What is a robust decision?

Georg Ch. Pflug

February 2014

Wikipedia: A wide variety of concepts, methods, and tools have been developed to address decision challenges that confront a large degree of uncertainty. Rosenhead was among the first to lay out a systematic decision framework for robust decisions (*Remark: flexible decisions*). Similar themes have emerged from the literatures on scenario planning, robust control, imprecise probability, and info-gap decision theory and methods. An early review of many of these approaches is contained in the Third Assessment Report of the Intergovernmental Panel on Climate Change.

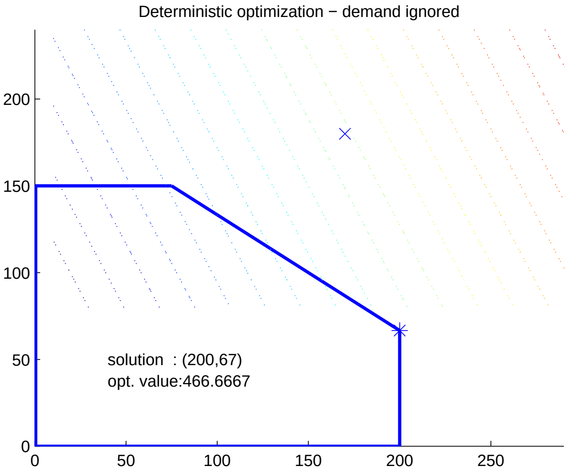
The basic Example

A producer has to determine his production plan for two types of goods. Let

x_1, x_2	the production quantity (decision variable)
$x_1 \le m_1, x_2 \le m_2$	individual capacity constraints
$a_1 x_1 + a_2 x_2 \le b$	joint capacity constraint
c_1, c_2	production cost per unit
s_1, s_2	selling price per unit

$$\text{Objective: } \max\{(s_1 - c_1)x_1 + (s_2 - c_2)x_2 : x \in \mathbb{X}\}$$

with $\mathbb{X} = \{(x_1, x_2) : x_1 \le m_1, x_2 \le m_2, a_1 x_1 + a_2 x_2 \le b\}$ being the feasible set.

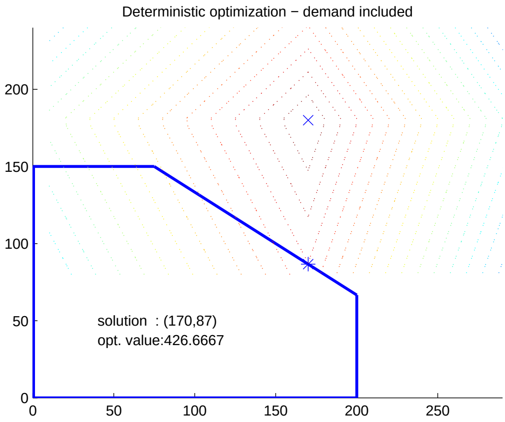

Deterministic optimization − demand ignored
200
150
100
50
0
0
50
100
150
200
250
solution : (200,67)
opt. value:466.6667

Adding a demand value

$$\mu_1, \mu_2 \quad \text{the deterministic demands}$$

Objective:

$$\max\{\min(x_1, \mu_1)s_1 + \min(x_2, \mu_2)s_2 - c_1 x_1 - c_2 x_2 : x \in \mathbb{X}\}$$

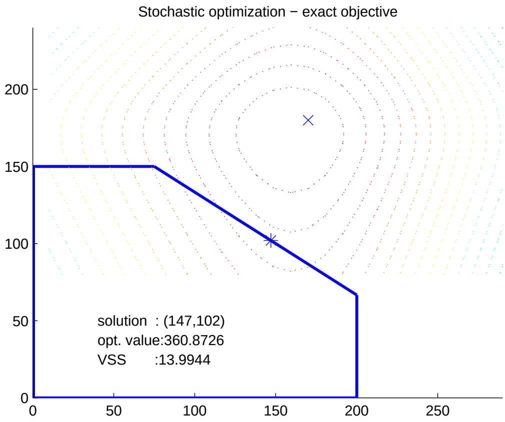


A stochastic demand

$$\xi_1 \sim N(\mu_1, \sigma_1^2), \xi_2 = N(\mu_2, \sigma_2^2) \quad \text{the random demands}$$

Objective:

$$\max\{\mathbb{E}[\min(x_1, \xi_1)s_1 + \min(x_2, \xi_2)s_2 - c_1 x_1 - c_2 x_2] : x \in \mathbb{X}\}$$



The value of the stochastic solution

Let (x_1^+, x_2^+) be the solution of the deterministic problem (with demand limits). It can be seen as a special case of the stochastic problem, where all random variables are in fact deterministic and equal to their expectation. The VSS value is

$$\begin{aligned} VSS &= \max\{\mathbb{E}[\min(x_1,\xi_1)s_1 + \min(x_2,\xi_2)s_2 - c_1x_1 - c_2x_2] : x \in \mathbb{X}\} \\ &- \mathbb{E}[\min(x_1^+,\xi_1)s_1 + \min(x_2^+,\xi_2)s_2 - c_1x_1^+ - c_2x_2^+] \end{aligned}$$

