses

convergence issues, which are illustrated in Section 4 with a simple and clarifying example. In contrast, Section 5 provides the results of a realistic case study based on the self-scheduling problem of a power producer, highlighting the goodness of the proposed methodology. Finally, in Section 6 some relevant conclusions are duly drawn. Mathematical formulation of the case study is presented in Appendix A.

2 Optimal risk-management problem

The mathematical structure of the studied stochastic programming model can be defined as follows:

$$\underset{\boldsymbol{x}}{\text{Minimize}}\ q\left(f(\boldsymbol{x},\boldsymbol{y})\right), \tag{2.1}$$

subject to

$$\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0} \tag{2.2}$$

$$\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}, \tag{2.3}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the vector including the decision variables, $\boldsymbol{y} \in \mathbb{R}^m$ is the stochastic parameter vector including all random parameters involved, $f(\boldsymbol{x},\boldsymbol{y})$ is the objective function representing losses, and $\boldsymbol{h}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{x})$ represent equality and inequality constraints that must be satisfied. Note that $f(\boldsymbol{x},\boldsymbol{y})$, $\boldsymbol{h}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{x})$ might be nonlinear functions and $\boldsymbol{x}$ might include integer variables.

Due to the stochastic nature of $\boldsymbol{y}$, the objective function $f(\boldsymbol{x},\boldsymbol{y})$ is a random variable whose probability distribution function depends on the probability distribution functions of the random parameters $\boldsymbol{y}$, and the values of the decision variables $\boldsymbol{x}$. And since a random variable cannot be minimized, the objective function is replaced by a risk measure given by the functional $q\left(f(\boldsymbol{x},\boldsymbol{y})\right)$.

Note that in this paper we only consider stochastic variables affecting the objective function. The applicability of the proposed method to stochastic variables affecting constraints is a subject for further research.

2.1 Conditional Value-at-Risk (CVaR) based on finite sampling

A possible risk measure widely used in current regulations for finance business is the percentile of the loss distribution. An example of this kind of measures is the Value-at-Risk at the α (0.9, 0.95 or 0.99) confidence level, so called α-VaR. An alternative to VaR, also included within the type of percentile risk measures (Krokhmal et al (2002)), is the Conditional Value-at-Risk (CVaR), which quantifies the expected losses associated with the upper tail of the loss distribution. CVaR has gained popularity in real-life applications, specially in financial and economic decision making problems (see e.g. Conejo et al (2008), Jabr (2005), Lim et al (2010) or Mínguez et al (2011)), because it preserves linear programming solvability and overcomes some of the drawbacks of other possible risk measures, such as VaR. CVaR constitutes a measure of risk with significant advantages over VaR being a coherent risk measure in the sense of Artzner et al (1999), and taking into account the extremely high losses that may occur, albeit at low probabilities, in the tail of the loss distribution.

The advantage of using CVaR is the possibility of using the formula derived by Rockafellar and Uryasev (2002), which allows the characterization of α-VaR ($\eta_\alpha(\boldsymbol{x})$) and α-CVaR ($\phi_\alpha(\boldsymbol{x})$) for models based on scenarios and finite sampling. Thus, problem (2.1)–(2.3) replacing the objective function by using the CVaR concept becomes:

$$\underset{\boldsymbol{x},\,\eta_\alpha}{\text{Minimize}}\ \eta_\alpha + \frac{1}{1-\alpha}\mathrm{E}\left\{[f(\boldsymbol{x},\boldsymbol{y})-\eta_\alpha]^+\right\}, \tag{2.4}$$

subject to (2.2)–(2.3), where $E\{\cdot\}$ is the expected value operator, $[a]^+ = a$ for $a > 0$ and $[a]^+ = 0$ for $a \leq 0$. For the particular case in which discrete probabilities p_i are associated with scenario models $\boldsymbol{y}_i$; $\forall i \in \mathcal{N}$, equation (2.4) becomes (Rockafellar and Uryasev, 2000, 2002):

$$\underset{\boldsymbol{x},\,\eta_\alpha}{\text{Minimize}}\ \eta_\alpha + \frac{1}{1-\alpha}\sum_{\forall i\in\mathcal{N}} p_i\left[f(\boldsymbol{x},\boldsymbol{y}_i)-\eta_\alpha\right]^+. \tag{2.5}$$

Since the probability is concentrated in finitely many points $\boldsymbol{y}_i$, the distribution of the loss $z_i = f(\boldsymbol{x},\boldsymbol{y}_i)$ for a given $\boldsymbol{x}$ is likewise concentrated in finitely many points. Assuming that those losses are ordered as $z_1 < z_2 < z_3 < \ldots < z_N$, where N is the cardinality of set $\mathcal{N}$, the α-VaR ($\eta_\alpha(\boldsymbol{x})$) of the loss is given by z_{i_α} such that the index position i_α holds the following condition:

$$\sum_{i=1}^{i_\alpha-1} p_i < \alpha \leq \sum_{i=1}^{i_\alpha} p_i. \tag{2.6}$$

Proof of this statement is given in Rockafellar and Uryasev (2002).

Finally, the linearization formula proposed by Rockafellar and Uryasev (2002) remains valid, and the original problem (2.1)–(2.3) is approximated as follows:

$$\underset{\boldsymbol{x},\,\eta_\alpha,\,u_i}{\text{Minimize}}\ \eta_\alpha + \frac{1}{(1-\alpha)}\sum_{\forall i\in\mathcal{N}} p_i u_i\,, \tag{2.7}$$

subject to

$$\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0} \tag{2.8}$$

$$\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0} \tag{2.9}$$

$$u_i \geq f(\boldsymbol{x},\boldsymbol{y}_i) - \eta_\alpha; \quad \forall i \in \mathcal{N} \tag{2.10}$$

$$u_i \geq 0; \quad \forall i \in \mathcal{N}, \tag{2.11}$$

where u_i; $\forall i \in \mathcal{N}$ are auxiliary variables. Note that problem (2.7)–(2.11) is linear assuming that i) constraints (2.8)–(2.9) are linear, ii) function $f(\boldsymbol{x},\boldsymbol{y}_i)$; $\forall i \in \mathcal{N}$ is linear with respect to $\boldsymbol{x}$, and iii) $\boldsymbol{x} \in \mathbb{R}^m$.

The computational advantages shown by the linearization formula proposed by Rockafellar and Uryasev (2002) and the simplicity on its implementation has tempted practitioners to apply formulation (2.7)–(2.11) to more general problems rather than linear programming formulations. The drawback is that it is not clear how it is affected by non-convexities and/or non-linearities of $\boldsymbol{g}$ and/or $\boldsymbol{h}$, or integrality of $\boldsymbol{x}$. Although this problem is relevant and may constitute a subject for specific research, it is out of the scope of this paper. The aim of this paper is to present an alternative solution approach for problem (2.7)–(2.11), and not to set under what conditions it converges.

Remark 2.1 The iterative method proposed in this paper to solve problem (2.7)–(2.11) requires this problem to converge to a unique solution. If this is not the case, the iterative algorithm may converge to a different solution or may not converge, and it becomes an heuristic procedure.

2.2 Different formulations for the efficient frontier

Dealing with CVaR for solving stochastic programming problems has the additional advantage that it can be used as a risk metric for counter-balance the expected profit and risk as follows:

1. Minimizing some function of expected profit and risk.
2. Minimizing risk conditional on a minimum expected profit.
3. Minimizing the negative expected profit including a constraint on the maximum allowed risk.

Krokhmal et al (2002) proved that under certain regularity conditions, the three problems above are equivalent and generate the same efficient frontier. For this reason, and in order to extend the applicability of the proposed methodology to these problems, an extension of the proposed approach is given to solve the following stochastic optimization problem:

$$\underset{\boldsymbol{x}}{\text{Minimize}}\ \phi_{\alpha}(\boldsymbol{x}) - \delta \mathrm{E}\{p_r(\boldsymbol{x},\boldsymbol{y})\}\,, \tag{2.12}$$

subject to

$$\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0} \tag{2.13}$$

$$\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}, \tag{2.14}$$

where $\phi_{\alpha}(\boldsymbol{x})$ corresponds to the objective function in equation (2.7), $p_r(\boldsymbol{x},\boldsymbol{y})$ is a function representing profit, and $\delta > 0$ is a positive constant to counter balance the negative expected profit and the risk using α-CVaR. Note that the approach given for problem (2.12)–(2.14) could be easily extended for solving any of the three stochastic programming problems cited above.

