Stochastic Optimal Control Problems
- Part III: Some numerical aspects

Hasnaa Zidani[1]

[1]Ensta ParisTech, University Paris-Saclay

Thematic trimester "SVAN", IMPA, 2016

$$\text{Find } x \in X, \quad \max_{\beta \in \mathcal{B}} \min_{\alpha \in \mathcal{A}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

$$\text{Find } x \in X, \quad \max_{\beta\in\mathcal{B}} \min_{\alpha\in\mathcal{A}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

✎ $X = \mathbb{R}^N$ (or $X = \mathbb{R}^{\mathbb{N}}$)

✎ For $f, g \in X$, $\min(f, g) = (\min(f_i, g_i))_{i=1,\cdots,N}$.

$$\text{Find } x \in X, \quad \max_{\beta \in \mathcal{B}} \min_{\alpha \in \mathcal{A}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

- ✎ $X = \mathbb{R}^N$ (or $X = \mathbb{R}^{\mathbb{N}}$)
- ✎ For $f, g \in X$, $\min(f, g) = (\min(f_i, g_i))_{i=1,\cdots,N}$.
- ✎ $\mathcal{A}$ and $\mathcal{B}$ are compact sets of metric spaces

$$\text{Find } x \in X, \quad \max_{\beta \in \mathcal{B}} \min_{\alpha \in \mathcal{A}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

- ✎ $X = \mathbb{R}^N$ (or $X = \mathbb{R}^{\mathbb{N}}$)
- ✎ For $f, g \in X$, $\min(f, g) = (\min(f_i, g_i))_{i=1,\cdots,N}$.
- ✎ $\mathcal{A}$ and $\mathcal{B}$ are compact sets of metric spaces
- ✎ For every $(\alpha, \beta) \in \mathcal{A} \times \mathcal{B}$, the matrix $A^{\alpha,\beta}$ and vector $b^{\alpha,\beta} \in X$ are known.

$$\text{Find } x \in X, \quad \min_{\alpha\in\mathcal{A}} \max_{\beta\in\mathcal{B}} (A^{\alpha,\beta}x - b^{\alpha,\beta}) = 0,$$

Example 1: Obstacle Problem (OP)

$$\text{Find } x \in \mathbb{R}^N, \ \min(Qx - b, \ \mathbf{x} - \mathbf{g}) = 0$$

Example 1: Obstacle Problem (OP)

$$\text{Find } x \in \mathbb{R}^N, \ \min(Qx - b, \ \mathbf{x} - \mathbf{g}) = 0$$

- If $Q \geq 0$ sym., (OP) is equivalent to

$$\left| \begin{array}{l} \text{Minimize} \quad \frac{1}{2}(Qx, x) - (b, x) \\ x \in \mathbb{R}^N \text{ and } \mathbf{x} \geq \mathbf{g} \end{array} \right.$$

Example 1: Obstacle Problem (OP)

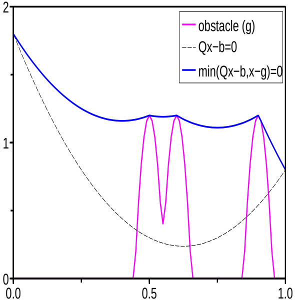
$$\text{Find } x \in \mathbb{R}^N, \ \min(Qx - b, \ \mathbf{x} - \mathbf{g}) = 0$$

- If $Q \geq 0$ sym., (OP) is equivalent to

$$\left| \begin{array}{l} \text{Minimize} \quad \frac{1}{2}(Qx, x) - (b, x) \\ x \in \mathbb{R}^N \text{ and } \mathbf{x} \geq \mathbf{g} \end{array} \right.$$

- Variational inequality:

$$\left| \begin{array}{l} \min(-\Delta u(s) - f(s), u(s) - g(s)) = 0 \\ \qquad\qquad \text{a.e. } \ s \in (0, 1), \\ u(0) = u_g, u(1) = u_d \end{array} \right.$$

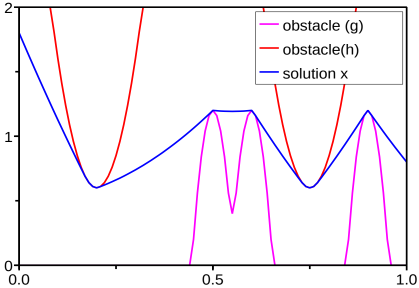


Example 2: Double Obstacle Problem (DOP)

$$\text{Find } x \in \mathbb{R}^N, \ \max(\min(Qx - b, x - g), x - h) = 0$$

- If $Q \geq 0$ sym., (DOP) is equivalent to

$$\left| \begin{array}{l} \text{Minimize} \quad \frac{1}{2}(Qx, x) - (b, x) \\ x \in \mathbb{R}^N \text{ and } \mathbf{h} \geq \mathbf{x} \geq \mathbf{g} \end{array} \right.$$



- (OP) is equivalent to solve

$$\min(A^0 x - b^0,\ A^1 x - b^1) = 0,$$

with $A^0 := Q, \quad b^0 := b \qquad$ and $\quad A^1 := I_d \quad b^1 := g.$

➤ (OP) is equivalent to solve

$$\min_{\alpha\in\{0,1\}} (A^\alpha x - b^\alpha) = 0,$$

with $A^0 := Q, \quad b^0 := b \qquad \text{and} \quad A^1 := I_d \quad b^1 := g.$