The clairvoyant's solution

The decision of the stochastic decision problem has to be made before the demand is observed (here-and-now). A clairvoyant would be able to make the decision after the observation of the demand (wait-and-see), i.e. he solves the following problem:

$$\begin{aligned} & (x_1(\xi), x_2(\xi)) \\ = \quad & \text{argmax}_{\,x_1,x_2}\{\min(x_1,\xi_1)s_1 + \min(x_2,\mu_2)s_2 - c_1x_1 - c_2x_2 : x \in \mathbb{X}\} \end{aligned}$$

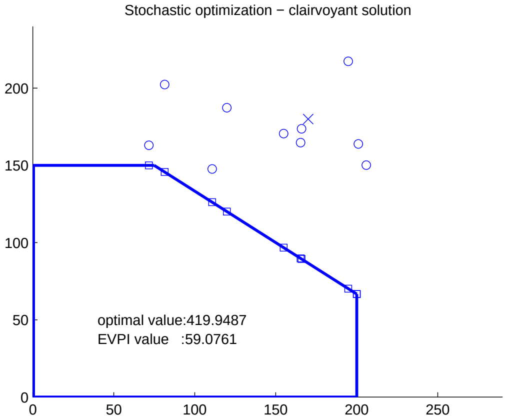
with $\xi = (\xi_1, \xi_2)$.
The clairvoyant's optimal solution is

$$\mathbb{E}[\min(x_1(\xi),\xi_1)s_1 + \min(x_2(\xi),\xi_2)s_2 - c_1x_1(\xi) - c_2x_2(\xi)]$$

The expected value of perfect information (EVPI)

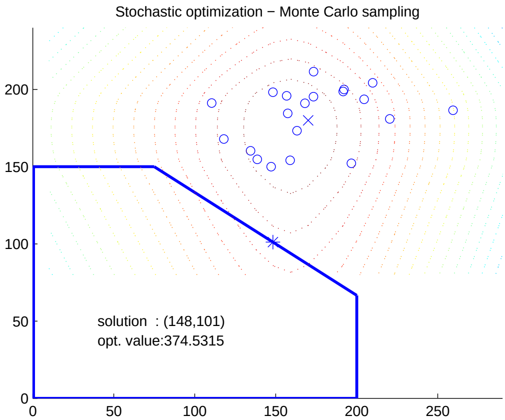
The EVPI value is:

$$\begin{aligned} EVPI &= \text{optimal value of the clairvoyant's problem} \\ &- \text{optimal value of the here-and-now problem} \end{aligned}$$

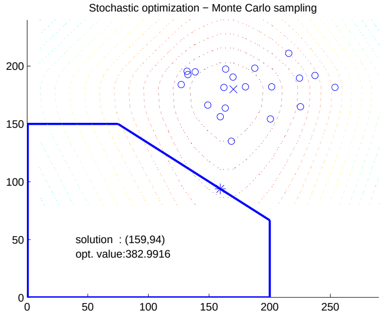


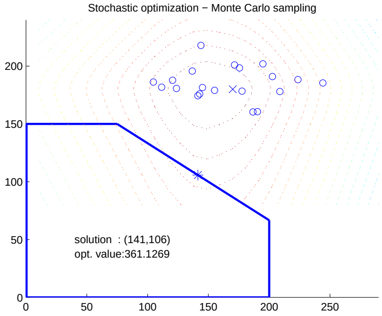
Approximation by MC sampling

Objective: $\max\{\sum_{i=1}^{n} p_i[\min(x_1, \xi_{1,i})s_1 + \min(x_2, \xi_{2,i})s_2] : x \in \mathbb{X}\}$



MC approximations give different solution in different runs

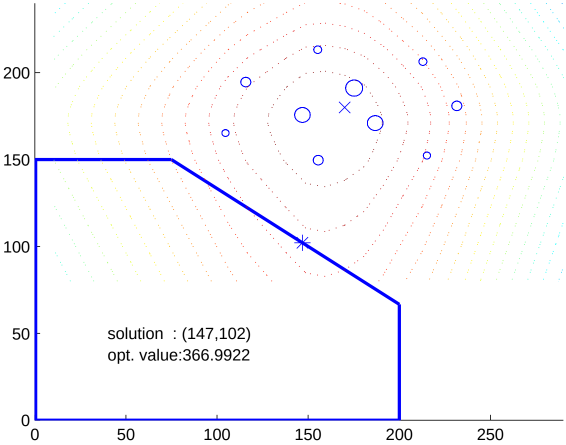




Approximation by quantization

Instead of sampling, one may use optimally placed weighted points, which are chosen in order to minimize a distance between them and the probability distribution of the problem.
To calculate optimal points is a nonlinear nonconvex optimization problem of its own. For the normal distributions, optimal points have been calculated by Gilles Pagès and published on his web-pages (the Pagès-pages).