The main reason to also take into consideration this problem structure is the interest shown by practitioners, especially on the power system economics community, which use the objective function given in (2.12) to make decisions bearing in mind profit versus risk (see, for example, Conejo et al (2008); Carrión et al (2009); Zhang and Wang (2009); Hatami et al (2011)). One concern for that community is the large amount of scenarios and variables required to pose realistic case studies, thus requiring techniques to speeding up and/or making tractable the solution of their problems. This fact persuades us to extend the possible application of the iterative approach presented in this paper to solve problem (2.12)–(2.14).

3 Iterative Reduction Technique

Let consider the problem (2.7)–(2.11) and assume that the optimal solution is unique (according to remark 2.1) and corresponds to $\boldsymbol{x}^*$ and η_{α}^*. With this information, given a scenarios set $\mathcal{N}$, scenarios can be classified into two different subsets:

- **Active scenarios (AS):** Scenarios placed to the right-hand side of η_α^*, that is,

$$z_i = f(\boldsymbol{x}, \boldsymbol{y}_i) \geq \eta_\alpha^*; \quad \forall i \in \mathcal{N}_{\text{AS}}, \tag{3.1}$$

  where $\mathcal{N}_{\text{AS}} \subset \mathcal{N}$ is the subset of active scenarios.
- **Inactive scenarios (IS):** Scenarios placed to the left-hand side of η_α^*, that is,

$$z_i = f(\boldsymbol{x}, \boldsymbol{y}_i) < \eta_\alpha^*; \quad \forall i \in \mathcal{N}_{\text{IS}}, \tag{3.2}$$

where $|\mathcal{N}_{\text{AS}}| + |\mathcal{N}_{\text{IS}}| = N$, and according to expression (2.6) the number of active scenarios is equal to $i_\beta = N - i_\alpha + 1$. Note that i_β is the index position associated with the α-VaR (η_α) but starting counting scenarios from higher to lower losses. Condition (2.6), using this new index, becomes:

$$\sum_{i=1}^{i_\beta - 1} p_i \leq 1 - \alpha < \sum_{i=1}^{i_\beta} p_i. \tag{3.3}$$

We advocate the use of this expression for computational convenience.

Note also that auxiliary variables u_i are equal to zero if the loss of scenario i ($z_i = f(\boldsymbol{x}, \boldsymbol{y}_i)$) is lower than η_α, and otherwise are equal to the difference between the loss of the corresponding scenario ($z_i = f(\boldsymbol{x}, \boldsymbol{y}_i)$) and η_α. This is represented in Figure 3.1.

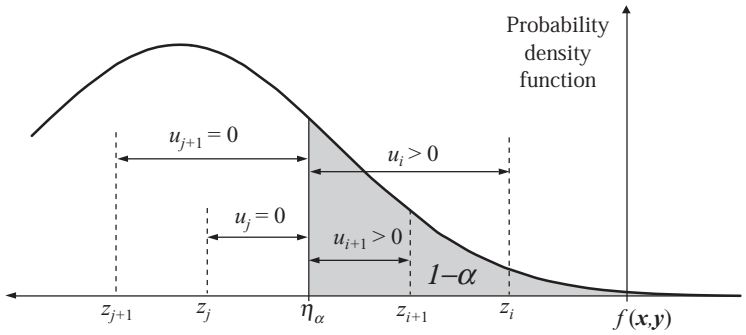

Fig. 3.1 Graphical representation of auxiliary variable u_i.

Proposition 3.1 *Given a set of scenarios $\mathcal{N} \equiv \{\boldsymbol{y}_i;\ i = 1, \ldots, N\}$, and assuming that the active set $\mathcal{N}_{\text{AS}}$ from the optimal solution of problem (2.7)–(2.11) is known, the optimal solution of the problem:*

$$\underset{\boldsymbol{x}, \eta_\alpha, u_i}{\text{Minimize}}\ \eta_\alpha + \frac{1}{(1-\alpha)} \sum_{\forall i \in \mathcal{N}_{\text{AS}}} p_i u_i\,, \tag{3.4}$$

subject to

$$\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0} \tag{3.5}$$

$$\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0} \tag{3.6}$$

$$u_i \geq f(\boldsymbol{x}, \boldsymbol{y}_i) - \eta_\alpha; \quad \forall i \in \mathcal{N}_{\text{AS}} \tag{3.7}$$

$$u_i \geq 0; \quad \forall i \in \mathcal{N}_{\text{AS}}, \tag{3.8}$$

coincides with the optimal solution of problem (2.7)–(2.11).

Proof Both problems have the same constraints, because inactive constraints $u_i \geq f(\boldsymbol{x}, \boldsymbol{y}_i) - \eta_\alpha;\ \forall i \in \mathcal{N}_{\text{IS}}$ may be removed from (2.10). In addition, the objective function (2.7) does not change because $u_i = 0;\ \forall i \in \mathcal{N}_{\text{IS}}$. Therefore the optimal solutions of both problems coincide.

The advantage of using (3.4)–(3.8) instead of (2.7)–(2.11) is that the size of the problem in terms of number of variables and constraints becomes smaller. Assuming equally likely scenarios, i.e. $p_i = 1/N;\ \forall i \in \mathcal{N}$, and according to (3.3), the upper bound of the maximum number of active scenarios is:

$$i_\beta < (1-\alpha)N + 1, \tag{3.9}$$

which implies that the higher the confidence level α, the lower the number of active scenarios. This condition may represent an important reduction in terms of variables, constraints, and complexity.

We can conclude that if the set of active scenarios were known in advance, the decision-making problem could be solved reducing the number of variables and constraints of the program considerably, which would result in a drastic reduction in the computational burden. However, we do not know in advance the active scenario set at the optimal solution. An iterative approach is presented to solve this shortcoming.

3.1 Sequential procedure

The functioning of the proposed algorithm is based the following sequential procedure (for a given iteration ν):

1. Given subset $\mathcal{N}_{\text{AS}}^{(\nu)}$ taken from the initial scenario set $\mathcal{N}$ with minimum cardinality, and so that
$$\sum_{\forall i \in \mathcal{N}_{\text{AS}}^{(\nu)}} p_i \geq b(1-\alpha), \text{ and } |\mathcal{N}_{\text{AS}}^{(\nu)}| = N^{(\nu)} \leq N, \tag{3.10}$$
where $b > 1$ is a given constant.
2. Solve subproblem (3.4)–(3.8) using $\mathcal{N}_{\text{AS}}^{(\nu)}$, obtaining the optimal solution for the given subset, i.e, $\boldsymbol{x}^{(\nu)}$, $\eta_\alpha^{(\nu)}$.
3. If losses for each scenario from subset $\mathcal{N}_{\text{AS}}^{(\nu)}$ are ordered as $z_1^{(\nu)} < z_2^{(\nu)} < z_3^{(\nu)} < \ldots < z_{N^{(\nu)}}^{(\nu)}$, then $\eta_\alpha^{(\nu)}$ corresponds to $z_{i_\alpha^{(\nu)}}^{(\nu)}$, which is associated with the index position $i_\alpha^{(\nu)}$ holding condition (2.6). In addition, the index related to (3.3) corresponds to $i_\beta^{(\nu)} = N^{(\nu)} - i_\alpha^{(\nu)} + 1$.
4. Given $\boldsymbol{x}^{(\nu)}$, the ordered losses from set $\mathcal{N}$ are $z_1^{(\nu')} < z_2^{(\nu')} < z_3^{(\nu')} < \ldots < z_N^{(\nu')}$. The loss value $z_{i_\alpha^{(\nu)}}^{(\nu)}$ corresponds to position $i_\alpha^{(\nu')}$ within this new ordered list. According to (2.6) the true confidence level γ is between the following bounds:
$$\sum_{i=1}^{i_\alpha^{(\nu')}-1} p_i < \gamma \leq \sum_{i=1}^{i_\alpha^{(\nu')}} p_i. \tag{3.11}$$

Analogously, $i_{\beta}^{(\nu')} = N - i_{\alpha}^{(\nu')} + 1$, and according to (3.3) the true significance level $(1-\gamma)$ is between the following bounds:

$$\sum_{i=1}^{i_{\beta}^{(\nu')}-1} p_i \leq 1-\gamma < \sum_{i=1}^{i_{\beta}^{(\nu')}} p_i. \tag{3.12}$$

3.2 Convergence issues

An advantage of the proposed sequential procedure is that it is possible to detect, if the optimal solution of the original problem (2.7)–(2.11) has been achieved. This is based on the following proposition.

