Projecting generation decisions induced by a stochastic program on a family of supply curve functions

Boris Defourny, Louis Wehenkel
Department of Electrical Engineering and Computer Science
University of Liège – Sart-Tilman B28 – B-4000 Liège
bdf@montefiore.ulg.ac.be, L.Wehenkel@ulg.ac.be

***Abstract*— We propose to post-process the results of a scenario based stochastic program by projecting its decisions on a parameterized space of policies. By doing so the risk of overfitting to the set of scenarios used in the stochastic program is reduced. A proper choice of the structure of the space of policies allows one to exploit them in the context of novel scenarios, be it for Monte-Carlo based value estimation or for use in real-life conditions. These ideas are presented in the context of planning the exploitation of electric energy resources or for evaluating the economic value of a portfolio of assets.**

I. Introduction

Investment decisions in the electric energy generation industry often involve at some stage of the decision process the valuation of a portfolio of production assets, reservoirs and contracts. Indeed, the economic value of a new resource depends not only on its own characteristics, but also on how it can be exploited in combination with existing ones, and how its availability may impact the way other resources are managed.

In this paper, we consider the problem of computing good energy resource management strategies under uncertainty, within a fixed set of energy resources, and for a given time horizon. Optimally exploiting a producer's set of resources is challenging, due to the many sources of uncertainties, the dynamic dimension of the problem, and the number of decisions to take. We believe that the implications of an improvement over existing methods would go beyond the sphere of operational research. Indeed, we think that the valuation of hypothetical sets of resources would benefit from more precise models for their exploitation.

Independently of the temporal horizon over which electricity production is optimized, the uncertainty comes from the possible evolutions of demand, weather, gas price, . . . It also comes from events like the unexpected unavailability of a power plant, which amount to uncertainty on the generation capacity itself. If the producer operates in a competitive market environment, a further uncertainty source is the electricity market itself. A good resource management strategy thus has to take into account the fact that the future is uncertain and it should model possible recourse decisions based on new information.

We suppose that uncertainty is represented by a given set of possible scenarios organized in the form of a tree [2]. This is appropriate when we lack statistical models while historical information is available to build a set of possible future scenarios. If statistical models are available, they can be used to generate a set of scenarios by Monte-Carlo sampling. Given a scenario tree, an optimization problem may be stated to search for an optimal policy contingent on the particular scenarios of the tree. This policy then has to be generalized in order to allow its exploitation on any new scenario. The policy generalization is thus important when we want to exploit the policy in practice or when we want to evaluate its expected cost in a sound way, by testing it against a set of unseen scenarios.

In this paper, we focus on the policy generalization problem (we refer the interested reader to [3] for some existing scenario tree building methods, and [6] for some methods that are under development). To generalize the decisions derived from a scenario tree, we propose to project them on an a priori defined class of policies expressed in the form of parameterized supply curve functions (SCF). This can be achieved by formulating an optimization problem to compute SCF's which mimic as closely as possible the decisions computed on the basis of the given tree of training scenarios. An interesting feature of this approach is its natural compliance with market logic. Market rules could even suggest a particular format for the SCF's. Besides, a policy whose performance would rely on subtleties that could hardly be approximated under a market-imposed format might lead to overestimating the value of the portfolio of resources.

The paper is organized as follows. In section II, we detail a framework to generalize decisions chosen on the basis of a simplified representation of uncertainty. In section III, we describe a problem of energy resource management under uncertainty. In section IV we explain how we can infer supply curve functions from the optimal decisions contingent on the simplified model of uncertainty. The section V raises some open issues.

II. General framework

We formally introduce the stochastic programming framework in which we have cast our problem. Here we do not assume that the uncertainty model satisfies the Markov property nor any specific structure of the cost function. Notice however that in order to lead to a tractable solution, the final optimization problem should be convex or quasi-convex, which obviously imposes some restrictions on the cost function.

A. Stochastic processes

First we recall the usual framework in which one can define uncertain dynamic processes to be observed between now and a time horizon T, taking account of the fact that information about these processes become progressively available as time elapses.

