On the convergence of decomposition methods for multistage stochastic convex programs

P.Girardeau (Univ. of Auckland), V. Leclère (ENPC),
A. Philpott (Univ. of Auckland).

July 29, 2013

Contents

Contents

Problem statement

- We consider a probabilistic world encoded on a (non-recombining) finite tree $\mathcal{N}$, such that each node m has a probability Φ_m.
- We consider a discrete time controled dynamical system where the state x follow the equation

$$x_m = f_m(x_{p(m)}, u_m)$$

where
  - $p(m)$ is the parent node of m,
  - u_m is the control choosen from node $p(m)$ to reach node m, so knowing the uncertainty,
  - the uncertainty in the evolution is taken into account in the evolution function f_m itself.
- Loosely speaking we want to minimize the expected cost

$$\mathbb{E}\left(\sum_{t=0}^{T} C_t(x_t, u_t) + V_T(x_T)\right).$$

Problem statement

$$\min_{x,u} \sum_{n \in \mathcal{N} \setminus \{\mathcal{L}\}} \sum_{m \in r(n)} \Phi_m C_m(x_n, u_m) + \sum_{m \in \mathcal{L}} \Phi_m V_m(x_m) \tag{1a}$$

$$\text{s.t.} \quad x_m = f_m\left(x_{p(m)}, u_m\right), \qquad \forall m \in \mathcal{N} \setminus \{0\}, \tag{1b}$$

$$x_0 \text{ is given}, \tag{1c}$$

$$x_m \in \mathcal{X}_m, \qquad \forall m \in \mathcal{N}, \tag{1d}$$

$$u_m \in \mathcal{U}_m(x_{p(m)}), \qquad \forall m \in \mathcal{N} \setminus \{0\}. \tag{1e}$$

- x_m is the state, u_m the control for node m,
- C_m the cost function at node m, V_m the final cost function, f_m the dynamic function (depending on uncertainties),
- Φ_m the probability of node m,
- $\mathcal{X}_m$ constraint set on the state, $\mathcal{U}_m$ a constraint multifunction on the control.

Contents

Bellman's function

We introduce, for each node n, the Bellman function $V_n(x)$.

$$V_n(x) = \sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \min_{u \in \mathcal{U}(x)} \Big\{ C_m(x,u) + V_m \circ f_m(x,u) \Big\}.$$

And selecting for any node n the control $u_n(x)$ in

$$u_m(x) \in \operatorname*{argmin}_{u \in \mathcal{U}_m(x)} \Big\{ C_m(x,u) + V_m \circ f_m(x,u) \Big\},$$

gives an optimal strategy.

Moreover assuming that we have for any children m of a node n an outer approximation of V_m, i.e a function V_m^k such that $V_m^k \leq V_m$, then the function

$$V_n^k : x \mapsto \sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \min_{u \in \mathcal{U}(x)} \Big\{ C_m(x,u) + V_m^k \circ f_m(x,u) \Big\},$$

is an outer approximation of V_n.

Bellman's function

We introduce, for each node n, the Bellman function $V_n(x)$.

$$V_n(x) = \sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \min_{u \in \mathcal{U}(x)} \Big\{ C_m(x,u) + V_m \circ f_m(x,u) \Big\}.$$

And selecting for any node n the control $u_n(x)$ in

$$u_m(x) \in \operatorname*{argmin}_{u \in \mathcal{U}_m(x)} \Big\{ C_m(x,u) + V_m \circ f_m(x,u) \Big\},$$

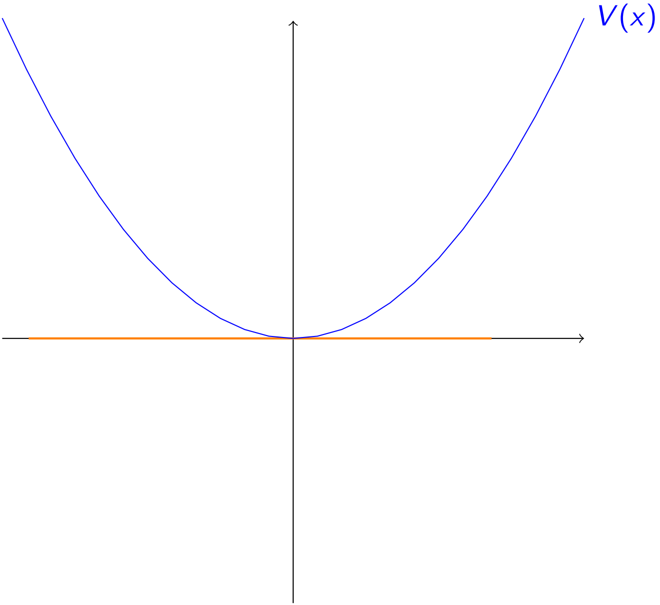
gives an optimal strategy.