Proposition 3.2 *Let consider $\boldsymbol{x}^{(j)}$ as the j-th subproblem solution within the iterative process, which is based on subset $\mathcal{N}_{\mathrm{AS}}^{(j)}$. The value-at-risk $\eta_{\alpha}^{(j)}$ is associated with the $i_{\beta}^{(j)}$ position from the ordered (descending order) loss subsample list and holding condition (3.3). The solution $\boldsymbol{x}^{(j)}$, $\eta_{\alpha}^{(j)}$ corresponds to the optimal solution of the original problem (2.7)–(2.11) iff the position related to $\eta_{\alpha}^{(j)}$ in the ordered (descending order) sample list considering all scenarios, i.e. $i_{\beta}^{(j')}$, remains unaltered, i.e. $i_{\beta}^{(j')} = i_{\beta}^{(j)}$.*

Proof According to (3.3), the solution of subproblem (3.4)–(3.8) considering $\mathcal{N}_{\mathrm{AS}}^{(j)}$ holds the following condition:

$$\sum_{i=1}^{i_{\beta}^{(j)}-1} p_i \leq 1-\alpha < \sum_{i=1}^{i_{\beta}^{(j)}} p_i. \tag{3.13}$$

If considering all scenarios, the position related to $\eta_{\alpha}^{(j)}$ within the loss ordered (descending order) list remains unchanged, condition (3.12) also remains valid and so that $\gamma = \alpha$. Thus, according to (3.11):

$$\sum_{i=1}^{i_{\alpha}^{(j')}-1} p_i < \alpha \leq \sum_{i=1}^{i_{\alpha}^{(j')}} p_i. \tag{3.14}$$

Note that (3.14) coincides with expression (2.6), which is the condition of problem (2.7)–(2.11) fulfilled at the optimum, *l.q.q.d.*

Another important issue is the proof of convergence of the iterative method.

Proposition 3.3 *Considering that problem (2.7)–(2.11) converges to a unique solution, convergence of the sequential procedure explained in the previous section can be achieved using the following rules of thumb:*

1. *Constant b satisfies $b > 1$ because otherwise subproblems are unbounded, i.e. $\eta_{\alpha} \to \infty$.*
2. *The smaller is the b-value, the smaller is the number of scenarios, which in turn reduces the complexity of the subproblems and increases computational efficiency. However, the b-value must be large enough so that the true significance bounds given by (3.12) decrease between consecutive iterations, otherwise the value of the parameter b must be increased.*

Proof Note that according to the proof of proposition 3.2, the optimal solution of the original problem is achieved if significance level $(1-\gamma)$ in (3.12) decreases until it equals to $1-\alpha$. Since the b-value is increased whenever the significant level does not decrease, in the worst case scenario the b-value would increase until reaching its maximum posible value $b^{\max} = 1/(1-\alpha)$. This case considers all scenarios, which under assumption that problem (2.7)–(2.11) converges to a unique solution, it ensures convergence, *l.q.q.d.*

The drawback of the proposed procedure is that we have not found proof of convergence of the iterative process in terms of what is the appropriate b-value selection. However, this is not a problem from the practical point of view because we propose a heuristic procedure for b-value selection and updating. In addition, the method does not ensure reducing computational time, especially for small-size problems, however, numerical simulations indicate that the heuristic procedure achieves convergence with an important reduction on model complexity and computational time for large-size problems.

Note that the main idea of the algorithm is to look for, within an iterative scheme, the region in the random variable subspace defined by scenarios and decision variables, related to the upper tail of the loss distribution. For this reason, in case the bounds (3.12) do not decrease between consecutive iterations, it means that there is no enough information embedded in the selected subset to focus on the area of interest.

3.3 Iterative algorithm

In this section, the iterative algorithm including the heuristic procedure for b-value selection and updating is presented. The corresponding algorithm proceeds as follows:

Algorithm 3.1 *(Iterative reduction method).*

Input: Target confidence level α, set $\mathcal{N}$ of scenarios, objective function and constraints.

Step 1: Initialization. Set parameter $b = 2$. Initialize the iteration counter $\nu = 1$, set the reference significance level to $\mathrm{CL}^{(\nu)} = \infty$ and select randomly an scenario subset $\mathcal{N}_{\mathrm{AS}}^{(\nu)} \subset \mathcal{N}$ holding (3.10).

Step 2: Sequential procedure At each iteration, repeat the sequential procedure given in subsection 3.1.

Step 3: Convergence checking. If $i_\beta^{(\nu')} = i_\beta^{(\nu)}$ the optimal solution of the original problem has been found, go to Step 5, otherwise check if the true upper confidence bound decreases with respect to $\mathrm{CL}^{(\nu)}$ between consecutive iterations, i.e. $\sum_{i=1}^{i_\beta^{(\nu)'}} p_i < \mathrm{CL}^{(\nu)}$. If it decreases then update the reference significance bound $\mathrm{CL}^{(\nu+1)} = \sum_{i=1}^{i_\beta^{(\nu')}} p_i$ and go to Step 4, otherwise increase b-value $(b = b + \Delta b)$ and continue in Step 4.
Note that a Δb value of 0.5 has been shown to be appropriate for practical cases.

Step 4: Update the subset of active scenarios. Given the ordered losses list considering all scenarios, update iteration counter $\nu \rightarrow \nu + 1$ and select as new subset $\mathcal{N}_{\mathrm{AS}}^{(\nu)}$ the minimum number of scenarios on the right tail of the loss distribution so that $\sum_{\forall i \in \mathcal{N}_{\mathrm{AS}}^{(\nu)}} p_i \geq b(1-\alpha)$. Continue in Step 2.

Step 5: Output. The optimal decision variable values correspond to $\boldsymbol{x}^{(\boldsymbol{\nu})}$.

□

Figure 3.2 shows the flow chart of this algorithm.

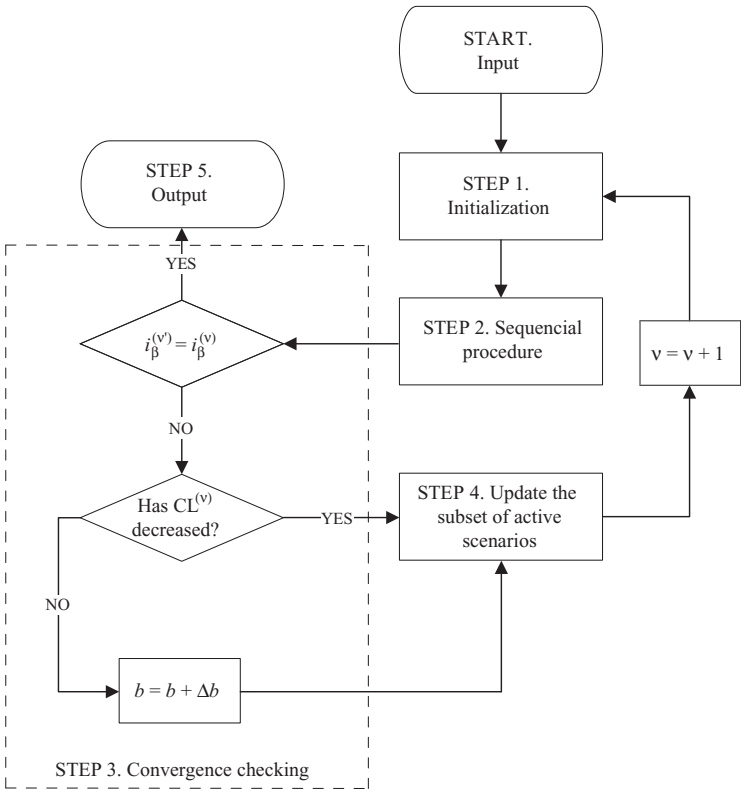

Fig. 3.2 Flow chart of the algorithm.

Note that depending on the problem structure, the calculation of the objective function for each scenario, after subproblem solution $\boldsymbol{x}^{(\nu)}$ is obtained, may require different solution approaches, e.g.:

- One-stage stochastic programming problems: They require the evaluation of a given function.
- Two-stage stochastic programming problems: They require solving an optimization problem involving only the considered scenario and assuming that first-stage decision variables are fixed to the optimal values $\boldsymbol{x}^{(\nu)}$ obtained at the corresponding iteration.

3.4 Application to different stochastic programming formulations

So far, the development of the iterative procedure has been applied on stochastic programming problems focused on the minimization of the CVaR. However, to counter-balance risk versus profit, an extension of the algorithm is presented. The aim is to solve problem (2.12)–(2.14), where the objective function is equal to:

$$\phi_\alpha(\boldsymbol{x}) - \delta \mathrm{E}\{p_r(\boldsymbol{x}, \boldsymbol{y})\}. \tag{3.15}$$

Note that this objective function constitutes a linear combination between CVaR and expected profit. Parameter $\delta > 0$ is a weighting constant controlling risk positioning.