- (OP) is equivalent to solve

$$\min_{\alpha \in \{0,1\}} (A^\alpha x - b^\alpha) = 0,$$

with $A^0 := Q, \quad b^0 := b \qquad \text{and} \quad A^1 := I_d \quad b^1 := g.$

- In the same way, (DOP) is equivalent to: Find $x \in \mathbb{R}^N$,

$$\max_{\beta \in \{0,1\}} \min_{\alpha \in \{0,1\}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

$$\text{with} \qquad \begin{aligned} & A^{0,0} := Q, \quad b^{0,0} := b \\ & A^{1,0} := I_d, \qquad b^{1,0} := g \\ & A^{0,1} = A^{1,1} := I_d \qquad b^{0,1} = b^{1,1} := h. \end{aligned}$$

Example 3: Stochastic Path Problems

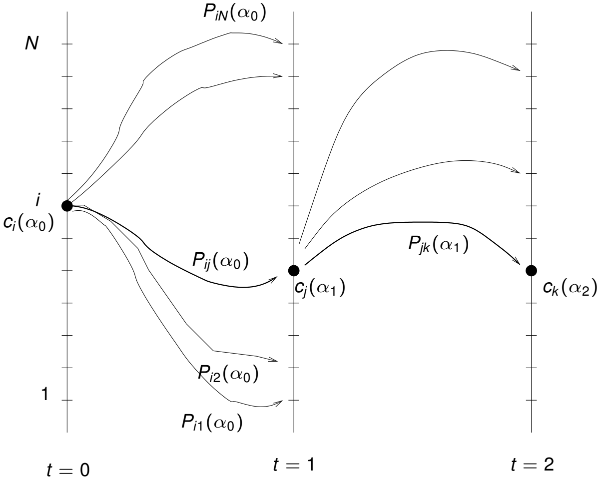
Bertsekas, Tsitsiklis, Kushner, Shiryaev, Quadrat, ...

- Consider a familty of N states denoted $(\xi_l)_{l=1,\cdots,N}$.
- Consider the set of admissible policies:

$$\mathcal{A}_{\mathtt{ad}} := \{\alpha = (\alpha_1, \cdot, \alpha_t, \cdots) \mid \alpha_t \in U\},$$

where U is a compact set of $\mathbb{R}^m$.

- $P(\alpha)$: the transition probability matrix corresponding to $\alpha \in U$, that is the matrix with elements $[P(\alpha)]_{ij} = p_{ij}(\alpha)$.
- Let also denote $c(\alpha)$ the vector of expected costs $c_i(\alpha)$, at node ξ_i, corresponding to the policy α.



$$c_i(\alpha_0) \qquad + \sum_j P_{ij}(\alpha_0) c_j(\alpha_1) \qquad + \sum_{j,k} P_{ij}(\alpha_0) P_{jk}(\alpha_1) c_k(\alpha_2)$$

- The expected cost corresponding to a policy $\alpha = \{\alpha_0, \alpha_1, \cdots\} \in \mathcal{A}_{\text{ad}}$ is given by:

$$W(\alpha) = \sum_{t=1}^{\infty} \frac{1}{(1+\lambda)^{t+1}} \left[P(\alpha_0)P(\alpha_1)\cdots P(\alpha_{t-1})\right] c(\alpha_t),$$

  where $W(\alpha) \in \mathbb{R}^N$.

- The optimal expected cost is:

$$V = \min_{\alpha \in \mathcal{A}_{\text{ad}}} W(\alpha).$$

- The Bellman principle yields to:

$$(1+\lambda)V = \min_{\alpha \in U}[c(\alpha) + P(\alpha)V].$$

Example 4: Two-person game

- Let us consider the discrete-time system (ϵ is fixed)

$$y_{k+1} = y_k + \sqrt{2\epsilon} b v_k, \quad k \geq 0, \qquad \text{with } y_0 = \xi$$

- Let Ω a convex set of $\mathbb{R}^2$, and $\mathcal{T}$ its boundary.

- We assume that we have two opponent players.
  - Player 1 (the evader) starts from ξ, and his goal is to reach the target $\mathcal{T}$.
  - Player 2 (the pursuer) is trying to obstruct him.

  The rules of the game are simple. At each timestep:

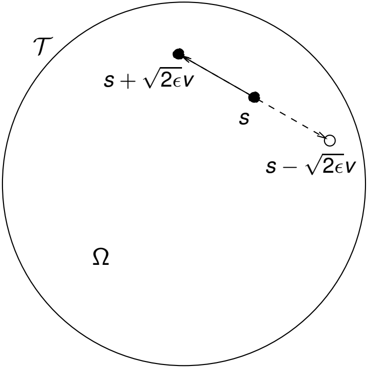
  1. Player 1 chooses a vector $v \in \mathbb{R}^2$ with $\|v\| = 1$.
  2. Player 2 chooses $b = \pm 1$ and replaces v with bv.

- Each step of the game costs ϵ.