Let $(\Omega, \mathcal{F}, P)$ denote a probability space, and $\mathbb{T} = \{1, \ldots, T\}$ a finite set of discrete time indices. Let $\mathcal{W} = \{\mathcal{W}_t : t \in \mathbb{T}\}$ be a stochastic process of dimension d defined on the probability space. Given $\omega \in \Omega$, we will denote by w^ω the corresponding realization of $\mathcal{W}$, $w_t^\omega \in \mathbb{R}^d$ its value at time t, and $w_{1:t}^\omega$ its sequence of values up to time t. We call w^ω a scenario.

Let $\mathbb{F} = (\mathcal{F}_t)_{t \in \mathbb{T}}$ be the filtration on $(\Omega, \mathcal{F}, P)$ generated by $\mathcal{W}$. The filtration is an increasing sequence of sub σ-fields of $\mathcal{F}$, i.e. $\{\emptyset, \Omega\} \subset \mathcal{F}_1 \subset \mathcal{F}_2 \subset \ldots \subset \mathcal{F}_T \subset \mathcal{F}$, and can be thought of as a model of growing information with time. The sub σ-fields are here $\mathcal{F}_t = \sigma(\mathcal{W}_u : u \leq t)$, i.e. the smallest σ-fields such that the random variables $\{\mathcal{W}_u : u \leq t\}$ are $\mathcal{F}_t$-measurable. This technical condition, the measurability condition, is the one ensuring that at any time, the past of the stochastic process is fully known but its future remains uncertain.

B. Scenario trees

Next we set up a framework to work with the uncertain process $\mathcal{W}$ without relying on a full description of the probability space. We are motivated either by a lack of reliable statistical data, either by the wish to lighten the description. We consider a set of scenarios deemed representative of $\mathcal{W}$, organized in a way that makes their probability and the information structure of the set close to the initial probability measure P and filtration $\mathbb{F}$ in some sense.

Let $S = \{\omega_1, \ldots, \omega_N\}$ be a subset of N points of Ω, and let $W_S = \{w^{\omega_1}, \ldots, w^{\omega_N}\}$ be the set of corresponding scenarios. Let $A_0, A_1, \ldots A_T$ be a sequence of nested partitions of S. The sequence starts with the trivial partition $A_0 = \{S\}$, goes on with $A_t = \{A_t^1, \ldots, A_t^{k_t}\}$, and ends with the partition that splits S in N singletons : $A_T = \{\{\omega_1\}, \ldots, \{\omega_N\}\}$. The nesting property implies that for any ω_i and ω_j that are in the same class A_t, $t > 1$, they will be in the same class of A_{t-1}. Coming back to the scenarios, we require that $w_{1:t}^\omega$ be constant in a same A_t^u, i.e. $\forall t \in \{1, \ldots, T\}, \forall u \in \{1, \ldots, k_t\}$:

$$(\omega_i \in A_t^u \wedge \omega_j \in A_t^u) \Rightarrow w_{1:t}^{\omega_i} = w_{1:t}^{\omega_j}. \qquad (1)$$

We can now define the scenario tree $\mathcal{T}$ associated to S. It is a graph with N leaves (terminal vertices) labeled from 1 to N, and N directed paths of length T from the root vertex to the leaves. The edges are labeled such that the sequence of labels collected while following the i-th path corresponds to $w^{\omega_i} = (w_1^{\omega_i}, w_2^{\omega_i}, \ldots, w_T^{\omega_i})$. Moreover, assuming that a probability distribution p_i is defined on the $\omega_i \in S$, we may associate to each edge the probability that a path passes through this edge.

A scenario tree can be viewed as a measurable discretization of the stochastic process $\mathcal{W}$. It provides a discretized version of the distributions $P(\mathcal{W}_{1:t}), \forall t = 1, \ldots, T$ which respects the measurability condition.

C. System dynamics and decision variables

We set up a framework to cast a policy search problem. Say we want to steer a system subject to dynamic constraints. There is a cost associated to the system trajectories (time-indexed sequences of states), and to the actions taken. We suppose that the cost and the dynamics depend on an uncertain process $\mathcal{W}$. A scenario tree $\mathcal{T}$ is given as a model of the uncertainties induced by the process $\mathcal{W}$. The tree ideally achieves a good tradeoff between the quality of its uncertainty model and the tractability of the subsequent policy search problem.