The main concern for solving problem (2.12)–(2.14) within the proposed iterative scheme is that the expected profit calculation requires all scenarios, which does not allow the application of the reduction technique. For this reason, in this paper we propose to solve the following approximation of the original problem (2.12)–(2.14):

$$\underset{\boldsymbol{x}}{\text{Minimize}}\ \phi_\alpha(\boldsymbol{x}) - \delta p_r(\boldsymbol{x}, \bar{\boldsymbol{y}})\,, \tag{3.16}$$

subject to (2.13)–(2.14). Note that the expected value of the profit has been replaced by a central value $p_r(\boldsymbol{x}, \bar{\boldsymbol{y}})$ of the profit distribution, substituting the random variables with their expected values ($\bar{\boldsymbol{y}}$). This approximation is based on a Multivariate Taylor series of the profit function around the point $p_r(\boldsymbol{x}, \bar{\boldsymbol{y}})$:

$$p_r(\boldsymbol{x}, \boldsymbol{y}) = p_r(\boldsymbol{x}, \bar{\boldsymbol{y}}) + \sum_{i=1}^{\infty} \frac{1}{i!} \left. \frac{\partial^i p_r(\boldsymbol{x}, \boldsymbol{y})}{\partial \boldsymbol{y}^i} \right|_{\bar{\boldsymbol{y}}} (\boldsymbol{y} - \bar{\boldsymbol{y}})^i, \tag{3.17}$$

and it corresponds to a first order approximation. Thus, this central value coincides with the expected value of (3.17) if the objective function depends linearly on $\boldsymbol{y}$, otherwise it is an approximation. The goodness of this approach strongly depends on the type of profit function, and it has to be established for each particular case. Nonetheless, once its degree of approximation is established, it may be acceptable for practical purposes. Especially since it may represent an important reduction on computational time for large problems while keeping an acceptable degree of accuracy. Note also that, once the optimal decision variables $\boldsymbol{x}^*$ are known, the difference between $p_r(\boldsymbol{x}, \bar{\boldsymbol{y}})$ and $\mathrm{E}\{p_r(\boldsymbol{x}, \boldsymbol{y})\}$ can be easily calculated to check whether it is acceptable or not.

Furthermore, using Jensen inequality, it is known that if profit function p_r is convex with respect to y then $\phi_\alpha(\boldsymbol{x}) - \delta p_r(\boldsymbol{x}, \bar{\boldsymbol{y}}) \geq \phi_\alpha(\boldsymbol{x}) - \delta \mathrm{E}\{p_r(\boldsymbol{x}, \boldsymbol{y})\}$.

4 Illustrative Example

In order to illustrate the functioning of the method and the graphical interpretation of the iterative process, a simple example with only two decision variables is presented below.

Consider an energy producer seeking to establish a market bidding strategy for a two-hour time horizon, where prices at every hour are uncorrelated normal random variables ($\lambda_t \sim N(\mu_{\lambda_t}, \sigma^2_{\lambda_t})$) with the following distributions $\lambda_1 \sim N(14, 8^2)$ \$/MW and $\lambda_2 \sim N(7, 1^2)$ \$/MW, respectively.

The producer decides to minimize his losses using the CVaR for a $\alpha = 0.95$ confidence level, i.e. minimize $\phi_{0.95}$, as follows:

$$\underset{x_1, x_2}{\text{minimize}}\ \phi_{0.95} = q(-\lambda_1 x_1 - \lambda_2 x_2)\,, \tag{4.1}$$

subject to

$$x_1 + x_2 = 1, \tag{4.2}$$

where x_t is the production at hour t, and $q(\cdot)$ represents α-CVaR. The optimal solution of problem (4.1)–(4.2) is $x_1^* = 0.0725$ MW, $x_2^* = 0.927$ MW, $\eta_{0.95}^* = -\$5.71$ and $\phi_{0.95}^* = -\$5.25$.

Assuming that the discrete approach is used, we sample from the corresponding probability distribution functions 1000 scenarios of prices, and use Rockafellar and Uryasev (2002) formulation given by (2.7)–(2.11). This one-stage stochastic programming problem is formulated as the linear programming problem shown below:

$$\underset{x_1, x_2}{\text{minimize}}\ \eta_\alpha + \frac{1}{(1-0.95)1000} \sum_{i=1}^{1000} u_i\,, \tag{4.3}$$

subject to

$$x_1 + x_2 = 1 \tag{4.4}$$

$$u_i \geq -(\lambda_{1i} x_1 + \lambda_{2i} x_2) - \eta_\alpha; \quad i = 1, \ldots, 1000 \tag{4.5}$$

$$u_i \geq 0; \quad i = 1, \ldots, 1000, \tag{4.6}$$

where λ_{1i} and λ_{2i} represent, respectively, the prices λ_1 and λ_2 for scenario i.

The optimal solution of problem (4.3)–(4.6) with one particular sample is $x_1^* = 0.061$ MW, $x_2^* = 0.939$ MW, $\eta_{0.95} = -\$5.694$, and $\phi_{0.95} = -\$5.293$. The solution is depicted in Figure 4.1, where scenarios whose losses are lower than $\eta_{0.95}$ (inactive scenarios) are depicted in medium grey color, and those whose losses are upper or equal (active scenarios) in light grey color. Conditions (2.6) and (3.3) at the optimum become:

$$0.949 < 0.95 \leq 0.951; \text{ and } 0.049 \leq 0.05 < 0.051. \tag{4.7}$$

Note that the lower and upper significance bounds are inconsistent assuming equally likely scenarios. However, the problem uses equally likely scenarios associated with the input random variables $\boldsymbol{y}$, but the optimal solution has two losses of the same quantity and equal to $\eta_{0.95}$. Both are associated with strictly active scenarios shown in Figure 4.1 (light gray circles with black contours). Note that strictly active scenarios are active scenarios satisfying inequality (3.1) as an equality. These losses merge into one unique loss-scenario with probability 0.002, which explains (4.7).

The above problem has also been solved using the iterative reduction technique presented in Section 3 with a b-value equal to 2 and $\Delta b = 0.5$. The algorithm converges to the optimal solution in 6 iterations. Table 4.1 provides the evolution of the production variables (x_1, x_2), the value-at-risk (η_α), the conditional value-at-risk (ϕ_α), the reference significance level (CL), the lower and upper bounds of 1 minus the true confidence level $((1-\alpha)_{\text{lo}}, (1-\alpha)_{\text{up}})$ for the subproblem, the

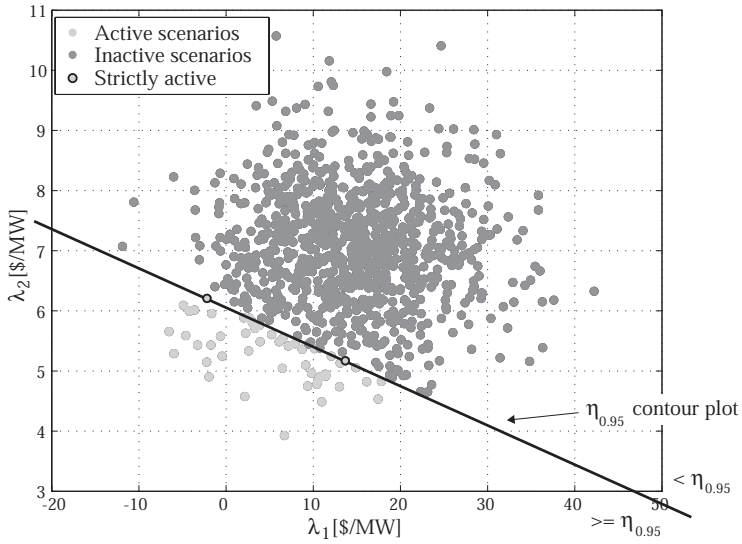

Fig. 4.1 Graphical illustration of the solution of the illustrative example considering 1000 scenarios.

lower and upper bounds of 1 minus the true significance level $((1-\gamma)_{\text{lo}}, (1-\gamma)_{\text{up}})$, the index position related to the value-at risk from the ordered loss subsample list $(i_\beta^{(\nu)})$, the index position from the ordered loss list considering all scenarios $(i_\beta^{(\nu')})$, and the b-value. Note that 1 minus the true confidence level upper bound increases at iteration 3 and thus the b-value is increased to 2.5. At iteration 4, 1 minus the true confidence level upper bound decreases with respect to the previous iteration, but this value is still higher with respect to the reference confidence level (CL) obtained at iteration 2. Therefore, the b-value is increased again at iteration 4 in order to accelerate convergence. Note also that the optimal solution obtained by the iterative algorithm corresponds to the solution of the problem considering all scenarios. The evolution of the algorithm is illustrated in Figure 4.2. In this figure, scenarios whose losses are lower than $\eta_{0.95}^{(\nu)}$ (inactive scenarios) are depicted in medium grey color, those scenarios whose losses are upper or equal to $\eta_{0.95}^{(\nu)}$ (active scenarios) are depicted in light grey color, and those scenarios belonging to the scenario subset at iteration ν are depicted in black color.