We consider the payoff

$$\vartheta(\xi) := \begin{cases} k\varepsilon \text{ if Player 1 needs } k \text{ steps to reach } \mathcal{T}, \\ \text{starting from } \xi \text{ and following an optimal strategy.} \end{cases}$$

$$\Longrightarrow \begin{cases} \vartheta(\xi) = \min_{\|v\|=1} \max_{b=\pm 1} \left(\epsilon + \vartheta(\xi + \sqrt{2\epsilon}\, bv)\right), & \xi \in \Omega \\ \vartheta(\xi) = 0, \quad s \notin \Omega \end{cases}$$



- Consider $(\xi_i)_{i=1,\cdots,N}$: a grid on Ω.
- Consider the scheme

$$V_i = \min_{\|v\|=1} \max_{b=\pm 1} \left(\epsilon + [V](\xi_i + \sqrt{2\epsilon}\, bv)\right), \quad 1 \le i \le N$$

with V_i stands for an approximation of $\vartheta(\xi_i)$, and $[V]$ an interpolation of $(V_i)_{i=1,\cdots,N}$ on Ω:

$$\left\| \begin{array}{l} [V](\xi_i + \sqrt{2\epsilon}\, bv) = (P^{b,v} V)_i \\ [V](\xi) = 0, \text{whenever } \xi \notin \Omega \end{array} \right.$$

$(P^{b,v}_{ij} \ge 0$ and $\sum_j P^{b,v}_{ij} = 1$ or < 1 for border points).

- Final discrete equation:

$$V = \min_{\|v\|=1} \max_{b=\pm 1} \left(\epsilon + P^{b,v} V\right), \quad U \in \mathbb{R}^N.$$

- Remark: This model is related to front propagation with mean curvature motion Ref: Kohn-Serfaty

Example 5: Infinte Horizon Control problem

- Consider the OCP:

$$\vartheta(x) = \begin{cases} \min \displaystyle\sum_{j=0}^{\infty} (1-\lambda)^j \ell(y_j, u_j); \\ y_{j+1} = f(y_j, u_j), \quad y_0 = x, \\ u_j \in U \ \forall\, j \in \mathbb{N}, \end{cases}$$

where f and ℓ are Lipsch. continuous functions, and U is a compact set.

Example 5: Infinte Horizon Control problem

- Consider the OCP:

$$\vartheta(x) = \begin{cases} \min \displaystyle\sum_{j=0}^{\infty} (1-\lambda)^j \ell(y_j, u_j); \\ y_{j+1} = f(y_j, u_j), \quad y_0 = x, \\ u_j \in U \;\forall\, j \in \mathbb{N}, \end{cases}$$

where f and ℓ are Lipsch. continuous functions, and U is a compact set.

- The Dynamic Programming Principle gives:

$$\vartheta(x) = \min_{u \in U} \left\{ \ell(x, u) + (1-\lambda)\vartheta(f(x, u)) \right\}.$$

- Consider a uniform grid $\mathcal{G}$ with a constant mesh size. By ξ_i, we denote the nodes of $\mathcal{G}$.

- Consider a uniform grid $\mathcal{G}$ with a constant mesh size. By ξ_i, we denote the nodes of $\mathcal{G}$.
- An approximation of the DPP on $\mathcal{G}$ is obtained as:

$$(\vartheta(\xi_i) \simeq) V_i = \min_{u \in U} \{\ell(\xi_i, u) + (1 - \lambda)[V](f(x_i, u))\}.$$

- Consider a uniform grid $\mathcal{G}$ with a constant mesh size. By ξ_i, we denote the nodes of $\mathcal{G}$.
- An approximation of the DPP on $\mathcal{G}$ is obtained as:

$$(\vartheta(\xi_i) \simeq) V_i = \min_{u \in U} \{\ell(\xi_i, u) + (1-\lambda)[V](f(x_i, u))\} .$$

- Let μ_{ij}^u positive coefficients such that:

$$0 \leq \mu_{ij}^u \leq 1; \quad \sum_{j \geq 0} \mu_{ij}^u = 1;$$

$$f(\xi_i, u) = \sum_{j \geq 0} \mu_{ij}^u \xi_j .$$

- Consider a uniform grid $\mathcal{G}$ with a constant mesh size. By ξ_i, we denote the nodes of $\mathcal{G}$.
- An approximation of the DPP on $\mathcal{G}$ is obtained as:

$$(\vartheta(\xi_i) \simeq) V_i = \min_{u \in U} \{\ell(\xi_i, u) + (1-\lambda)[V](f(x_i, u))\}.$$

- Let μ_{ij}^u positive coefficients such that:

$$0 \leq \mu_{ij}^u \leq 1; \quad \sum_{j \geq 0} \mu_{ij}^u = 1;$$
$$f(\xi_i, u) = \sum_{j \geq 0} \mu_{ij}^u \xi_j.$$

- Set M^u the matrix with coefficients $M_{ij}^u = \mu_{ij}^u$. The DPP can be re-written as:

$$V = \min_{u \in U} \{L(u) + (1-\lambda) M^u V\},$$

where $L(u)$ is the vector with coefficients $\ell(\xi_i, u)$.

Example 6: Irreversible investment models

- Consider the SOCP :

$$\vartheta(x) = \begin{cases} \max \quad \mathbb{E}\Big[\sum_{t=0}^{\infty}(1-\lambda)^j(C(X_t) - \beta u_t)\Big]; \\ X_{t+1} = (1-\delta)X_t + u_t + \omega_t \sigma X_t, \quad X_0 = x, \\ u_t \in U \;\forall\; t \in \mathbb{N}, \end{cases}$$

Example 6: Irreversible investment models

➤ Consider the SOCP :

$$\vartheta(x) = \begin{cases} \max \quad \mathbb{E}\Big[\sum_{t=0}^{\infty}(1-\lambda)^j(C(X_t) - \beta u_t)\Big]; \\ X_{t+1} = (1-\delta)X_t + u_t + \omega_t\sigma X_t, \quad X_0 = x, \\ u_t \in U \; \forall\, t \in \mathbb{N}, \end{cases}$$

- X_t is the generating capacity of firm at time t
- u_t is the number of capital unit acquired by the firm at a cost βu_t where $\beta > 0$ is interpreted as a conversion factor,
- $\delta > 0$ is the depreciation rate of production, and σ its volatilities.
- The random variable ω_t takes values ± 1 with probability $\frac{1}{2}$.
- The profit function $C : \mathbb{R} \to \mathbb{R}$ is concave and increasing.