Let $X_t \subset \mathbb{R}^{n_t}, \forall t = 1, \ldots, T+1$ and $U_t \subset \mathbb{R}^{m_t}, \forall t = 1, \ldots, T$ denote the state space and the action (decision) space of the dynamic system at time t. We assume that X_t and U_t are continuous and/or of high dimension.

In a tree-based stochastic programming framework, one defines an optimization problem which is parameterized by the scenarios of the tree $\mathcal{T}$. More precisely, to each edge from a node at time $t-1$ to a node at time t, the value of $w_t^{\omega_i}$ representing the realization of the stochastic process $\mathcal{W}$ at time t (common to the scenarios encoded in $\mathcal{T}$ traversing this edge) becomes a parameter of the problem.

On the other hand, the system state and decision variables at the different time steps and in the different scenarios become the variables of the optimization problem. More precisely, for $t = 1 \ldots, T+1$ and each node n at time t we define a state variable x_t^n, and for $t = 1, \ldots, T$ we define a control variable u_t^n.

The optimization problem is further constrained by

- initial conditions on the system state x_1 and system dynamics $x_{t+1} = f_t(x_t, u_t, w_t), \forall t = 1, \ldots T$;
- technical constraints on u_t and x_t.

The size of the optimization problem is directly proportional to the size of $\mathcal{T}$.

D. Optimization problem

Typically, the objective function one wants to minimize in a stochastic programming problem is an expectation over all possible realizations of the stochastic process $\mathcal{W}$ of an additive over time cost function, e.g.

$$\text{minimize } E_{\mathcal{W}}\left\{\sum_{t=1}^{T} c_t(x_t^w, u_t^w, w) + g(x_{T+1}^w)\right\}, \quad (2)$$

where c_t denotes the instantaneous cost and g the terminal cost. The superscript ^{w} stresses the dependence of states and actions at any time on the realization of the stochastic process.

Replacing the stochastic process $\mathcal{W}$ by a simplified uncertainty model represented by a tree $\mathcal{T}$ allows to replace the expectations by means, i.e. finite weighted sums over the sample points of Ω behind the scenarios in the tree. The resulting, much simpler formulation becomes thus

$$\text{minimize } \sum_{i=1}^{N} p_i\left\{\sum_{t=1}^{T} c_t(x_t^{\omega_i}, u_t^{\omega_i}, \omega_i) + g(x_{T+1}^{\omega_i})\right\}, \quad (3)$$

where the superscript $^{\omega_i}$ stresses the dependence of states and actions on the particular scenario.

In this minimization problem, the variables $u_t^{\omega_i}$ are constrained to be nonanticipative, i.e.

$$u_t^{\omega_i} = u_t^{\omega_j} \text{ if } w_{1:t-1}^{\omega_i} = w_{1:t-1}^{\omega_j}, \text{ or } t = 1. \quad (4)$$

These constraints ensure that the decisions at time t are only function of the information that could already be gathered at time t.

The initial condition is imposed by the set of constraints

$$x_1^{\omega_i} = x_1, \forall i = 1, \ldots, N, \quad (5)$$

and the system dynamics are expressed by the following set of constraints ($\forall i = 1, \ldots, N, \forall t = 1, \ldots, T$) :

$$x_{t+1}^{\omega_i} = f_t(x_t^{\omega_i}, u_t^{\omega_i}, w_t^{\omega_i}). \quad (6)$$

It is worth noticing that these three sets of constraints impose also that

$$x_t^{\omega_i} = x_t^{\omega_j} \text{ if } w_{1:t-1}^{\omega_i} = w_{1:t-1}^{\omega_j}, \text{ or } t = 1, \quad (7)$$

and hence imply that both state and decision variables may indeed be plugged into the scenario tree as discussed in the previous subsection.

Further constraints on the allowed decision variables might be imposed by imposing a class of policies $\mu_t : (X_t, W_{1:t-1}) \rightarrow U_t$ where $\mu_t \in \mathcal{H}$. A good choice for the class $\mathcal{H}$ might come from insight about what the optimal policy of the problem at hand should look like.