Besides the 1000 scenario case, this example has also been solved for an increasing number of scenarios and considering a confidence level α of 0.95. For the iterative algorithm, b is equal to 2, and problems considering 1000, 10000, 20000, 50000 and 100000 scenarios converge in 6, 6, 10, 12 and 10 iterations, respectively. We have solved 100 simulation cases for each problem to get statistically sound conclusions. Table 4.2 provides the mean and standard deviation value of computational time required the get the solution considering all scenarios (second and third columns) and using the iterative algorithm (fourth and fifth columns). Note that reported times in the example includes the building and the resolution of the model. For the computational simulations, we have used simplex algorithm

ν	x_1 [MW]	x_2 [MW]	η_α [$]	ϕ_α [$]	CL	$(1-\alpha)_{\text{lo}}$	$(1-\alpha)_{\text{up}}$	$(1-\gamma)_{\text{lo}}$	$(1-\gamma)_{\text{up}}$	$i_\beta^{(\nu)}$	$i_\beta^{(\nu')}$	b
1	0.194	0.806	-8.386	-6.552	Inf	0.502	0.504	0.049	0.051	50	503	2
2	0.000	1.000	-6.164	-5.528	0.504	0.199	0.200	0.050	0.051	51	200	2
3	1.000	0.000	-14.380	-7.002	0.200	0.502	0.503	0.049	0.051	50	503	2
4	0.000	1.000	-6.531	-5.944	0.200	0.318	0.319	0.050	0.051	51	319	2.5
5	0.095	0.905	-5.898	-5.317	0.200	0.065	0.067	0.049	0.051	50	66	3
6	0.061	0.939	-5.694	-5.293	0.067	0.049	0.051	0.049	0.051	50	50	3

Table 4.1 Evolution of the iterative reduction algorithm for the illustrative example.

(Matlab function "linprog") on a Linux-based server with one processor clocking at 2.3 GHz and 8 GB of RAM.

Scenarios	All scenarios		Iterative Alg.	
	μ	σ	μ	σ
1000	0.19	0.003	0.23	0.083
10000	22.72	0.04	8.48	1.95
20000	70.60	0.06	34.60	6.51
50000	789.55	0.66	272.12	57.75
100000	8575.30	100.40	1187.02	343.86

Table 4.2 Computational time [s] for different number of scenarios in the illustrative example.

From results given in Table 4.2 the following observations are pertinent:

1. The iterative algorithm improves efficiency with respect to the approach using all scenarios above a minimum number of scenarios, which for this particular case is around 10000. The efficiency increases as the number of scenarios also increases.
2. Although both methods attain the same solution, the iterative approach does not guarantee the reduction on computational time with respect to the method considering all scenarios. This is the case for the 1000 scenarios problem, where the iterative approach is slower.

5 Self-Scheduling Case Study

In order to show the performance of the method using a realistic case study, let consider the self-scheduling problem of a power producer.

Electric energy can be traded in a pool market and through bilateral contracts. The pool consists of a day-ahead market while the bilateral contracts allows trading electricity up to one year ahead. The bilateral contracts present a lower average price than the pool but involve reduced volatility. Thus, it allows hedging against the financial risk inherent in pool price volatility (see Conejo et al, 2008).

A power producer needs to define its involvement in both the pool market and through bilateral contracts so that its loss is minimized or, equivalently, its profit

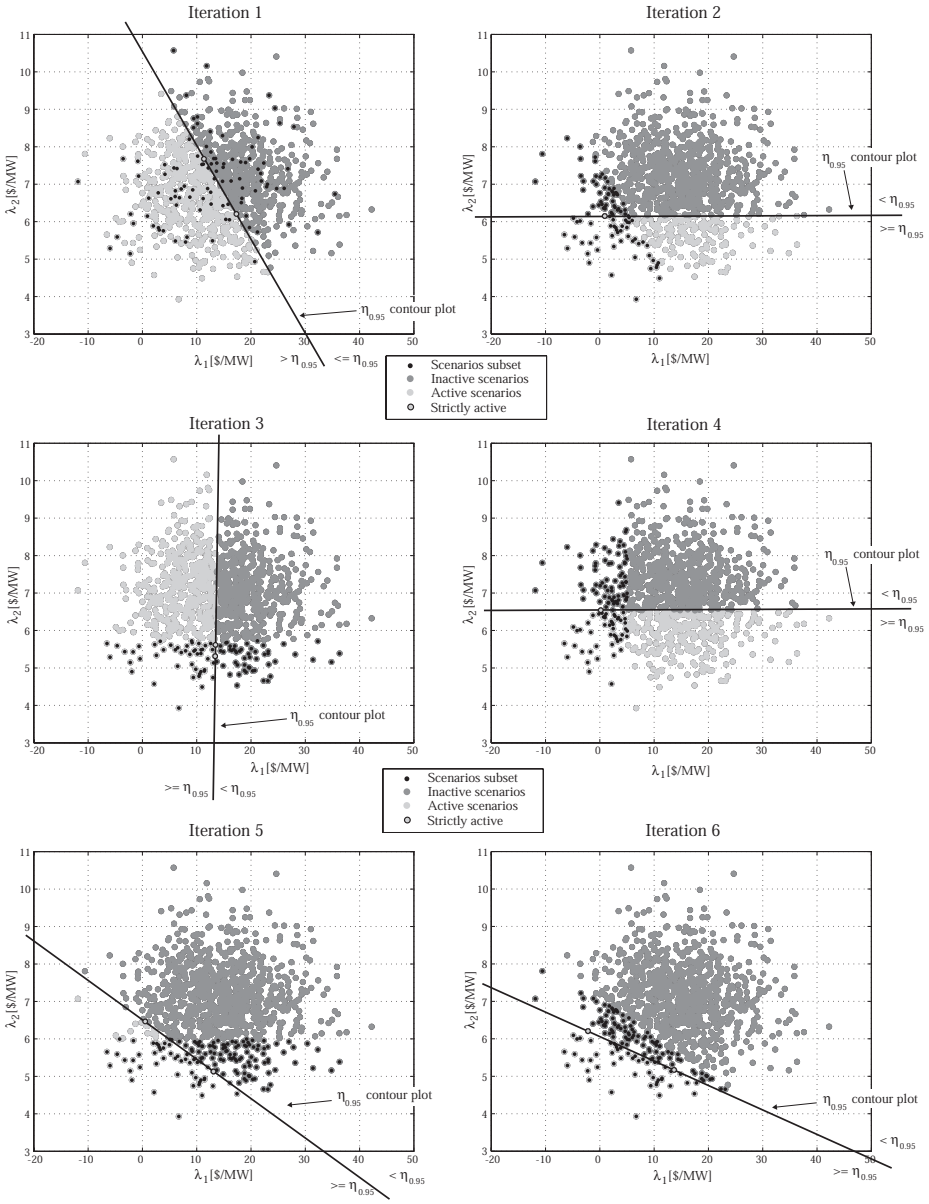

Fig. 4.2 Graphical illustration of the iterative reduction algorithm for the illustrative example considering 1000 scenarios.

is maximized, within a particular risk level based on loss or profit volatility, respectively. The producer decides how much power to sell through bilateral contracts at a fixed price spanning the time horizon, and for each period of the time horizon, it decides how much power to sell in the pool. This problem is mathematically formulated as a two-stage stochastic programming problem with recourse and un-

der rather general assumptions (Birge and Louveaux, 1997) can be equivalently expressed as the following nonlinear mixed-integer programming problem:

$$\begin{aligned}\underset{x_{ti}^{\mathrm{P}},\, x^{\mathrm{C}}}{\text{Minimize}} \quad & q\Bigg(\sum_{t=1}^{N_T} \sum_{i=1}^{N} \Big(-\lambda_{ti} x_{ti}^{\mathrm{P}} + c_{ti} \Big) - \lambda^{\mathrm{C}} x^{\mathrm{C}} N_T \Bigg) \\ & -\delta E \Bigg\{ \sum_{t=1}^{N_T} \sum_{i=1}^{N} \Big(\lambda_{ti} x_{ti}^{\mathrm{P}} - c_{ti} \Big) + \lambda^{\mathrm{C}} x^{\mathrm{C}} N_T \Bigg\}, \end{aligned} \tag{5.1}$$

subject to

$$(x_{ti}^{\mathrm{P}}, x^{\mathrm{C}}) \in \Pi, \tag{5.2}$$

where $q(\cdot)$ represents the conditional value-at-risk of the random loss objective function (negative of profits), $E\{\cdot\}$ represents the expected value of the random profit objective function, δ is non-negative weighting factor enforcing the tradeoff between expected profit and risk, x^{C} is the power sold through bilateral contracts (first-stage optimization variable), x_{ti}^{P} is the power sold in the pool during period t and scenario i (second-stage optimization variables), λ_{ti} is the pool price in period t and scenario i (stochastic parameters), λ^{C} is the bilateral contracts price, c_{ti} is the total production cost during period t and scenario i, Π is the feasible operating region of the producer, N_T is the number of considered time periods, and N is the number of considered scenarios, i.e cardinality of set $\mathcal{N}$.