Example 6: Irreversible investment models

- Consider the SOCP :

$$\vartheta(x) = \begin{cases} \max \quad \mathbb{E}\Big[\sum_{t=0}^{\infty}(1-\lambda)^j(C(X_t)-\beta u_t)\Big]; \\ X_{t+1} = (1-\delta)X_t + u_t + \omega_t \sigma X_t, \quad X_0 = x, \\ u_t \in U \; \forall \; t \in \mathbb{N}, \end{cases}$$

- The Dynamic Programming Principle gives:

$$\vartheta(x) = \min_{u \in U} \mathbb{E}\Big[C(x) - \beta u + (1-\lambda)\vartheta\big((1-\delta)x + u + \omega\sigma x\big)\Big].$$

Example 6: Irreversible investment models

- Consider the SOCP :

$$\vartheta(x) = \begin{cases} \max \ \mathbb{E}\Big[\sum_{t=0}^{\infty}(1-\lambda)^j(C(X_t)-\beta u_t)\Big]; \\ X_{t+1} = (1-\delta)X_t + u_t + \omega_t \sigma X_t, \quad X_0 = x, \\ u_t \in U \ \forall \ t \in \mathbb{N}, \end{cases}$$

- The Dynamic Programming Principle gives:

$$\vartheta(x) = \min_{u \in U}\left\{C(x) - \beta u + \frac{(1-\lambda)}{2}\Big(\vartheta((1-\delta)x+u+\sigma x) + \vartheta((1-\delta)x+u+\sigma x)\Big)\right\}$$

$$\text{Find } x \in X, \quad \min_{\alpha \in \mathcal{A}} \max_{\beta \in \mathcal{B}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

- Extension of the Newton method for solving nonsmooth equations $F(x) = 0$ have been widely studied over the last two decades

  (Robinson, Mifflin, Kummer, Bolte-Daniilidis-Lewis, Kuntz-Scholtes, Facchinei-Pang, Qi-Sun, Ito-Kunish, Hintermuller, Ulbrich, ...)

- Let F be locally Lipschitz. F is *semismooth* at x iff F is directionally differentiable at x and

$$\max_{M \in \partial F(x+h)} \|F(x+h) - F(x) - Mh\| = o(\|h\|).$$

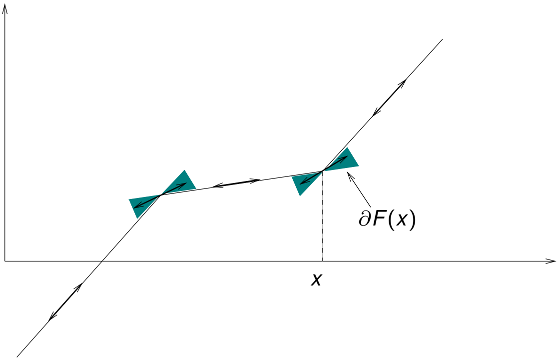


Figure: Example of a semi-smooth function

➤ **Nonsmooth Newton Algorithm (semismooth function F)**

(i) Choose a regular $x^0 \in X$. Set $k = 0$.

(ii) If $F(x^k) = 0$ then stop.

(iii) Take $M^k \in \partial F(x^k)$, and solve

$$F(x^k) + M^k(x^{k+1} - x^k) = 0$$

(iv) set $k = k + 1$ and return to (ii).

➤ **Nonsmooth Newton Algorithm (semismooth function F)**

(i) Choose a regular $x^0 \in X$. Set $k = 0$.
(ii) If $F(x^k) = 0$ then stop.
(iii) Take $M^k \in \partial F(x^k)$, and solve

$$F(x^k) + M^k(x^{k+1} - x^k) = 0$$

(iv) set $k = k + 1$ and return to (ii).

➤ **Superlinear convergence result**

Let $F : \mathbb{R}^N \times \mathbb{R}^N$ is a semi-smooth function, and a regular point $x^* \in \mathbb{R}^N$ such that $F(x^*) = 0$. Then

$$\exists \delta > 0, \ \forall x^0 \in B(x^*, \delta), \ \lim_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0$$

We say that x is a regular point of F if each $g \in \partial F(x)$ is invertible

- A mapping $F : X \to X$ is called slantly differentiable in the open subset $D \subset X$ if there exists a family of mappings $G : D \to \mathcal{L}(X, X)$ such that

$$\|F(x+h) - F(x) - G(x+h)h\| = 0(\|h\|), \quad x \in D.$$

  Ref: Kummer'88, ...

- The slant differentiability is a more general concept than semismoothness concept. In fact, the slanting functions $G(x+h)$ are not required to be element of $\partial F(x+h)$.

- If F is semismooth on U, then a single-valued $V(x) \in \partial F(x)$, $x \in U$, serves as a slanting function.