Stochastic optimization − quantization



Robust optimization

An additional constraint: overproduction should be avoided. We add the constraint

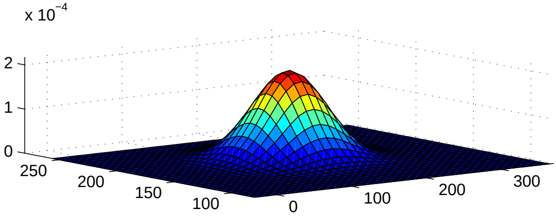
$$P\{x_1 > \xi_1 \text{ or } x_2 > \xi_2\} \leq 10\%$$

which is equivalent to

$$P\{x_1 \leq \xi_1 \text{ and } x_2 \leq \xi_2\} \geq 90\%$$

The scenario-robust method finds first an acceptance set A such that $P\{(\xi_1, \xi_2) \in A\} = 90\%$. and adds the constraints

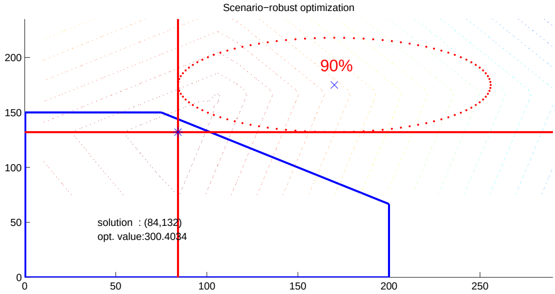
$$x_1 \leq z_1; x_2 \leq z_2 \text{ for all } (z_1, z_2) \in A$$



Scenario-robust optimization is very pessimistic

Objective:

$$\max\{\min_{(\xi_1,\xi_2)\in A_{1-\alpha}}[\min(x_1,\xi_1)s_1+\min(x_2,\xi_1)s_2] : x\in\mathbb{X}\}$$



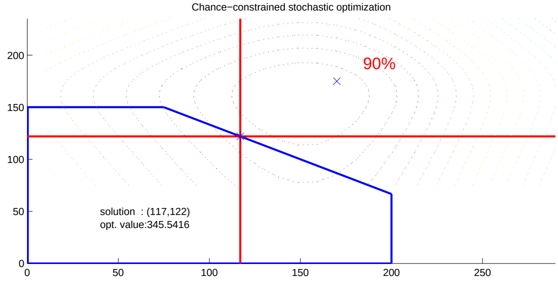
The wrong solution

Chance-constraint optimization is the correct way

The constraint

$$P\{x_1 \leq \xi_1 \text{ and } x_2 \leq \xi_2\} \geq 90\%$$

is added to the constraint set, but the acceptance set A may depend on the solution x and *is not chosen beforehand*.

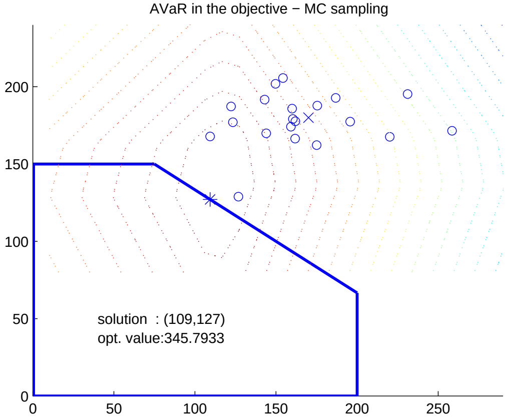


The correct solution

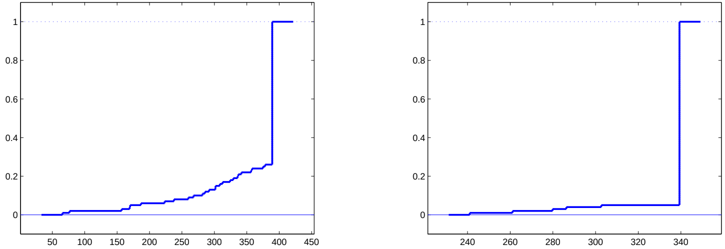
Including risk aversion

$$\mathbb{A}\text{V@R}_\alpha(Y) = \frac{1}{\alpha}\int_0^\alpha G_Y^{-1}(u)\,du \le \int_0^1 G_Y^{-1}(u)\,du = \mathbb{E}(Y)$$

Objective: $\max\{\mathbb{A}\text{V@R}_\alpha[\min(x_1,\xi_1)s_1 + \min(x_2,\xi_2)s_2] : x \in \mathbb{X}\}$



The influence of the risk objective to the profit distribution



Left: Objective = Expectation (risk neutral), Right: Objective = $\mathbb{A}$V@R (risk averse)

In the right picture, the downside risk is much smaller, but also the expected return is smaller.

Ambiguity

According to Ellsberg (1961) we face two types of non-determinism:

- *Uncertainty*: the probabilistic model is known, but the realizations of the random variables are unknown ("aleatoric uncertainty")
- *Ambiguity*: the probability model itself is not fully known ("epistemic uncertainty" - Knightian uncertainty according to F. Knight "Risk, Uncertainty and Profit" (1920)).

Coping with model error

In many applications, the model is identified based on data and therefore the problem is subject to *model error*. Instead of the estimated baseline model P, some other probability models Q may also be compatible with the data and describe the real phenomenon well.