On the other hand, we can choose to search for a policy in another class $\mathcal{H}'$, and then project the found policy on the class of interest $\mathcal{H}$. Indeed, technical considerations about the tractability of the optimization problem — mainly convexity issues — might prevent us to work directly with $\mathcal{H}$.

E. Policy projection

There are however drawbacks associated to the use of scenario trees when looking for a good policy. Indeed, the stochastic programming formulation leads to fitting as much as possible the decisions to the finite (and generally relatively small) set of scenarios in the tree. This leads to a strong risk of overfitting and underestimation of the optimal cost. Furthermore, the stochastic programming framework does not provide decisions in a form which allows one to apply them to novel unseen scenarios. Hence it does not allow one to assess the quality of a policy computed using an independent test set of scenarios for the same problem or to exploit the policy in practice.

Let's propose a framework to address these questions. Let $\mathcal{LS}$ be a learning set of optimal information-decision pairs and time steps (y_t^*, u_t^*, t) collected from the solution of the optimization problem under uncertainty. Here y_t represents the information available to the decision maker at time t, e.g. both the system state x_t as well as relevant past observations of the stochastic process $w_{1:t-1}$. Let $\mathcal{H}$ be a class of parameterized decision policies, i.e. functions from the information space to the decision space. Picking a particular policy amounts now to choose the value of a finite dimensional parameter vector, say $\lambda \in \mathbb{R}^p$. We call the corresponding policy Π_λ. Assume we can find a λ such that the decisions induced by Π_λ are close to those in the learning set $\mathcal{LS}$. If $\mathcal{H}$ is well chosen, we can hope that Π_λ will perform well.

The problem of finding the best parameter λ (picking a policy) can be viewed as a fitting problem, or an estimation problem (for example a maximum likelihood estimation problem), or a learning problem. We will say that the policy is computed by projecting the training set $\mathcal{LS}$ derived by stochastic programming on $\mathcal{H}$.

III. Application to Energy Resources Management

An electricity producer faces a repeated bidding problem under uncertainty, and wants to express her operating policy under the market-imposed supply curve format. The policy aims at maximizing the expectation of the cumulated profit over the time horizon, by optimally exploiting different production technologies with stock constraints, and hedging against unfavorable scenarios.

A. Format for the supply curve functions

At each time step, the producer has to submit a supply curve function to a market authority. The supply curve function relates quantities to prices. The clearing price equates the demand with the sum of the contributions of all suppliers for that price. At the end of the clearing process, each market participant knows how much to produce, and the market price. The revenue (price times quantity) is thus known, and the cost depends of the mix of technologies used to produce the prescribed quantity. In principle, the producer's supply curve function is chosen such that the revenue covers the cost, whatever the quantity.

We restrict our attention to piecewise linear supply curves, as this is a particular format imposed in some markets. A supply curve can thus be specified by a finite number of points. There are some additional requirements : the curves should be nondecreasing since a higher price should not lead to a decrease in the sold quantity, and the curves must start from the origin (no quantity at null price). Formally, we look for piecewise linear curves expressed as

$$((0,0),(q_1,\pi_1),(q_2,\pi_2),\ldots,(q_N,\pi_N))$$

where $0, q_1, \ldots, q_N$ and $0, \pi_1, \ldots, \pi_N$ form nonincreasing sequences of quantities and prices respectively. Unless no capacity limit exists, we add to the last segment a vertical ray $((q_N,\pi_N),(q_N,\infty))$.

B. Production technologies and their operating constraints

We suppose that the producer has two technologies : Thermal and Hydraulic.

The operating constraints and different cost structure are deemed to have such an impact on the bid profitability that it is worth modeling them finely in the bid optimization process. Fixed costs such as capital or maintenance costs are believed to have no effect on the operation strategy and are therefore disregarded.