- **Nonsmooth Newton Algorithm (Slantly differentiable functions)**
  (i) Choose a regular $x^0 \in X$. Set $k = 0$.
  (ii) If $F(x^k) = 0$ then stop.
  (iii) Compute x^{k+1} by solving

$$F(x^k) + G(x^k)(x^{k+1} - x^k) = 0$$

  (iv) set $k = k + 1$ and return to (ii).

- **Convergence result**
  Let $F : \mathbb{R}^N \times \mathbb{R}^N$ is slantly differentiable in an open neigborhood U of x^* with slanting function G. If $G(x)$ is nonsingular for all $x \in U$ and $\{\|G(x)^{-1}\| : x \in U\}$ is bounded, then

$$\exists \delta > 0,\ \forall x^0 \in B(x^*, \delta),\ \text{ the NNA converges superlinearly to } x^*$$

Ref: Ito-kunisch, Ulbrich, ...

$$\text{Find } x \in X, \quad \min_{\alpha\in\mathcal{A}} \max_{\beta\in\mathcal{B}} (A^{\alpha,\beta}x - b^{\alpha,\beta}) = 0,$$

$$\text{Find } x \in \mathbb{R}^N, \ \min_{\alpha\in\mathcal{A}}(A^\alpha x - b^\alpha) = 0. \qquad (P_{\min})$$

It is useful to note that problem $(P_{\min})$ is equivalent to

$$\text{Find } x \in \mathbb{R}^N, \quad \min_{\alpha\in\mathcal{A}^N}\Big(A(\alpha)x - b(\alpha)\Big) = 0$$

with $A_{ij}(\alpha) := A_{ij}^{\alpha_i}, \quad b_i(\alpha) = b_i^{\alpha_i}$.

$$\text{Find } x \in \mathbb{R}^N, \ \min_{\alpha\in\mathcal{A}}(A^\alpha x - b^\alpha) = 0. \qquad (P_{\min})$$

It is useful to note that problem $(P_{\min})$ is equivalent to

$$\text{Find } x \in \mathbb{R}^N, \quad \min_{\alpha\in\mathcal{A}^N}\Big(A(\alpha)x - b(\alpha)\Big) = 0$$

with $A_{ij}(\alpha) := A_{ij}^{\alpha_i}, \quad b_i(\alpha) = b_i^{\alpha_i}.$

Indeed, for all i,

$$\begin{aligned} 0 &= \min_{a\in\mathcal{A}}\left(A^a x - b^a\right)_i = \min_{\alpha\in\mathcal{A}^N}\Big(A^{\alpha_i}x - b^{\alpha_i}\Big)_i \\ &= \min_{\alpha\in\mathcal{A}^N}\Big(A(\alpha)x - b(\alpha)\Big)_i \end{aligned}$$

Howard's algorithm

Initialize α^0 *in* $\mathcal{A}^N$,

Iterate for $k \geq 0$:

(i) *find* $x^k \in \mathbb{R}^N$ *solution of* $A(\alpha^k)x^k = b(\alpha^k)$.

(ii) $\alpha^{k+1} := \text{argmin}_{\alpha \in \mathcal{A}^N} \big(A(\alpha)x^k - b(\alpha)\big)$.

• Howard's algorithm also called policy iterations method .

• Refs: Bellman (1955-57), Howard (1960), Puterman et al. (1979), Santos et al. (04), ...

Convergence results of Howard's algorithm

We use the following assumptions
(H1) $\alpha \in \mathcal{A}^N \to A(\alpha)$ and $\alpha \in \mathcal{A}^N \to b(\alpha)$ are continuous (obvious if $\mathcal{A}$ is finite).

(H2) $\forall \alpha \in \mathcal{A}^N$, $A(\alpha)$ is a monotone matrix:

$$A(\alpha)X \geq 0 \quad \Rightarrow \quad X \geq 0.$$

Theorem [Bokanowski-Maroso-HZ'09].
There exists a unique $x^* \in \mathbb{R}^N$ solution of $(P_{\min})$. Moreover, Howard's sequence (x^k) satisfies

(i) $x^k \leq x^{k+1}$ for all $k \geq 0$, and x^k converges to x^*

(ii) If $\mathcal{A}$ is infinite, $x^k \to x^*$ **super-linearly.**

(iii) If $\mathcal{A}$ is finite, the algorithm converges in $(\mathrm{Card}(\mathcal{A}))^N$ iterations

Idea of the proof (convergence)

- $x_k \leq x_{k+1}$:

$$\begin{aligned} A(\alpha^{k+1})x^k - b(\alpha^{k+1}) &= \min_{\alpha \in \mathcal{A}_\infty} (A(\alpha)x^k - b(\alpha)) \\ &\leq A(\alpha^k)x^k - b(\alpha^k) \\ &= 0 \\ &= A(\alpha^{k+1})x^{k+1} - b(\alpha^{k+1}). \end{aligned}$$

- Unicity of x^*: similar arguments.

- x_k bounded: $x_k = A(\alpha^k)^{-1} b(\alpha_k)$.

- $F(x^*) = 0$: using that $F(x_k) = A(\alpha^{k+1})x^k - b(\alpha^{k+1})$...

Link with Newton's algorithm

Let

$$F(x) := \min_{\alpha \in \mathcal{A}^N} (A(\alpha)x - b(\alpha)).$$

Then:

$$A(\alpha^{k+1})x^k - b(\alpha^{k+1}) = F(x^k) \qquad \text{policy improvement},$$
$$A(\alpha^{k+1})x^{k+1} - b(\alpha^{k+1}) = 0 \qquad \text{policy evaluation.}$$

Therefore

$$x^{k+1} = x^k - A(\alpha^{k+1})^{-1} F(x^k). \tag{1}$$

Superlinear Convergence

- For every $x \in \mathbb{R}^N$, set

$$\mathcal{A}(x) := \left\{ \alpha \in \mathcal{A}^N,\ A(\alpha)x - b(\alpha) = F(x) \right\}.$$

Then $x \longmapsto \mathcal{A}(x)$ is upper semicontinuous.