Therefore we define an ambiguity set of models $\mathcal{P}$ and solve a *minimax ambiguity model*

$$\max \min_{Q \in \mathcal{P}} \{\mathbb{E}_Q[\min(x_1, \xi_1)s_1 + \min(x_2, \xi_2)s_2 - c_1x_1 - c_2x_2] : x \in \mathbb{X}\}.$$

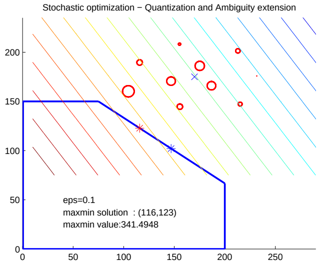
The ambiguity problem is a maximin problem and solved by algorithms for finding a saddlepoint (x^*, Q^*). Q^* is the *worst case model*.

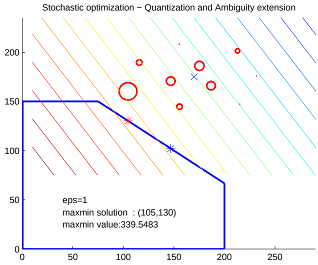
We choose typically the ambiguity set as a ball around the baseline model P:

$$\mathcal{P} = \{Q : d(P, Q) \leq \epsilon\}$$

where ϵ is the *ambiguity radius*.

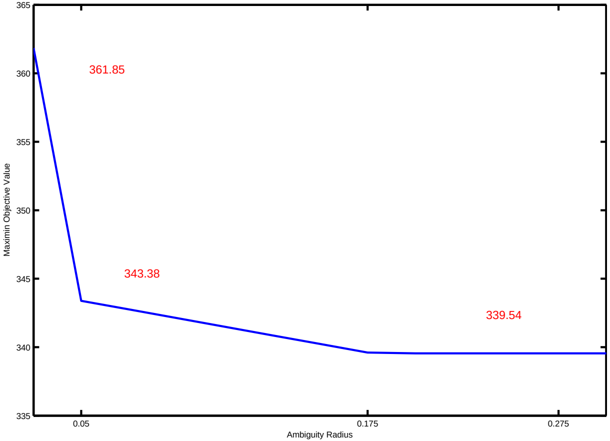
Ambiguity





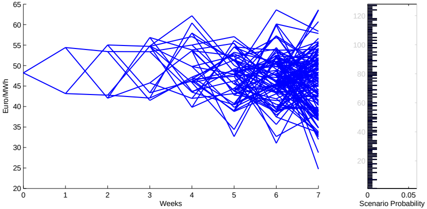
Left: Ambiguity radius $\epsilon = 0.1$ Right: Ambiguity radius $\epsilon = 1$

The price for distributional robustness

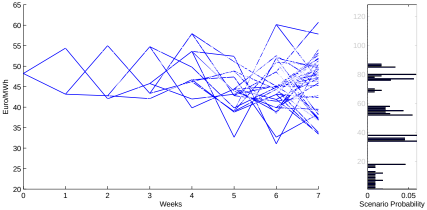


The drop in optimal value (the maximin value) as a function of the ambiguity radius ϵ.

Worst case scenario trees

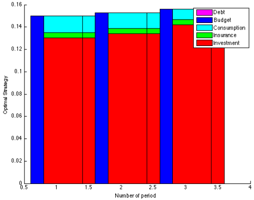


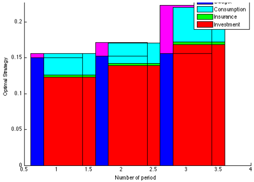
The original spotprice tree



The worst case spotprice tree

Examples of solutions





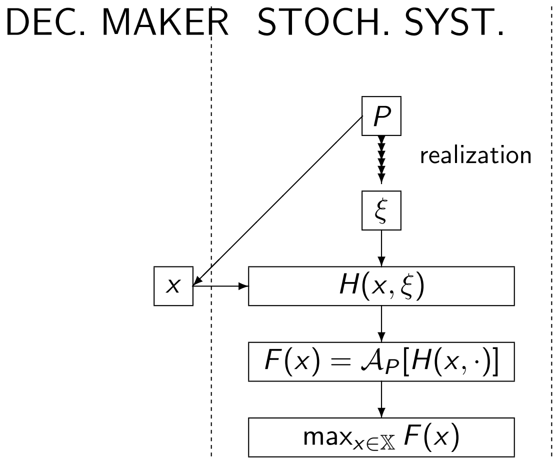
Sensitivity and Robustness

- If we find a best decision under a given model and test its quality under some variants of models, we speak about *sensitivity analysis*. How sensitive is the result under deviations of the model assumptions?
- If we find the best minimax decision, then this decision is acceptable under all considered deviations of the model assumptions. This is what *model robustness* (distributional robustness) is all about.