For thermal production, we assume a quadratic cost c_t with respect to the quantity g_t, and a restriction on the production due to the installed capacity G. The quadratic cost models the contribution of less efficient plants to further increase the total thermal production. (A more realistic option would be to consider a convex piecewise linear, or convex piecewise linear-quadratic cost. Moreover, we should let the cost depend on the fuel price. We will neglect that in our uncertainty model for the time being. It is not unrealistic if the producer has entered in a swap contract, i.e. a financial contract that compensates for fuel price variations.)

For hydraulic production, the operating cost is neglected. A limit on the rate of extraction of water, and a limit from the total water available in reservoirs, restrict the hydraulic production h_t. In fact the hydraulic production can spare thermal production. This is particularly valuable to profitably sustain a high demand, since the marginal thermal cost $\mathrm{d}c_t/\mathrm{d}g_t$ is assumed to be increasing.

The reservoir level x_t is lowered by the water used for electricity production h_t, and by water level increases due to precipitation v_t. This yields a linear state transition relation

$$x_{t+1} = x_t - \alpha h_t + v_t$$

expressed in the reservoir level unit.

C. Uncertainty model

The uncertainty is modeled as a set of scenarios that specify for each time step the realization of the random variables relevant to the problem. Statistical description of stochastic processes is not used explicitly — if ever it exists. The scenarios are believed to be indistinguishable at the beginning, a situation that can be modeled by resorting to a scenario tree. Each path from the root to a leave of the tree represents a given scenario over the considered period of time.

We can associate to the nodes of the tree some decisions, to be applied should the realization of the stochastic processes lead to that node. In principle the best hedged decisions are those associated with the earlier time steps, because the future is correctly represented by the sequel of the tree. Decisions closer to leaves rely more heavily on a deterministic description of the future and on the terminal cost functions, and so are typically less well hedged.

The scenario tree also specifies probabilities of its scenarios. These probabilities are involved in the computation of expectations appearing in the objective function of the optimization program under uncertainty.

D. Sources of uncertainty

The demand at time t is assumed to be inelastic, but its level d_t is uncertain. The bids of other suppliers are also uncertain. We can simplify the situation by neglecting any strategic behavior of other market participants and give instead some elasticity to the demand curve. Indeed, price cuts usually bring back some of the demand met otherwise by other suppliers. For example, the inverse [residual] demand curve

$$\pi_t = (d_t - q_t)/m_t \tag{8}$$

giving the market price π_t in function of the supplier's quantity $q_t = h_t + g_t$ assumes that the aggregated supply curve function of the other suppliers is $\pi_t = m_t Q_t$, where Q_t is the aggregated quantity of the other suppliers, and m_t acts as their marginal cost. Thus the uncertain demand seen by the producer could be represented by the process $(d_1,m_1),(d_2,m_2),\ldots$

Another important source of uncertainty is the water v_t that refills the reservoir. The perspective of a dry season should significantly change the way hydraulic plants are called up.

We completely neglect in our model other sources of uncertainty such as capacity reductions due to unexpected outages and fluctuations of thermal costs due to uncertainties in fuel purchase prices.

E. Optimization problem

We end up with a large optimization problem $\mathcal{P}(\mathcal{T})$ posed over a scenario tree $\mathcal{T}$. To keep notation coherent with section II, we use u_t instead of $\begin{bmatrix} h_t & g_t \end{bmatrix}$ to denote the decisions at time t, and w_t instead of $\begin{bmatrix} v_t & d_t & m_t \end{bmatrix}$ to denote the uncertain process at time t.

The problem can be stated as

$$\text{maximize } \mathbf{E}_{\mathcal{T}} \left\{ \sum_{t=1}^{T} b(x_t, u_t, w_t) + B(x_{T+1}) \right\} \quad (9)$$

$$\text{subject to } (u_t^\omega, x_t^\omega, w_t^\omega) \in F \quad (10)$$

$$x_1^\omega = x_1, \; x_{t+1}^\omega = f(x_t^\omega, u_t^\omega, w_t^\omega) \quad (11)$$

$$u_t^\omega = u_t^{\omega'} \text{ if } w_{1:t-1}^\omega = w_{1:t-1}^{\omega'} \text{ or } t = 1. \quad (12)$$

The function $b(\cdot)$ is the difference between the revenue and the cost at time t. The function $B(\cdot)$ is supposed to mitigate the effect of the time horizon truncation. Ideally B would provide a good estimate of the future benefits that could be generated by using the water remaining in the reservoirs at the end of the optimization horizon. Another possibility, often used in annual hydro-scheduling problems is to define B in such a way that programs leading to a large difference between the water levels at time $t = 1$ and $t = T + 1$ are penalized.