Superlinear Convergence

- For every $x \in \mathbb{R}^N$, set
$$\mathcal{A}(x) := \Big\{\alpha \in \mathcal{A}^N,\ A(\alpha)x - b(\alpha) = F(x)\Big\}.$$
Then $x \longmapsto \mathcal{A}(x)$ is upper semicontinuous.

- F is *slantly differentiable* with slanting function $x \longmapsto A(\alpha(x))$, with $\alpha(x) \in \mathcal{A}(x)$.

- Howard's algorithm can be interpreted as a nonsmooth Newton method for a slantly differentiable function: the superlinear convergence can be obtained by the general theory.

Application: Merton's portfolio problem

- Model:

$$\min_{\alpha \in \mathcal{A}} \left(\partial_t \vartheta - \frac{1}{2}\sigma^2 \alpha^2 s^2 \partial_{ss}^2 \vartheta - (\alpha\mu + (1-\alpha)r) x \partial_s \vartheta \right) = 0,$$
$$t \in [0,T],\ s \in (0, S_{\max}),$$
$$\vartheta(0,s) = \varphi(s), \quad s \in (0, S_{\max}).$$

- Assume $\varphi(x) = x^p$ (for some $p \in (0,1)$)
- Mixed boundary condition at $s = S_{\max}$:

$$\partial_x \vartheta(t, S_{\max}) = \frac{p}{S_{\max}} \vartheta(t, S_{\max}), \quad t \in [0,T]. \tag{2}$$

Finite Difference Scheme

• **Mesh:** Let $s_j = jh$ with $h = S_{\max}/N_s$ and $t_n = n\Delta t$ with $\Delta t = T/N$, where $N \geq 1$ and $N_s \geq 1$.

• **Implicit Euler scheme:**

$$\min_{\alpha \in \mathcal{A}} \Bigg(\frac{V_j^{n+1} - V_j^n}{\Delta t} - \frac{1}{2}\sigma^2 s_j^2 \alpha^2 \frac{V_{j-1}^{n+1} - 2V_j^{n+1} + V_{j+1}^{n+1}}{h^2}$$

$$-(\alpha\mu + (1-\alpha)r)s_j \frac{V_{j+1}^{n+1} - V_j^{n+1}}{h} \Bigg) = 0,$$

$$j = 0, \dots, N_s, \; n = 0, \dots, N-1,$$

$$\frac{V_{N_s}^{n+1} - V_{N_s-1}^{n+1}}{h} = \frac{p}{S_{\max}} V_{N_s}^{n+1}, \quad n = 0, \dots, N-1,$$

$$V_j^0 = \varphi(s_j), \quad j = 0, \dots, N_s.$$

Monotonicity.

For $b := V^n$ given (and for a given time iteration $n \geq 0$), the computation of $x = V^{n+1} \in \mathbb{R}^{N_s+1}$ (i.e, $x = (V_0^{n+1}, \ldots, V_{N_s}^{n+1})^T$) is equivalent to solve

$$\min_\alpha (A^\alpha x - b) = 0,$$

where $A_\alpha := I + \Delta t B_\alpha$ and B_α is the matrix of $\mathbb{R}^{(N_s+1)\times(N_s+1)}$ such that, for all $j = 0, \ldots, N_s - 1$,

$$(B_\alpha U)_j = +\frac{1}{2}\sigma^2 s_j^2 \alpha^2 \frac{-U_{j-1} + 2U_{j-1} - U_{j+1}}{h^2} \\ -(\alpha\mu + (1-\alpha)r)s_j \frac{U_{j+1} - U_j}{h},$$

(and similar expression for $(B_\alpha U)_{N_s}$) We obtain the monotonicity of the matrices A^α under a condition $\frac{\Delta t}{h} \leq C$.

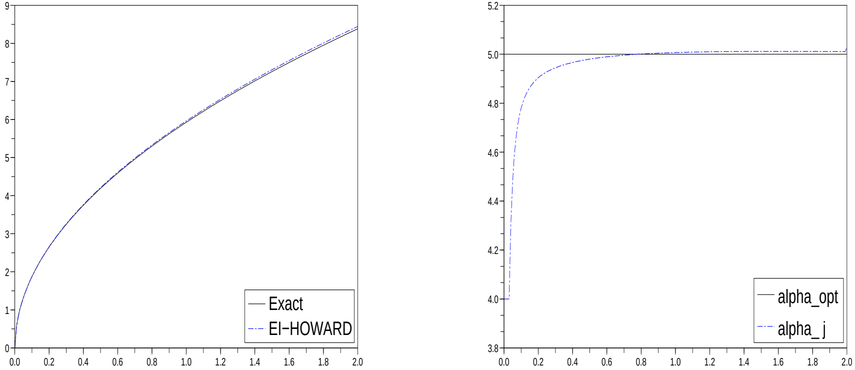


Figure: Plot of (U_j^N) (left) and of the discrete optimal control (α_j) at time $t_N = 1$ (right), with respect to s_j. Parameters: $S_{\max} = 2$, $\mathcal{A} = [4, 6]$, $p = \frac{1}{2}$, $\sigma = 0.2$, $r = 0.1$, $\mu = 0.2$, $T = 1$, and $N_s = 200$, $N = 20$.