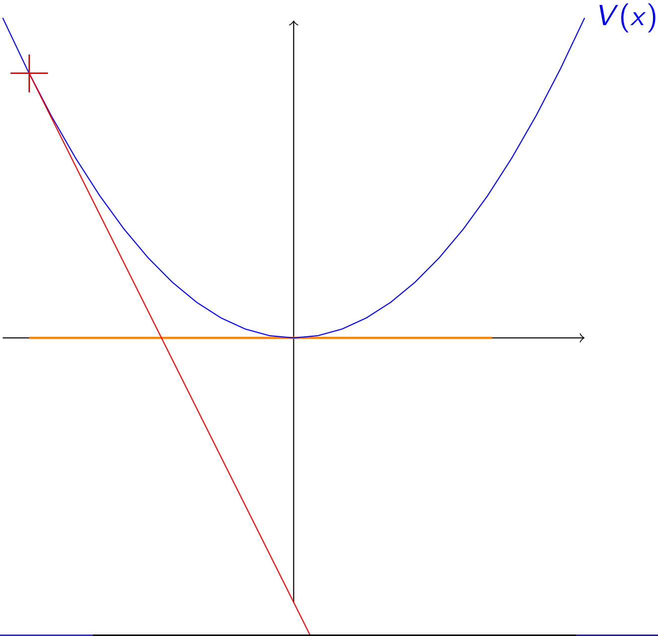
Moreover assuming that we have for any children m of a node n an outer approximation of V_m, i.e a function V_m^k such that $V_m^k \leq V_m$, then the function

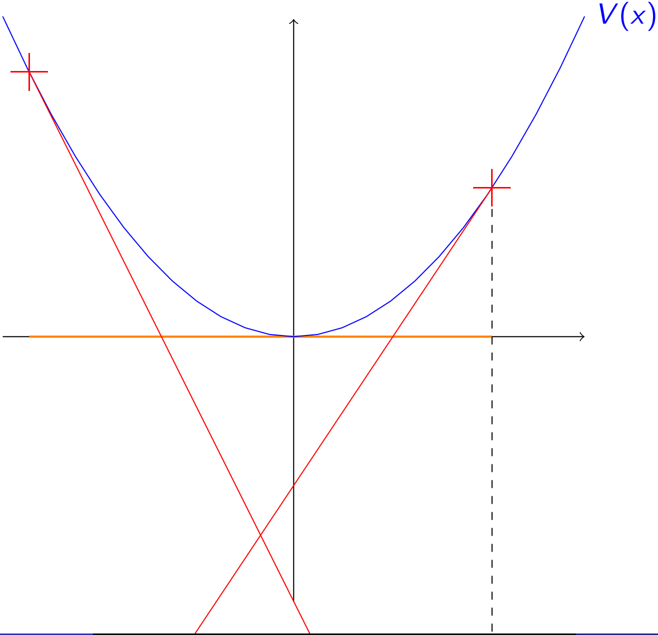
$$V_n^k : x \mapsto \sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \min_{u \in \mathcal{U}(x)} \Big\{ C_m(x,u) + V_m^k \circ f_m(x,u) \Big\},$$

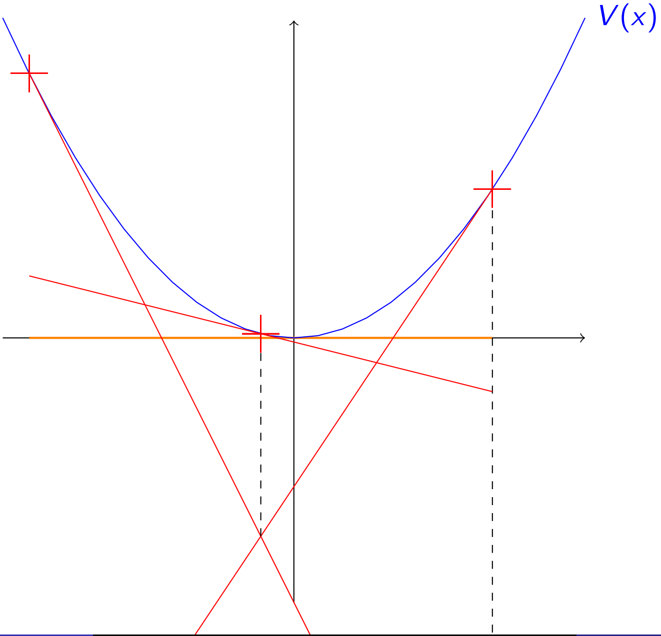
is an outer approximation of V_n.