The expectation operator $\mathbf{E}_{\mathcal{T}}$ applied to any function $f(x)$ stands for the mean $\sum_{\omega \in \mathcal{T}} p^\omega f(x^\omega)$ over all the scenarios ω of probability p^ω featured by the scenario tree $\mathcal{T}$. A path from the root to one of the N leaves of the tree defines a realization $w_{1:T}^\omega = [w_1^\omega, \ldots, w_T^\omega]$ of the uncertain process, i.e. a scenario. It is indexed by $1 \leq \omega \leq N$, and induces the creation of proper optimization variables : the decisions variables u_t^ω and the state variables x_t^ω. The constraints that we will call the operation constraints (10) and the dynamics constraints (11) have thus to be replicated for each scenario ω. The measurability constraints (12) ensures that the decision policy is implementable : if two scenarios ω, ω' are indistinguishable up to time t, the optimal decisions are necessarily the same up to time t.

F. Tractability

The solving step can involve dynamic programming, stochastic programming, and decomposition methods. It may become intractable if the scenario tree has many branches and the time horizon is long. For example, for a time horizon of T time steps and a binary tree with splits occurring every T/k time steps, we will have $N = 2^k$ scenarios. If for a given scenario ω and time step t we need n_x optimization variables to describe the state and n_u variables for the decision, that makes $NT(n_x + n_u)$ optimization variables. There will be NTn_u measurability constraints (12), plus $NT(m_{10} + m_{11})$ constraints, where m_{10} and m_{11} are the number of constraints per scenario of type (10) and (11) respectively.

This short counting argument illustrates the fact that in practice we are able to feature a very limited set of scenarios : we must resort to a “sparse” scenario tree $\mathcal{T}$.

In the sequel of the paper we consider that a solution $(x, u)_{\mathcal{T}}^* = (x_t^\omega, u_t^\omega)_{1 \leq \omega \leq N}^*$ has been found for that tree $\mathcal{T}$, and that we now try to generalize the decisions — at least those corresponding to the first time steps — to unseen scenarios, without losing too much optimality.

IV. Projection of the solution

Up to now the supply curve function format has not been taken into account. The optimization problem $\mathcal{P}(\mathcal{T})$ is directly expressed in terms of quantities. We could see $\mathcal{P}(\mathcal{T})$ as the relaxation of a problem where the bidding format is imposed. Our hope is that the projection, or “rounding”, of the optimal solution on the space of policies generated through the use of the supply curve functions format will make the solution less dependent of the particular sparse scenario tree $\mathcal{T}$ chosen to model the uncertainty.

A. Fitting a SCF on a set of decisions

First let’s assume that we want to express a set of decisions $\{h_{t_i}^{\omega_i}\}_{1 \leq i \leq D}$ and $\{g_{t_i}^{\omega_i}\}_{1 \leq i \leq D}$ with supply curve functions, one for each production technology. $(t_1, \ldots, t_D)$ and $(\omega_1, \ldots, \omega_D)$ are the time and scenario indices that define the set. The set is denoted $\mathcal{S}$. We reason with the quantity g.

The essential characteristic of a SCF is that it is nondecreasing :

$$\pi_i \geq \pi_j \text{ whenever } g_i \geq g_j, \quad (13)$$

where the π_i’s stand for the prices and can be deduced from our chosen price model, say (8), if they do not appear explicitly as optimization variables in $\mathcal{P}(\mathcal{T})$. The constraint (13) can be imposed in a fitting framework [4]. For example it suffices to add it to a least-square program to obtain the best nondecreasing function interpolating the data in the least square sense.

Let’s note that the nondecreasing constraint is easier to impose a posteriori. First, the ordering of the quantities is known once the optimization program $\mathcal{P}(\mathcal{T})$ is solved. But more importantly, it is not clear from the beginning which time steps and which scenarios should be clustered such that their associated decisions are expressed with the same supply curve functions.