Quadratic convergence (Rust and Santos 04')

Set $f_i(x, \alpha) := [A(\alpha)x - b]_i$.

- Assume that $\mathcal{A}$ is a compact interval of $\mathbb{R}$, for all $1 \leq i \leq N$,

$$f_i(x, \alpha) = r_i(x)\alpha_i^2 + s_i(x)\alpha_i + t_i(x) \qquad \forall x \in \mathbb{R}^N,$$

  with $r_i(x) > 0$, and with $r_i(\cdot)$ and $s_i(\cdot)$ lipschitz functions.
- In this case, for every $x \in X$, a minimizeer α^x is given by

$$\alpha_i^x := \text{argmin}_{\alpha_i \in \mathcal{A}} f_i(x, \alpha) = P_{\mathcal{A}}(-\frac{s_i(x)}{2r_i(x)})$$

  where $P_{\mathcal{A}}$ denotes the projection on the interval $\mathcal{A}$.
- Hence in the neighborhood of the solution x^*, we obtain that $\|\alpha^x - \alpha^{x^*}\| \leq Const\|x - x^*\|$. This implies also that $\|A(\alpha^x) - A(\alpha^{x^*})\| \leq Const\|x - x^*\|$. This leads to a global quadratic convergence result of Howard algorithm.

$$\text{Find } x \in X, \quad \min_{\alpha \in \mathcal{A}} \max_{\beta \in \mathcal{B}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

$$\text{find } x \in \mathbb{R}^N, \quad \min(Qx - b, x - g) = 0,$$

Algorithm (Ho-2) for the obstacle problem: same as (Ho-1), but chose $\alpha_i = 0$ in the case of equality $(Qx^k - b)_i = (x^k - g)_i$.

Theorem.

- Howard's algorithm **(Ho-2)** converges in at most N iterations (i.e, $x^k = x^{k+1}$ for some $k \leq N$).
- It is equivalent to the Primal-Dual Active set algorithm

Idea of the proof

- $x^k \geq g \; \forall k \geq 1$.
- $(\alpha^k)_{k\geq 0}$ is decreasing in $\mathcal{A}^N$.
- There exists a first index $k \in [0, N]$ such that $\alpha^k = \alpha^{k+1}$. Hence

$$\begin{aligned} F(x^{k+1}) &= A(\alpha^{k+2})x^{k+1} - b(\alpha^{k+2}) \\ &= A(\alpha^{k+1})x^{k+1} - b(\alpha^{k+1}) = 0 \end{aligned}$$

and we obtain $F(x^k) = F(x^{k+1}) = 0$.

Application: American options.

$$\min\left(\partial_t u - \frac{1}{2}\sigma^2 s^2 \partial^2_{ss} u - rs\partial_s u + ru,\ u - \varphi(x)\right) = 0, \tag{3a}$$

$$t \in [0, T],\ s \in (0, S_{\max}),$$

$$u(t, S_{\max}) = 0, \quad t \in [0, T], \tag{3b}$$

$$u(0, s) = \varphi(s), \quad x \in (0, S_{\max}). \tag{3c}$$

where $\sigma > 0$ represents a volatily, $r > 0$ is the interest rate, $S_{\max} > 0$ is large, $\varphi(s) := \max(K - s, 0)$ is the "Payoff" function ($K > 0$ is the "strike").

Finite Difference Scheme (Implicit Euler)

- **Implicit Euler scheme:**

$$\begin{cases} \min\left(\frac{U_j^{n+1}-U_j^n}{\Delta t} - \frac{1}{2}\sigma^2 s_j^2 \frac{(D^2U^{n+1})_j}{h^2} - rs_j\frac{D^+U_j^{n+1}}{h} + rU_j^{n+1};\right. \\ \qquad\qquad \left. U_j^{n+1} - g_j\right) = 0, \quad j = 0,\dots,N_s - 1,\ n = 0,\dots,N_T - 1, \\ U_{N_s}^{n+1} = 0, \quad n = 0,\dots,N_T - 1, \\ U_j^0 = g_j := \varphi(s_j), \quad j = 0,\dots,N_s - 1 \end{cases}$$

where $(D^2U)_j$ and $(D^+U)_j$ are finite differences defined by

$$(D^2U)_j := U_{j-1} - 2U_j + U_{j+1}, \quad (D^+U)_j := U_{j+1} - U_j,$$

- Stability without CFL condition.

• For $b := U^n$ given, the problem to find $x = U^{n+1} \in \mathbb{R}^{N_s}$ (i.e, $x = (U_0^{n+1}, \ldots, U_{N_s-1}^{n+1})^T$) is equivalent to $\min(Bx - b, x - g) = 0$, where $B = I + \Delta t A$ and A is the matrix of $\mathbb{R}^{N_s}$ such that for all $j = 0, \ldots, N_s - 1$:

$$(AU)_j = -\frac{1}{2}\sigma^2 s_j^2 \frac{U_{j+1} - 2U_{j-1} + U_{j+1}}{h^2} - rs_j \frac{U_{j+1} - U_j}{h} + rU_j,$$

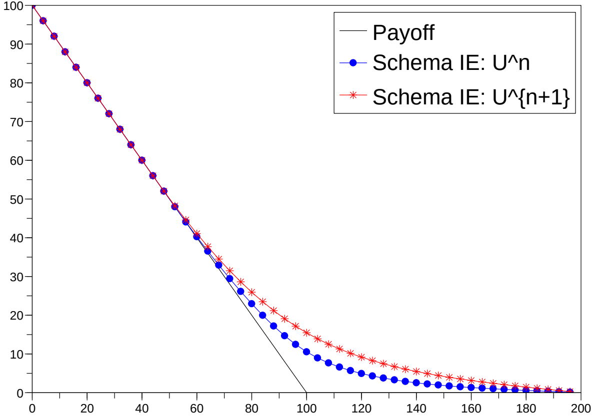
(assuming $U_{N_s} = 0$).