The objective function (5.1) represents the tradeoff between risk, using CVaR, and expected profit. The profit of the producer is computed as the revenues obtained from selling energy in the pool, minus the total production cost of the producer plus the revenues obtained from the energy sold through bilateral contracts, which results in a nonlinear function. Constraint (5.2) represents operation constraints. A detailed formulation of the problem is given in Appendix A.

Data for the power producer are taken from Conejo et al (2004). The futures market price is taken to be 36 \$/MWh. Pool prices are random parameters which are assumed to be normally distributed with mean $\boldsymbol{\lambda}^{\mathrm{est}}$ and covariance matrix $\boldsymbol{V}_{\lambda}$ taken from Conejo et al (2004). Scenarios are synthetically simulated sampling from the joint probability density function.

Modeling pool prices through equiprobable scenarios, the problem (5.1)–(5.2) using CVaR as a measure of risk and replacing the expected value of the profit by the profit considering expected values of random variables, becomes:

$$\underset{x_{ti}^{\mathrm{P}},\, \bar{x}_t^{\mathrm{P}},\, x^{\mathrm{C}}}{\text{Minimize}} \ \eta_\alpha + \frac{1}{(1-\alpha)N} \sum_{i=1}^{N} u_i - \delta \left(\sum_{t=1}^{N_T} \Big(\bar{\lambda}_t \bar{x}_t^{\mathrm{P}} - \bar{c}_t \Big) + \lambda^{\mathrm{C}} x^{\mathrm{C}} N_T \right), \tag{5.3}$$

subject to

$$(x_{ti}^{\mathrm{P}}, \bar{x}_t^{\mathrm{P}}, x^{\mathrm{C}}) \in \Pi \tag{5.4}$$

$$u_i \geq \left(\sum_{t=1}^{N_T} \Big(-\lambda_{ti} x_{ti}^{\mathrm{P}} + c_t \Big) - \lambda^{\mathrm{C}} x^{\mathrm{C}} N_T \right) - \eta_\alpha; \quad \forall i \in \mathcal{N} \tag{5.5}$$

$$u_i \geq 0; \quad \forall i \in \mathcal{N}, \tag{5.6}$$

where $\bar{\lambda}_t$ represents expected value of pool prices, and $\bar{x}_t^{\mathrm{P}}$ and $\bar{c}_t$ represent the power sold in the pool and the total production cost during period t, respectively, considering the expected value of pool prices.

This self-scheduling problem is a mixed-integer nonlinear programming problem which is solved using SBB solver under GAMS (GAMS Development Corporation, 2011; Rosenthal, 2008) on a Linux-based server with one processor clocking at 2.3 GHz and 8 GB of RAM.

In order to validate and highlight the advantages of the iterative reduction algorithm proposed in the paper, the self-scheduling problem is solved for an increasing number of pool prices scenarios considering:

1. Formulation (5.1)–(5.2) considering all scenarios.
2. Formulation (5.3)–(5.6) using the iterative reduction algorithm.

Note that both the problem including all scenarios and subproblems within the iterative scheme could be solved using speeding-up CVaR optimization techniques, such as cutting plane methods, or more efficient implementations based on C++ or any other programming language. However, we rather not to use them in order to show the real gaining achieved using the proposed approach.

The confidence level α considered is 0.95, the initial b-value is equal to 2 , and weighting factor δ is equal to 2. Note that increasing the b-value will increase the size of the problem. The optimal solution of the iterative algorithm is attained in 2 iterations for all cases. Table 5.1 provides the Value-at-Risk (η_α), the Conditional Value-at-Risk (ϕ_α), the mean (μ_P) and standard deviation (σ_P) of the discrete loss distribution, and the power traded through bilateral contracts for different number of scenarios, for the problem considering all scenarios (upper part of the table) and for the iterative reduction algorithm (lower part of the table). Since the profit of the producer is represented through a nonlinear function, the expected value of the profit function is replaced by the profit function considering the expected value of pool prices (see problem (5.3)–(5.6)). This fact provokes that the results of solving the problem with all scenarios are different to the ones obtained using the iterative reduction algorithm. These differences in percent are shown in the lower part of the table in parenthesis. Note that differences in the Value-at-Risk, the Conditional Value-at-Risk and the mean are negligible, in the standard deviation are below 1%, and in the power traded through bilateral contracts are below 0.5%.

However, Figure 5.1 shows the mean and standard deviation of the profit function for different values of parameter δ, which proof that results are equivalent and both methods provide the same efficient frontier. In the problem solved to get the efficient frontier, we consider: i) the confidence level $\alpha = 0.99$, ii) the initial b-value equal to 2, and iii) 100 scenarios. Note that this efficient frontier explains why results from Table 5.1 are slightly different, because those results were obtained using the same δ-value, however same solutions could be achieved by using slightly different δ-values.

Table 5.2 provides the size of the problem considering i) all scenarios, and ii) the iterative reduction algorithm. This table includes the number of equations (n_{eq}), the number of continuos variables (n_{cv}) and the number of discrete variables (n_{dv}) of the corresponding models. Note the considerable reduction in the size of the problem when the iterative algorithm is used, which improves tractability.

In order to check the advantages of the iterative reduction technique proposed in the paper in terms of computational burden, the self-scheduling problem is

Global problem considering all scenarios					
Scen.	η_α [$]	ϕ_α [$]	μ_P [$]	σ_P [$]	p^{C} [MW]
10	-30476.21	-30476.21	-31096.41	431.80	190.36
50	-30482.84	-30422.79	-31129.00	394.13	191.90
100	-30478.76	-30435.24	-31210.32	397.35	190.57
150	-30518.13	-30436.36	-31221.96	390.94	190.46
200	-30531.89	-30457.46	-31231.91	397.59	190.24
300	-30557.44	-30461.75	-31175.00	379.72	190.49
Iterative reduction algorithm					
Scen.	η_α [$] (%)	ϕ_α [$] (%)	μ_P [$] (%)	σ_P [$] (%)	p^{C} [MW] (%)
10	-30478.39 (0.007)	-30478.39 (0.007)	-31095.21 (-0.004)	429.47 (-0.54)	190.82 (0.24)
50	-30484.40 (0.005)	-30425.23 (0.008)	-31127.68 (-0.004)	392.17 (-0.50)	192.34 (0.23)
100	-30479.88 (0.004)	-30436.69 (0.005)	-31209.56 (-0.002)	396.17 (-0.30)	190.83 (0.14)
150	-30517.73 (-0.001)	-30435.98 (-0.001)	-31222.15 (0.0006)	391.24 (0.08)	190.39 (-0.04)
200	-30533.77 (0.006)	-30458.61 (0.004)	-31231.31 (-0.002)	396.67 (-0.23)	190.44 (0.11)
300	-30559.81 (0.007)	-30466.04 (0.014)	-31172.49 (-0.008)	376.22 (-0.92)	191.33 (0.44)

Table 5.1 Results of the case study for different number of pool prices scenarios ($\alpha = 0.95$, $b = 2$).

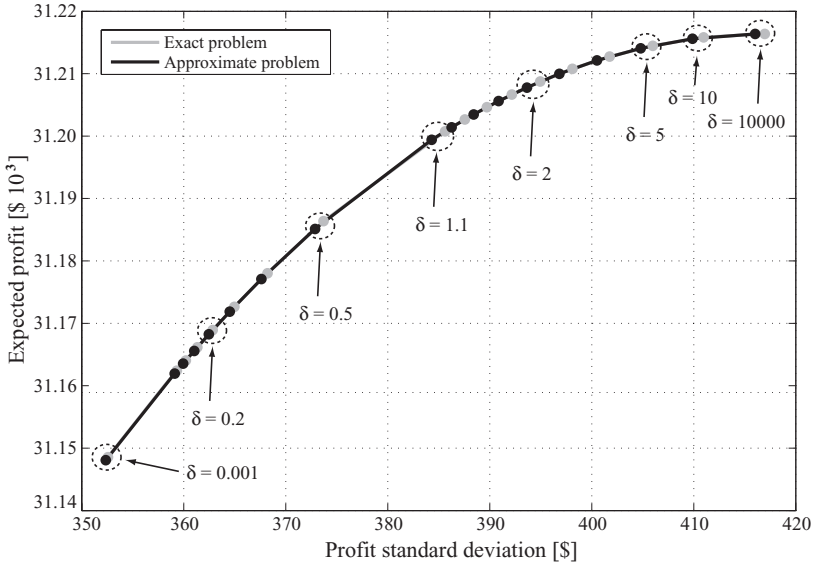

Fig. 5.1 Efficient frontier for different δ-values (α=0.99, b=2 and 100 scenarios).

solved for different values of the confidence level (0.9, 0.95 and 0.99), and for an

Scenarios	n_{eq}		n_{cv}		n_{dv}	
	Global	Iterative	Global	Iterative	Global	Iterative
10	1821	629	1115	420	96	96
50	8621	1309	5075	816	96	96
100	17121	2159	10025	1311	96	96
150	25621	3009	14975	1806	96	96
200	34121	3859	19925	2301	96	96
300	51121	5559	29825	3291	96	96

Table 5.2 Size of the problem for different number of scenarios ($\alpha = 0.95$, $b = 2$).

increasing number of pool prices scenarios (from 10 to 300 scenarios). For the iterative algorithm, b value is equal to 2 and δ is equal to 2 for all the problems.