Summary

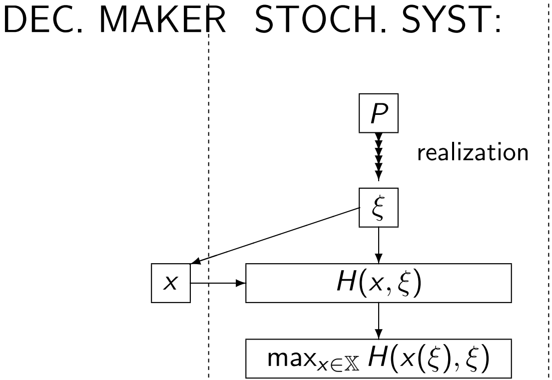
- Stochastic optimization allows to deal with uncertainties in decision parameters.
- The VSS and the EVPI values characterize the degree of "stochasticity" of the problem.
- To numerically solve a stochastic program, an approximation scheme, such as MC or quantization is typically required.
- Robust optimization chooses first a typical set of outcomes and looks then for the worst case, while for problems with probability constraints, the exception set is determined as part of the optimization.
- Risk functionals allow to formulate risk aversion.
- More sophisticated decision problems are stochastic multi-stage.
- Model uncertainty (ambiguity) requires distributionally robust decisions, which can be found by solving maximin problems.

Stochastic optimization: here and now



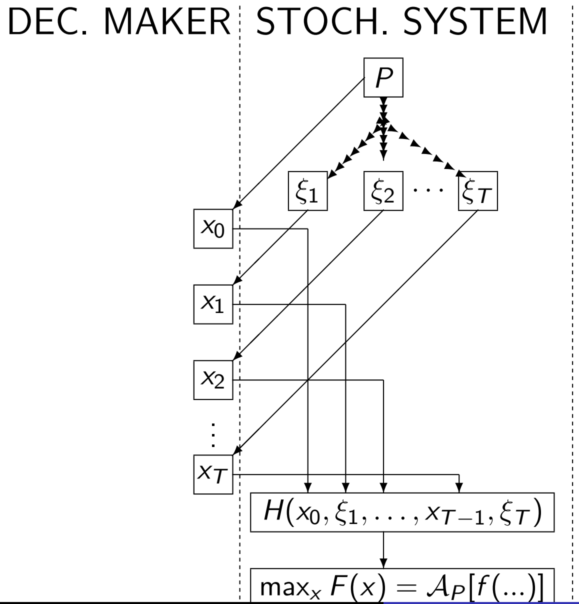
$\mathcal{A}_P$ is an utility functional, like expectation, risk corrected expectation or a distortion functional, e.g. the average value-at-risk.

Stochastic optimization: wait and see

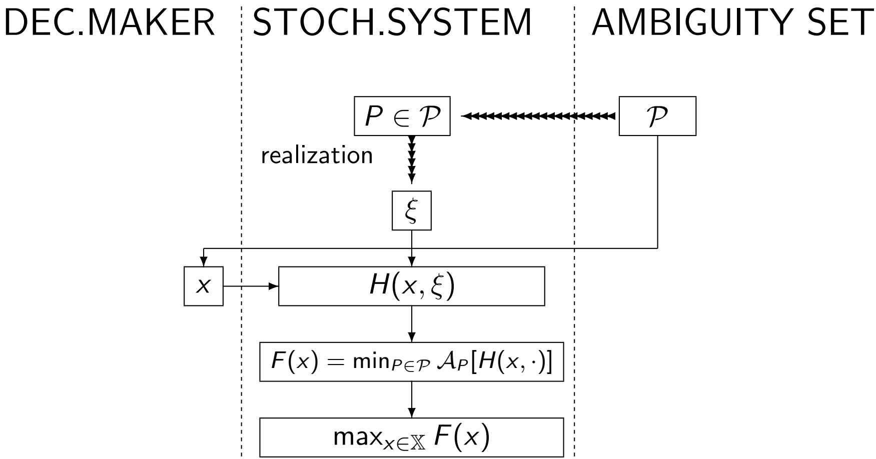


This problem decomposes scenariowise.

Multistage stochastic optimization



Stochastic optimization under ambiguity



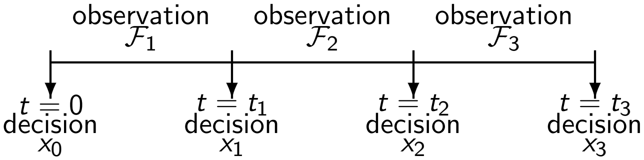
Formulation of a multistage stochastic program

$$\text{Problem } (\mathcal{P}): \; \max\{\mathcal{A}[H(x_0, \xi_1, \ldots, x_{T-1}, \xi_T)] : x \lhd \mathfrak{F}\},$$

where

$\xi = (\xi_1, \ldots, \xi_T)$	a random scenario process, on a probability space $(\Omega, \mathcal{F}, P)$
$x = (x_0, \ldots, x_{T-1})$	the sequence of decisions,
$H(x_0, \xi_1, \ldots, \xi_T)$	the profit function,
$\mathcal{A}$	a version-independent acceptability functional, such as the expectation $\mathbb{E}$
$\mathfrak{F} = (\mathcal{F}_1, \ldots, \mathcal{F}_T)$	a filtration (an increasing sequence of σ-algebras) on $(\Omega, \mathcal{F}, P)$
$\xi \lhd \mathfrak{F}$	ξ is adapted to $\mathfrak{F}$, i.e. $\sigma(\xi) \subseteq \mathfrak{F}$
$x \lhd \mathfrak{F}$	the nonanticipativity condition.