• B is an M-matrix. Hence ($H2$) is satisfied and we can apply Howard's algorithm and generate a sequence of approximations (x^k) (for a given time step t_n of the IE scheme).

• We choose to apply Howard's algorithm with starting point $x^0 := U^n$.

Maximal bound of the total number of Howard's iterations

Proposition. The total number of linear systems to be solved (using algorithm (Ho-2')) in the IE scheme, from $n = 0$ to $n = N_T - 1$, is bounded by N_s.



$$\text{Find } x \in X, \quad \min_{\alpha \in \mathcal{A}} \max_{\beta \in \mathcal{B}} (A^{\alpha,\beta} x - b^{\alpha,\beta}) = 0,$$

• Define the functions F and G on $\mathbb{R}^N$ by:

$$F^\beta(x) := \min_{\alpha\in\mathcal{A}}(A^{\alpha,\beta}x - b^{\alpha,\beta}), \quad \text{and} \quad G(x) := \max_{\beta\in\mathcal{B}} F^\beta(x) \quad \text{for } x \in \mathbb{R}^N.$$

Algorithm (Ho-3)

Initialize $\beta^0 \in \mathcal{B}^N$, and iterate for $k \geq 0$:

(i) *Find* x^k *such that* $F^{\beta^k}(x^k) = 0$

(ii) *Set*

$$\beta^{k+1} := \mathit{argmax}_{\beta\in\mathcal{B}} F^\beta(x^k)$$

- Note that, for every $k \geq 0$, the equation $F^{\beta^k}(x) = 0$ is a min-problem. The resolution in step (i) of the above algorithm can be performed with the Howard's algorithm.
- The above algorithm is no more a Newton-like method !

• Define the functions F and G on $\mathbb{R}^N$ by:

$$F^\beta(x) := \min_{\alpha\in\mathcal{A}}(A^{\alpha,\beta}x - b^{\alpha,\beta}), \quad \text{and} \quad G(x) := \max_{\beta\in\mathcal{B}} F^\beta(x) \quad \text{for } x \in \mathbb{R}^N.$$

Algorithm (Ho-3) Let $(\eta_k)_{k\leq 0}$ be in $\mathbb{R}^+$.

Initialize $\beta^0 \in \mathcal{B}^N$, and iterate for $k \geq 0$:

(i) *Find* x^k *such that* $\|F^{\beta^k}(x^k)\| \leq \eta_k$

(ii) *Set*

$$\beta^{k+1} := argmax_{\beta\in\mathcal{B}} F^\beta(x^k)$$

- Note that, for every $k \geq 0$, the equation $F^{\beta^k}(x) = 0$ is a min-problem. The resolution in step (i) of the above algorithm can be performed with the Howard's algorithm.
- The above algorithm is no more a Newton-like method !

Theorem Assume the monotonicity property of the matrices. Let $(\eta_k)_{k\geq 0}$ be a sequence of $\mathbb{R}^+$, with $\sum_{k\geq 0} \eta_k < \infty$. Then the sequence of iterates (x^k) given by Algorithm Ho-3 converges to the unique solution x^* of $G(x^*) = 0$.
Furthermore, we have the lower bound estimate

$$x^k \geq x^* - C\eta_k, \quad \text{with } C := \max_{\alpha\in\mathcal{A}^N,\, \beta\in\mathcal{B}^N} \|B(\alpha,\beta)^{-1}\|. \tag{4}$$

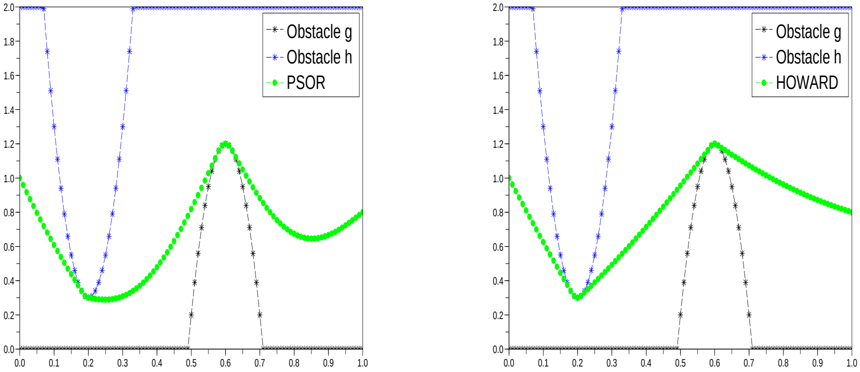


Figure: PSOR (left, with $k = 200$ iterations) and Howard's algorithm (right, with $k = 14$ iterations; 88 linear systems) for the double obstacle problem with $N = 99$. Values U_j^n are plotted vs. s_j.

... many thanks for your attention!