Table 5.3 provides the computational time required to get the optimal solution considering i) all scenarios and ii) the iterative reduction algorithm. We make note that for all cases the iterative algorithm gets the solution in 2 iterations and reduces considerably the computational time with respect to the approach including all scenarios. Besides, the reduction in time tends to increase as the number of scenarios and the confidence level increase, achieving a saving of up to 99% (140 times faster) in the computational time required to solve the problem with $\alpha = 0.99$ and 300 scenarios. Finally, Figure 5.2 represents the evolution of the computational time for increasing number of scenarios and considering both solution methods.

Scenarios	$\alpha = 0.9$		$\alpha = 0.95$		$\alpha = 0.99$	
	Global	Iterative	Global	Iterative	Global	Iterative
10	0.28	0.09	0.32	0.16	0.29	0.12
50	5.91	1.54	7.18	0.71	6.33	0.29
100	32.92	5.94	28.91	1.98	19.80	0.47
150	73.563	13.86	55.78	4.15	45.06	0.77
200	141.53	25.41	98.61	6.69	90.76	1.01
300	390.07	71.38	342.51	15.33	257.67	1.78

Table 5.3 Computational time [min] for different number of scenarios and α values.

6 Conclusions

This paper proposes an iterative method for solving stochastic programming problems based on scenarios and considering CVaR as the measure of risk. This iterative approach allows obtaining the solution of the stochastic problem solving, within an iterative scheme, problems with a reduced number of scenarios. This strategy improves substantially computational efficiency for large-scale problems.

An example related to a power producer trading in a day-ahead electricity market is used for the purpose of illustration. In addition, simulations carried out

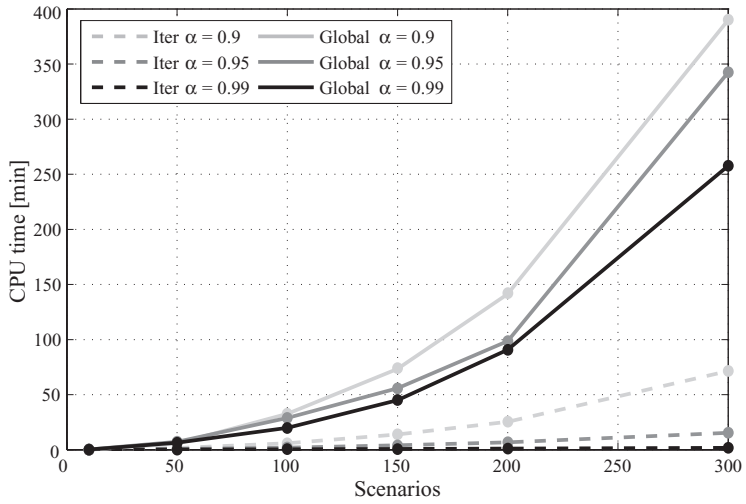

Fig. 5.2 Evolution of the computational time [min].

for a case study based on the self-scheduling problem of a power producer show the advantages of the proposed method, which can be summarized as follows:

1. The probability information embedded in the original set of scenarios is not lost. Therefore, the iterative method attains the same solution as the method considering all scenarios.
2. For each iteration, a problem with a reduced number of scenarios is solved, resulting in a considerable reduction of the computational time required to get the optimal solution of the problem.
3. The relative reduction in computational time tends to increase as the number of scenarios and the confidence level increase. Note that high confidence levels are commonly used in economics, management science and other stochastic user contexts.
4. It allows taking advantage of speeding up CVaR-optimization techniques for the resolution of subproblems.
5. The method is not limited to CVaR optimization. An extension is also given to counter-balance risk versus expected profit. However, this method is only equivalent under certain conditions, otherwise it constitutes an alternative and approximate method.
6. The method is independent platform, and can be used within any mathematical programming framework. The behavior is always analogous. The iterative algorithm is slower below a threshold number of scenarios, and increases efficiency above that threshold with respect to the problem including all scenarios. However, how those differences evolve with respect to the number of scenarios depends on several factors, such as the solver, the programming language, the computational resources, etc.

The main drawback of the proposed method is that although a heuristic procedure has been developed to achieve convergence, it does not assure reduction

of computational complexity and/or CPU time. However, simulation results show dramatic reductions in computing time for all cases, which encourage practitioners to use the proposed approach for large-scale problems. Nevertheless, further research must be done on the theoretical aspects related to convergence.

In addition, it would be interesting to analyze the influence of the solution method used to solve subproblems, on the overall performance of the iterative procedure. This is also a subject for further research.

Acknowledgements R. García-Bertrand is partly supported by the Ministry of Science and Innovation of Spain through CICYT Project ENE2009-07836. R. Mínguez is partly supported by the Spanish Ministry of Science and Innovation through the "Ramon y Cajal" program (RYC-2008-03207) and project "AMVAR" (CTM2010-15009) from Spanish Ministry of Science and Innovation.

We also thank the editor and referees for their very helpful comments and suggestions, which have led to an improved manuscript.

A Mathematical Formulation of the Case Study

The detailed formulation of the self-scheduling problem of a power producer solved in Section 5 is shown below:

$$\begin{aligned}\underset{x^{\mathrm{G}}_{ti},x^{\mathrm{P}}_{ti},x^{\mathrm{C}},v_t,y_t,z_t}{\text{Minimize}} \quad & q\Big(\sum_{t=1}^{N_T}\sum_{i=1}^{N}\Big(-\lambda_{ti}x^{\mathrm{P}}_{ti}+C^{\mathrm{L}}x^{\mathrm{G}}_{ti}+C^{\mathrm{Q}}(x^{\mathrm{G}}_{ti})^2\Big)\\ & +\sum_{t=1}^{N_T}\Big(C^{\mathrm{F}}v_t+C^{\mathrm{SD}}z_t+c^{\mathrm{SU}}_t(s_t)y_t\Big)-\lambda^{\mathrm{C}}x^{\mathrm{C}}N_T\Big)\\ & -\delta E\Big\{\sum_{t=1}^{N_T}\sum_{i=1}^{N}\Big(\lambda_{ti}x^{\mathrm{P}}_{ti}-C^{\mathrm{L}}x^{\mathrm{G}}_{ti}-C^{\mathrm{Q}}(x^{\mathrm{G}}_{ti})^2\Big)\\ & -\sum_{t=1}^{N_T}\Big(C^{\mathrm{F}}v_t+C^{\mathrm{SD}}z_t+c^{\mathrm{SU}}_t(s_t)y_t\Big)+\lambda^{\mathrm{C}}x^{\mathrm{C}}N_T\Big\},\end{aligned} \tag{A.1}$$

subject to

$$x^{\mathrm{G}}_{ti}=x^{\mathrm{P}}_{ti}+x^{\mathrm{C}};\quad t=1,\ldots,N_T;\ i=1,\ldots,N \tag{A.2}$$
$$x^{\mathrm{G}}_{ti}\geq \underline{P}\,v_t;\quad t=1,\ldots,N_T;\ i=1,\ldots,N \tag{A.3}$$
$$x^{\mathrm{G}}_{ti}\leq x^{\max}_{ti}v_t;\quad t=1,\ldots,N_T;\ i=1,\ldots,N \tag{A.4}$$
$$x^{\max}_{ti}\leq x^{\mathrm{G}}_{t-1,i}+R^{\mathrm{U}}v_t+R^{\mathrm{SU}}y_t;\quad t=1,\ldots,N_T;\ i=1,\ldots,N \tag{A.5}$$
$$x^{\max}_{ti}\leq \overline{P}(v_t-z_{t+1})+R^{\mathrm{SD}}z_{t+1};\quad t=1,\ldots,N_T;\ i=1,\ldots,N \tag{A.6}$$
$$x^{\mathrm{G}}_{t-1,i}\leq x^{\mathrm{G}}_{ti}+R^{\mathrm{D}}v_t+R^{\mathrm{SD}}z_t;\quad t=1,\ldots,N_T;\ i=1,\ldots,N \tag{A.7}$$
$$(s_{t-1}-T^{\mathrm{U}})(v_{t-1}-v_t)\geq 0;\quad t=1,\ldots,N_T \tag{A.8}$$
$$(s_{t-1}+T^{\mathrm{D}})(v_t-v_{t-1})\leq 0;\quad t=1,\ldots,N_T \tag{A.9}$$
$$y_t-z_t=v_t-v_{t-1};\quad t=1,\ldots,N_T \tag{A.10}$$
$$y_t+z_t\leq 1;\quad t=1,\ldots,N_T \tag{A.11}$$
$$v_t,\ y_t,\ z_t\in\{0,1\};\quad t=1,\ldots,N_T. \tag{A.12}$$