B. Choosing which decisions to merge in a same SCF

We should group decisions that occurred in similar situations. This amounts to cluster the objects in the learning set on the basis of the state variables and the time steps, so that we can later build distinct supply curve functions on each cluster. In our running problem the reservoir level is the key variable that makes up a situation. The demand itself, level and elasticity, should not interfere too much, since it is the role of the supply curve function to be robust with respect to the demand.

Forecasts can also differentiate the situations. In a more advanced version of the problem, we could have in the state space an indicator for trends. In our problem we were dealing mainly with weather-related processes and a water stock. If we had a gas stock, an indicator for future gas price would be part of the state space, since the indicator could alter the arbitrage between gas turbines and a concurrent technology in the exploitation context.

The number of distinct situations and thus the number of supply curve functions forming the policy is subject to a tradeoff. On the one hand, we would like to specialize the supply curve functions as much as possible with respect to the information that will be available at the moment of the bid. On the other hand, we need to be able to distinguish between the situations.

From the trivial fact that the more curves we try to fit, the less points we have per curve, we suggest next a method to ensure that the learning set is large enough.

C. Generating more decisions for a same SCF

If we go back to the scenario tree building problem itself, something we considered solved a priori, we remark that we had to face a similar tradeoff. We had to represent various situations — except that we did not know at that stage the optimal decisions and stock levels — while keeping the tree complexity low. Hence the following observation : for each scenario in the tree, we have a bundle of scenarios very similar to it. This leads to the idea of solving the program $\mathcal{P}$ on a sequence of perturbed trees $\{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_M\}$ to collect more objects in the learning set. The perturbed trees are obtained by replacing each scenario by one of their substitute in the bundle.

With sufficient data points in the learning set, we can now hope to build distinct supply curves valid for a limited number of time steps and a small reservoir level spread.

V. Open issues

In this paper we sketched a method to generalize decisions computed from a selection of scenarios. The generalization step is necessary to test policies against new scenarios and estimate their expected cost. Ultimately the expected cost of a good policy reflects the performance of an underlying set of production means, and thus can guide investment strategies. We intend to conduct numerical experiments and see if a policy parameterized with supply curve functions suggests sensible operation rules and performs well.

What is the risk of using supply curve functions for arbitraging repeatedly between production technologies ? Could the supply curve functions induce aberrant decisions in some cases, in the sense that their use would clearly damage the expected cumulated profit over the studied time horizon ? The requirement that the quantities of each technology are non-decreasing with respect to the price might prove too strong. After all, usually only the aggregated quantity in function of the price has to be submitted to the market authority. Therefore, we will consider relaxing our requirements if the numerical results are not satisfying. Whether this would have to be done in all situations or only on a subset of situations – typically for crisis situations, such as the loss of generating capacity – remains an open issue at the current stage of this research.

VI. Acknowledgments

This paper presents research results of the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office. The authors want to express their gratitude to Damien Ernst for inspiring discussions about stochastic dynamic programming. They also want to thank the researchers from the R&D team OSIRIS (Optimisation, Simulation, Risque et Statistiques) at Electricité de France (EDF) for their discussions about generation planning practice.

The scientific responsibility of this work rests with its authors.

References

[1] R. T. Rockafellar, *Optimization under uncertainty*, Lecture notes, University of Washington.

[2] R. T. Rockafellar and Roger J.-B. Wets, *Scenarios and policy aggregation in optimization under uncertainty*, Math. Oper. Res., Vol. 16 (1), pp. 119–147, 1991.

[3] J. Dupacova, N. Growe-Kuska and W. Romisch, *Scenario reduction in stochastic programming : an approach using probability metrics*, Mathematical Programming, Series A 95, pp. 493–511, 2003.

[4] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[5] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 2 Vol., 3rd Ed., Athena Scientifi c, 2007.

[6] B. Defourny, L. Wehenkel and R. Sepulchre, *Inferring scenario trees for stochastic programming : A machine learning perspective*, 26th Benelux Meeting on Systems and Control, Lommel, Belgium, March 13–15, 2007.