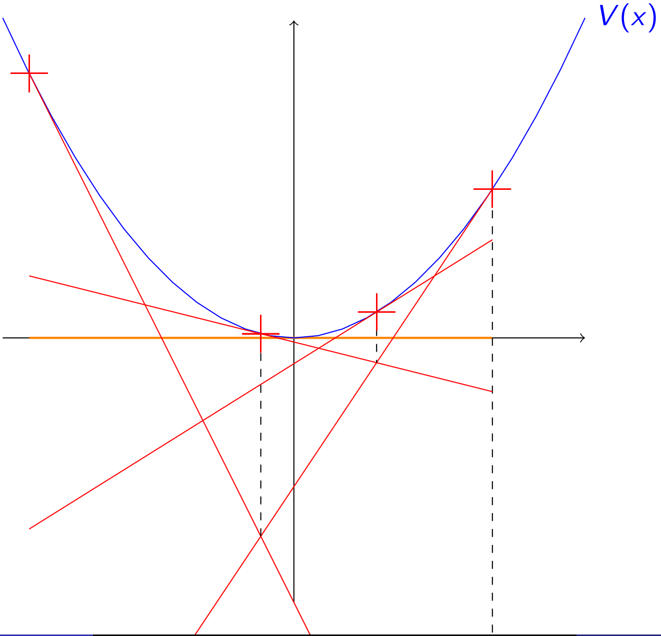
Contents


$V(x)$


$V(x)$


V(x)


$V(x)$


$V(x)$

Contents

Bird eye view

The class of algorithms studied consist in having an outer-approximation of V_n denoted V_n^{k-1} for any node and updating him by adding cuts comparable to the Kelley's algorithm. The algorithm is two-folds :

- In a first place, called forward path, given the actual set of outer approximation we generate state trajectories on the whole tree denoted x_n^k.
- In a second place, we select a set of nodes $(n_i)_{i \in I}$. For each of this nodes we update the outer approximations by adding a cut. For the other nodes we simply conserve our former outer approximation.

Forward path

In the first part, also called Forward path we visit the tree from the root node, were $x_0^k = x_0$ and recursively choose, for any children m of a node n,

$$u_m^k \in \underset{u \in \mathcal{U}_m(x_n^k)}{\operatorname{argmin}} \Big\{ C_m(x_n^k, u) + V_m^{k-1} \circ f_m(x_n^k, u) \Big\},$$

and define

$$x_m^k = f_m(x_n^k, u_m^k).$$

"Backward path" (1/2)

We consider a selecting stochastic process $(y^k)_{k\in\mathbb{N}}$, with value in $\{0,1\}^{|\mathcal{N}|}$. For each selected node n, i.e node n such that $y_n^k = 1$, and for every child node m of node n, solve:

$$\hat{\theta}_m^k = \min_{u_m,x} \quad C_m(x,u_m) + V_m^{k-1} \circ f_m(x,u_m), \tag{2a}$$

$$x = x_{n_i}^k \qquad [\hat{\beta}_m^k] \tag{2b}$$

$$u_m \in \mathcal{U}_m(x) \tag{2c}$$

$$f_m(x,u_m) \in \mathcal{X}_m \tag{2d}$$

"Backward path" (2/2)

For each selected node n, define :

$$\theta_n^k = \sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \hat{\theta}_m^k \quad \text{and} \quad \beta_n^k = \sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \hat{\beta}_m^k.$$

Thus

$$x \mapsto \theta_n^k + \left\langle \beta_n^k, x - x_n^k \right\rangle$$

is an affine function below V_n. Finally, we update the outer approximations. For every selected node n

$$V_n^k(x) := \max\left(V_n^{k-1}(x), \theta_n^k + \left\langle \beta_n^k, x - x_n^k \right\rangle \right), \quad x \in \mathcal{X}_t.$$

Contents

Selection process

Definition

Let τ be a positive integer. The process $(y^k)_{k\in\mathbb{N}}$ is called a *τ-admissible selection process* if

(i) For all $k \in \mathbb{N}$, and all $\kappa \in \{0, \dots, \tau - 1\}$, if the node m is selected at time $k\tau + \kappa$ (i.e. $y_m^{k\tau+\kappa} = 1$) then all ancesters n of m, have not been selected between time $k\tau$ and time $k\tau + \kappa - 1$.

(ii) $\tilde{y}_n^k := \max\{y_n^j \mid k\tau \leq j < (k+1)\tau\}$ (which determine if V_n is updated during the “big step” k) satisfies, $(\tilde{y}_m^k)_{k\in\mathbb{N}}$ is i.i.d. and $\tilde{y}_m^k$ is independent of the selection process up to time $k\tau - 1$.