Non-anticipativity



Types of stochastic programs

- If the functional is the expectation, we call it a risk-neutral problem
- If the functional models the acceptability of risk, it is called a risk-sensitive problem
- Some stochastic programs are formulated with the help of multiperiod functionals:

$$\max\{\mathcal{A}[H_1(x_0, \xi_1); H_2(x_1, \xi_2); \ldots, H(x_{T-1}, \xi_T)]$$

- If the probability of an event is constrained, we call it a chance-constrained problem
- Some stochastic programs (but not all) allow formulation as a dynamic program
- If the stochastic program is linear, we distinguish between models with random right hand side, random costs and random technology matrix.

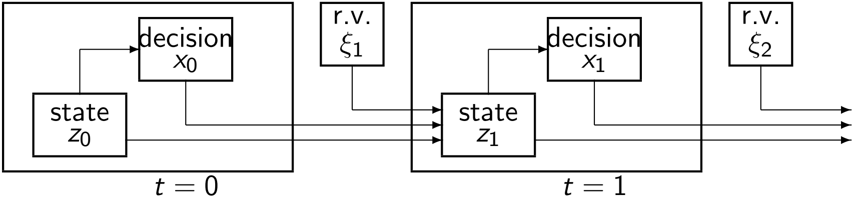
Solution through finite approximation

Instead of the process $\xi = (\xi_1, \ldots, \xi_T)$ and the filtration $\mathfrak{F} = (\mathcal{F}_1, \ldots, \mathcal{F}_T)$ we consider a simpler process $\tilde{\xi} = (\tilde{\xi}_1, \ldots, \tilde{\xi}_T)$ and the filtration $\tilde{\mathfrak{F}} = (\tilde{\mathcal{F}}_1, \ldots, \tilde{\mathcal{F}}_T)$.
Filtrations on a finite probability spaces are equivalent to trees (of subsets).
The basic problem is replaced by the simpler problem

$$\text{Problem } (\tilde{\mathcal{P}}): \ \max\{\mathcal{A}[H(x_0, \tilde{\xi}_1, \ldots, x_{T-1}, \tilde{\xi}_T)] : x \lhd \tilde{\mathfrak{F}}\},$$

$- >$ tree structured problem!

Dynamic optimization



The decision dynamics

We may represent the multistage dynamic decision model as a state-space model. We assume that there is a state vector ζ_t, which describes the situation of the decision maker at time t immediately before he must make the decision x_t, for each $t = 1, \ldots, T$, and its realization is denoted by z_t. The initial state $\zeta_0 = z_0$, which precedes the deterministic decision at time 0, is known and given by ξ_0. To assume the existence of such a state vector is no restriction at all, since we may always take the whole observed past and the decisions already made as the state:

$$\zeta_t = (x^{t-1}, \xi^t), \qquad t = 1, \ldots, T.$$

Here $x^t = (x_1, \ldots, x_t)$ and $\xi^t = (\xi_1, \ldots, \xi_t)$. However, the vector of required necessary information for the future decisions is often much shorter.

The state variable process $\zeta = (\zeta_1, \ldots, \zeta_T)$, with realizations $z = (z_1, \ldots, z_T)$, is a controlled stochastic process, which takes values in a state space $Z = Z_1 \times \cdots \times Z_T$. The control variables are the decisions x_t, $t = 1, \ldots, T$. The state ζ_t at time t depends on the previous state ζ_{t-1}, the decision x_{t-1} following it, and the last observed scenario history ξ^t. A transition function g_t describes the state dynamics:

$$\zeta_t = g_t(\zeta_{t-1}, x_{t-1}, \xi^t), \qquad t = 1, \ldots, T. \tag{1}$$

At the terminal stage T, no decisions are made, only the outcome $\zeta_T = z_T$ is observed.

Note that ζ_t is a random variable with realization z_t, which is a function of the realization of ξ^t, for each $t = 1, \ldots, T$.
The decision x of the multistage stochastic problem is a vector of functions $x = (x_0, \ldots, x_{T-1})$, $t = 1, \ldots, T-1$, maps Ξ^t to $\mathbb{R}^{m_t}$.
We require that the feasible decision x_t at time t satisfies a constraint of the form

$$x_t \in \mathcal{X}_t(\zeta_t), \qquad t = 1, \ldots, T,$$

where $\mathcal{X}_t$ are closed convex multifunctions with closed convex values.

Bellmann equation

If the problem decomposes into a sum of profit functions and the probability functional is the expectation, one may use a stage-wise decomposition.
For each stage, one defines a value function $V_t(z_t)$. Notice that z_t has to carry all relevant information for the future.
Backward equation:

$$V_t(z_t) = \max_{x_t \in \mathcal{X}_t} \mathbb{E}[H_t(x_t, \zeta_t, \xi_{t+1}) + V_{t+1}(\zeta_{t+1})]$$

with

$$\zeta_{t+1} = g_{t+1}(\zeta_t, x_t, \xi^{t+1}).$$

Typically the functions $V_t(z)$ are discretized by using linear, quadratic or cubic interpolations between finitely many support points.