Variable x^{G}_{ti} is the power generated in period t and scenario i, x^{P}_{ti} is the power sold in the pool during period t and scenario i, and x^{C} is the power sold through bilateral contracts. Binary variables v_t, y_t and z_t represent if unit is committed, is started-up and is shut-down in period t, respectively. Pool price in period t and scenario i is λ_{ti}, and λ^{C} is the price

of the bilateral contracts. Variable costs are represented through a quadratic function whose linear and quadratic coefficients are C^{L} and C^{Q}, respectively. Fixed cost is C^{F}, shut-down cost is C^{SD} and start-up cost is $c_t^{\mathrm{SU}}(\cdot)$, which is a function of the time the unit has been shutdown in period t, s_t. Minimum power output of the unit is $\underline{P}$ and capacity of the unit is $\overline{P}$. Available maximum power output in period t and scenario i is $x_{ti}^{\max}$. Ramp-up limit, ramp-down limit, start-up ramp limit and shut-down ramp limit are R^{U}, R^{D}, R^{SU} and R^{SD}, respectively. Minimum up and down time of the unit are T^{U} and T^{D}, respectively. The number of periods unit has been on (+) or off (-) at the end of period t is s_t. Finally, N_T is the number of considered time periods, and N is the number of considered scenarios.

The objective function (A.1) represents the tradeoff between risk and expected profit. The profit comprises revenues from selling energy as well as production costs, fixed costs, start-up costs and shut-down costs. Constraints (A.2) express the power generated by the unit. Constraints (A.3) state the minimum power that must be produced by the unit. Constraints (A.4) force the unit to work below its available maximum power output. Constraints (A.5) and (A.6) state that the available maximum power output at every period depends on ramp rate limits. Constraints (A.7) limit the power generated at every period depending on ramp rate limits. Constraints (A.8) and (A.9) enforce feasibility in terms of minimum up and down time constraints, respectively. Constraints (A.10) and (A.11) preserve the logic of running, start-up, and shut-down status changes. And finally, constraints (A.12) constitute binary variables declaration.

Note that additional equations are needed to compute the time the unit has been shutdown in period t (s_t) and the number of periods the unit has been on or off at the end of period t (x_t). For sake of clarity these equations are not included in this appendix, but more information of these expressions can be found in Arroyo and Conejo (2000).

References

Ahmed S (2006) Convexity and decomposition of mean-risk stochastic programs. Math Program, Ser A 106(3):443–446

Arroyo JM, Conejo AJ (2000) Optimal response of a thermal unit to an electricity spot market. IEEE Trans Power Syst 15(3):1098–1104

Artzner P, Delbaen F, Eber JM, Heath D (1999) Coherent measures of risk. Math Financ 9(3):203–228

Birge JR, Louveaux F (1997) Introduction to Stochastic Programming. Springer Verlag, New York

Carrión M, Arroyo JM, Conejo AJ (2009) A bilevel stochastic programming approach for retailer futures market trading. IEEE Trans Power Syst 24(3):1446–1456

Conejo AJ, Nogales FJ, Arroyo JM, García-Bertrand R (2004) Risk-constrained self-scheduling of a thermal power producer. IEEE Trans Power Syst 19(3):1569–1574

Conejo AJ, García-Bertrand R, Carrión M, Caballero A, de Andrés A (2008) Optimal involvement in futures markets of a power producer. IEEE Trans Power Syst 23(2):703–711

Dupačová J, Consigli G, Wallace SW (2000) Scenarios for multistage stochastic programs. Ann Oper Res 100(1):25–53

Dupačová J, Gröwe-Kuska N, Römisch W (2003) Scenario reduction in stochastic programming. An approach using probability metrics. Math Program, Ser A 95(3):493–511

Eichhorn A, Römisch W, Wegner I (2005) Mean-risk optimization of electricity portfolios using multiperiod polyhedral risk measures. In: IEEE St. Petersburg PowerTech Proceedings

Fábián CI (2008) Handlingn CVaR objectives and constraints in two-stage stochastic models. Eur J Oper Res 191(3):888–911

Fábián CI, Vesprémi A (2008) Algorithms for handling CVaR constraints in dynamic stochastic programming models with applications to finance. J Risk 10(3):111–131

GAMS Development Corporation (2011) GAMS - The solver manuals. Washington, DC

Gröwe-Kuska N, Heitsch H, Römisch W (2003) Scenario reduction and scenario tree construction for power management problems. In: 2003 IEEE Bologna Power Tech Conference Proceedings, vol 3

Hatami A, Seifi H, Sheikh-El Eslami MK (2011) A stochastic-based decision-making framework for an electricity retailer: Time-of-use pricing and electricity portfolio optimization. IEEE Trans Power Syst 26(4):1808–1816

Heitsch H, Römisch W (2003) Scenario reduction algorithms in stochastic programming. Comput Optim Appl 24(2-3):187–206
Heitsch H, Römisch W (2007) A note on scenario reduction for two-stage stochastic programs. Oper Res Lett 35(6):731–738
Heitsch H, Römisch W (2009a) Scenario tree modeling for multistage stochastic programs. Math Program 118(2):371–406
Heitsch H, Römisch W (2009b) Scenario tree reduction for multistage stochastic programs. Comput Manag Sci 6(2):117–133
Heitsch H, Römisch W, Strugarek C (2006) Stability of multistage stochastic programs. SIAM J Optimization 17(2):511–525
Henrion R, Küchler C, Römisch W (2008) Discrepancy distances and scenario reduction in two-stage stochastic mixed-integer programming. J Ind Manage Optim 4(2):363–384
Henrion R, Küchler C, Römisch W (2009) Scenario reduction in stochastic programming with respect to discrepancy distances. Comput Optim Appl 43(1):67–93
Høyland K, Wallace SW (2001) Generating scenario trees for multistage decision problems. Manage Sci 47(2):295–307
Høyland K, Kaut M, Wallace SW (2003) A heuristic for moment-matching scenario generation. Comput Optim Appl 24(2-3):169–185
Jabr RA (2005) Robust self-scheduling under price uncertainty using conditional value-at-risk. IEEE Trans Power Syst 20(4):1852–1858
Krokhmal P, Palmquist J, Uryasev S (2002) Portfolio optimization with conditional value-at-risk objective and constraints. J Risk 4(2):11–27
Künzi-Bay A, Mayer J (2006) Computation aspects of minimizing conditional value-at risk. Comput Manag Sci 3(1):3–27
Lim C, Sherali HD, Uryasev S (2010) Portfolio optimization by minimizing conditional value-at-risk via nondifferentiable optimization. Comput Optim Appl 46(3):391–415
Mínguez R, Conejo AJ, García-Bertrand R (2011) Reliability and decomposition techniques to solve certain class of stochastic programming problems. Reliab Eng Syst Safety 96(2):314–323
Olsson M, Soder L (2004) Generation of regulating power price scenarios. In: 2004 International Conference on Probabilistic Methods Applied to Power Systems, pp 26–31
Pflug GC (2001) Scenario tree generation for multiperiod financial optimization by optimal discretization. Math Program 89(2):251–271
Pineda S, Conejo AJ (2010) Scenario reduction for risk-averse electricity trading. IET Gener Transm Distrib 4(6):694–705
Rockafellar RT, Uryasev S (2000) Optimization of conditional value-at risk. J Risk 2(3):21–41
Rockafellar RT, Uryasev S (2002) Conditional value-at risk for general loss distributions. J Bank Finan 26(7):1443–1471
Rosenthal RE (2008) GAMS - A User's Guide. GAMS Development Corporation, Washington D.C.
Zhang W, Wang X (2009) Hedge contract characterization and risk-constrained electricity procurement. IEEE Trans Power Syst 24(3):1547–1558