Benders decomposition

$$\left\| \begin{array}{l} \text{Minimize } f(x) + g(y) \\ \text{s.t.} \\ Tx + Ay \geq b \\ x \in S \\ y \geq 0 \end{array} \right.$$

Here x and y are vectors and T and A are matrices of appropriate sizes. We call x the first stage and y the second stage variables. Let $\mathcal{R} = \{x : \exists y \geq 0 \text{ such that } Tx + Ay \geq b\}$ be the implied feasibility set. $\mathcal{R}$ is a convex polyhedron. We will find representation for $\mathcal{R}$.

Let $\mathcal{H} = \{h : \exists y \geq 0 \text{ such that } Ay \geq h\}$. $\mathcal{H}$ is also a convex polyhedron. $h \in \mathcal{H}$ implies that the following linear program is feasible

$$\left\| \begin{array}{l} \text{Minimize } 0^\top y \\ \text{s.t.} \\ Ay \geq h \\ y \geq 0 \end{array} \right.$$

and its optimal value is 0, which is equivalent to the fact that its dual is bounded

$$\left\| \begin{array}{l} \text{Maximize } h^\top u \\ \text{s.t.} \\ A^\top u \leq 0 \\ u \geq 0 \end{array} \right.$$

and its optimal value is also 0.

This shows that $\mathcal{H}$ can be represented in the following way: Let $\mathcal{A} = \{u | A^\top u \leq 0, u \geq 0\}$. Then $\mathcal{H} = \{h : h^\top u \leq 0 : u \in \mathcal{A}\}$. Let $(u_i)_{i \in I}$ be the set of extremals of $\mathcal{A}$. Then $\mathcal{H} = \{h : h^\top u_i \leq 0 : i \in I\}$ and finally $\mathcal{R} = \{x : b - Tx \in \mathcal{H}\} = \{x : x^\top T^\top u_i \geq b^\top u_i : i \in I\}$. Let $w_i = T^\top u_i$ and $\nu_i = b^\top u_i$ and let W be the matrix with rows $w_i^\top$ and v be the vector with components v_i. Then

$$\mathcal{R} = \{x : Wx \geq v\}.$$

Let further

$$Q(x) = \min\{g(y) : Ay \geq b - Tx; y \geq 0\}.$$

For $x \notin \mathcal{R}$, we set $Q(x) = \infty$.
If g is convex, then $Q_1(h) = \min\{g(y) : Ay \geq h; y \geq 0\}$ is convex on $\mathcal{H}$ and therefore $Q(x) = \min\{g(y) : Ay \geq b - Tx; y \geq 0\}$ is convex on $\mathcal{R}$. As every convex function, Q coincides with the maximum of (possibly infinitely many) linear functions on R.

If g is linear, say $g(y) = d^\top y + \gamma$, then Q is a convex, piecewise linear function. To see this, let $Q_1(h) = \min\{d^\top y + \gamma : Ay \geq h; y \geq 0\}$. By duality, $Q_1(h) = \gamma + \max\{z^\top h : A^\top z \leq d; z \geq 0\}$. The polyhedron $\{z : A^\top z \leq d; z \geq 0\}$ is the convex hull of its extremals $(z_k)_{k \in K}$. Therefore $Q(x) = \gamma + \max\{-z_k^\top Tx + z_k^\top b : k \in K\}$. Let $r_k = -T^\top z_i$ and $\tau_k = z_k^\top b$. Then

$$Q(x) = \gamma + \max\{r_k^\top x + \tau_k : k \in K\}.$$

If g is the maximum of linear functions, say $g(y) = \max\{d_\ell^\top y + \gamma_\ell : \ell \in L\}$, then $Q_1(h) = \min\{\xi : \xi \geq d_\ell^\top y + \gamma_\ell\} : Ay \geq h; y \geq 0\}$. Let D be the matrix consisting of the rows d_ℓ and g the vector with components γ_ℓ. By duality, $Q_1(h) = \max\{z^\top h + v^\top g : A^\top z \leq Dv; v^\top \mathbb{1} = 1; v \geq 0; z \geq 0\}$. The polyhedron $\{(v, z) : A'z \leq Dv; v^\top \mathbb{1} = 1; v \geq 0; z \geq 0\}$ is the convex hull of its extremals $(v_k, z_k)_{k \in K}$. Let $r_k = -T^\top z_i$ and $\tau_k = z_k^\top b + v_k^\top g$. Then

$$Q(x) = \max\{r_k^\top x + \tau_k : k \in K\}.$$

Therefore $Q(x) = \max\{-z_k^\top Tx + z_k^\top b : k \in K\}$.

We call $\mathcal{R}^{(s)}$ an outer approximant of $\mathcal{R}$, if it consists only of a subset of constraints of $\mathcal{R}$. We call $Q^{(s)}$ a lower approximant of Q, if it is the maximum of a subset of the functions appearing in Q.
Here is the structure of the algorithm.
Let $\mathcal{R}^{(1)}$, $Q^{(1)}$ be some simple approximants of $\mathcal{R}$ and Q.

1. Set $s := 1$.
2. [Master] Solve $\min\{f(x) + Q^{(s)}(x) : x \in S, x \in \mathcal{R}^{(s)}\}$. Send x and $Q^{(s)}$ to slave.
3. [Slave] Solve $\min\{g(y) : Ay \geq b - Tx; y \geq 0\}$.
   If this program is feasible then $x \in \mathcal{R}$. If moreover $Q(x) = Q^s(x)$, we have found a solution and stop.
   3.1 If $y \notin \mathcal{R}$, then we have to find a constraint $w_i x \geq \nu_i$, which is not satisfied. Send this constraint (w_i, ν_i) ("feasibility cut") to the master.
   3.2 If $y \in \mathcal{R}$, but $Q^{(s)}(x) < Q(x)$, then we calculate a supporting hyperplane to Q in the point x: Let $r \in \partial Q(x)$ be a subgradient of Q at x and let $\tau = Q(x) - r'x$. Then for all z, $Q(z) \geq r'z + \tau$, with equality for $z = x$. Send r and τ ("optimality cut") to the master.
4. [Master] The master adds the feasibility cut to R^s to get $R^{(s+1)} = R^{(s)} \cap \{x : w_i x \geq \nu_i\}$ and/or updates $Q^{(s)}$ using the optimality cut $Q^{(s+1)} = \max[Q^{(s)}, r'x + \tau]$.
5. goto 2.

Since we add always new cuts and do not forget old ones and since R is a polyhedron with finitely many extremals and Q is convex piecewise linear, we find a solution in finitely many steps.
The calculation of feasibility cuts: Suppose that there is no $y \geq 0$ such that $Ay \geq b - Tx$. This means that the problem

$$\left\| \begin{array}{l} \text{Minimize } 0'y \\ \text{s.t.} \\ Ay \geq b - Tx \\ y \geq 0 \end{array} \right.$$

is infeasible and hence its dual

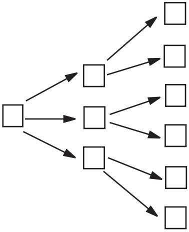
$$\left\| \begin{array}{l} \text{Maximize } (b - Tx)'u \\ \text{s.t.} \\ A'u \leq 0 \\ u \geq 0 \end{array} \right.$$

is unbounded. One may identify an extreme ray u, along which the problem is unbounded, i.e. $u \geq 0$, $A'u \leq 0$, $(b - Tx)'u \geq 0$, $(b - Tx)'u \neq 0$. Let $w = T'u$, $\nu = b'u$. The pair (w, ν) gives the new constraint $w'x \geq \nu$ for $\mathcal{R}$.

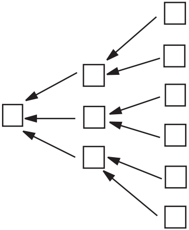
The calculation of optimality cuts: Solving $\min\{g(y) : Ay \geq b - Tx; y \geq 0\}$ leads to a pair (y, λ) of primal and dual solutions. The subgradient $r \in \partial Q(x)$ is $r = -T'\lambda$. The very same procedure applies to convex, nonlinear f and g with the only difference, that the procedure does not terminate in finitely many steps.

The same algorithm applies to the tree structured case, where the second stage decomposes completely into the sum of J functions:

$$\begin{array}{|l} \text{Minimize } f(x) + \sum_{j=1}^{J} g_j(y_j) \\ \text{s.t.} \\ T_j x + A_j y_j \geq b_j \\ x \in S \\ y_j \geq 0 \qquad \text{for all } j \end{array}$$



flow of primal information
(solutions passed to successor nodes)



flow of dual information
(feasibility and optimality cuts passed to predecessor nodes)

This problem is is solved by one master program and J slave programs: Each of the J slaves is responsible for one of the functions g_j. Notice that these functions are only linked through the first stage variables x. Therefore, the slave optimizations can be run in parallel: The master uses approximations $\mathcal{R}^{(s)}$ and $Q_j^{(s)}$ of the functions $Q_j(x) = \min\{g_j(y) : A_j y \geq b - T_j x; y \geq 0\}$. It solves $\min\{f(x) + \sum_{j=1}^J Q_j^{(s)}(x) : x \in S, x \in \cap_j \mathcal{R}_j^{(s)}\}$. It sends x and $Q_j^{(s)}$ to slave s and receives from him feasibility cuts for $\mathcal{R}_j$ and Q_j.

In the tree structure, every node which is not the root and not a leaf is at the same time master and slave. The root is only master, the leaf nodes are only slaves. Each nonterminal node n must optimize vector x_n of local decisions using its local constraints $x_n \in S_n$, the implied feasibility constraints $x_n \in \cap_{j \in n+} \mathcal{R}_j$. The function to be optimized is the sum of a local objective and the optimal value functions of the successor nodes $f_n(x_n) + \sum_{j \in n+} Q_j(x_n)$.

SDDP- Pereira

The SDDP approach is identical to the Benders decomposition, but does not construct the scenario tree right from the beginning, but uses values of the process which are sampled in during the procedure.
If the stochastic process ξ is independent, then this method is very efficient, since no conditional distributions appear. However, the independence assumption is very questionable and to modify the process in such a way that only independent innovations appear is not always possible.

Research at the University of Vienna

- Quality of approximation of stochastic programs (nested distance)
- Optimal approximations of stochastic processes by trees
- Properties of risk functionals, time consistency
- Ambiguity models
- Stochastic bilevel problems
- Energy and insurance models