(iii) All nodes are updated with positive probability in a “big step”:

$$\forall n \in \mathcal{N}\backslash\mathcal{L}, \qquad \mathbb{P}(\tilde{y}_n^k = 1) > 0.$$

Hypotheses

1. $\mathcal{X}_n$ is convex compact;
2. the multifunctions $\mathcal{U}_m$ are convex and convex compact valued (equivalent to taking the control in a compact set with constraint $g_m(x, u_m) \leq 0$ where g is jointly convex);
3. all cost functions C_n, and final cost functions V_m, are convex lower semicontinuous proper functions;
4. the functions f_m are linear;
5. all final cost functions V_m are Lipschitz-continuous on $\mathcal{X}_m$;
6. There exists $\delta > 0$ such that for all non terminal nodes n,
   1. $\forall x \in \mathcal{X}_n + B(0,\delta), \quad \forall m \in r(n), \qquad f_m(x, \mathcal{U}_m(x)) \cap \mathcal{X}_m \neq \emptyset,$
   2. $\forall x \in \mathcal{X}_n + B(0,\delta), \quad \forall u \in \mathcal{U}_m(x), \qquad C_n(x,u) < \infty.$

Hypotheses

1. $\mathcal{X}_n$ is convex compact;
2. the multifunctions $\mathcal{U}_m$ are convex and convex compact valued (equivalent to taking the control in a compact set with constraint $g_m(x, u_m) \leq 0$ where g is jointly convex);
3. all cost functions C_n, and final cost functions V_m, are convex lower semicontinuous proper functions;
4. the functions f_m are linear;
5. all final cost functions V_m are Lipschitz-continuous on $\mathcal{X}_m$;
6. There exists $\delta > 0$ such that for all non terminal nodes n,
    1. $\forall x \in \mathcal{X}_n + B(0, \delta), \quad \forall m \in r(n), \qquad f_m(x, \mathcal{U}_m(x)) \cap \mathcal{X}_m \neq \emptyset,$
    2. $\forall x \in \mathcal{X}_n + B(0, \delta), \quad \forall u \in \mathcal{U}_m(x), \qquad C_n(x, u) < \infty.$

Convergence result

Theorem

Assume the preceding hypotheses holds true and that the selection process is τ-admissible for some integer $\tau > 0$.
Then we have that, $\mathbb{P}$-almost surely the upper and lower bound are converging toward the optimal value, and we can obtain an $\varepsilon-$optimal strategy for any $\varepsilon > 0$.
More precisely

$$\sum_{m \in r(n)} \frac{\Phi_m}{\Phi_n} \left[C_m\left(x_n^{k\tau}, u_m^{k\tau}\right) + V_m^k \circ f_m\left(x_n^{k\tau}, u_m^{k\tau}\right)\right] - V_n\left(x_n^{k\tau}\right) \to 0.$$

and

$$\lim_{k \to +\infty} V_n\left(x_n^{k\tau}\right) - V_n^{k\tau}\left(x_n^{k\tau}\right) = 0.$$

Broad sketch of the proof

1. A first lemma gives regularity results on V_m and V_m^k.
2. A second lemma assert that if V_m^k converges toward V_m at points x_m^k for all children of n, then the estimation of the cost at node n is also convergent. The proof is close to Kelley's algorithm proof and rely on compacity of $\prod_{m\in r(n)} \mathcal{X}_m$.
3. Finally the proof is done by backward induction. It is straightforward for the node selected when they are selected, and extended to all nodes by an independence property and law of large number.

Contents

CUPPS

Here at each major iteration we choose a $T-1$-step scenario and compute the optimal trajectory while at the same time updating the value function for each node of the branch.
In our setting, this uses a 1-admissible selection process $(y^k)_{k\in\mathbb{N}}$ defined by an i.i.d. sequence of random variables, with y^0 selecting a single branch of the tree. Our Theorem shows that for every node n the upper and lower bound converges.

SDDP and DOASA 1/2

There are two phases in each major iteration of the SDDP algorithm, namely a forward pass, and a backward pass of $T-1$ steps.

Given a current polyhedral outer approximation of the Bellman function $(V_n^{\tilde{k}-1})_{n\in\mathcal{N}\setminus\mathcal{L}}$, a major iteration $\tilde{k}$ of the SDDP algorithm consists in:

- selecting uniformly a number N of scenarios ($N=1$ for DOASA);
- simulating the optimal strategy for the approximated problem on each of this scenarios yields a trajectory,
- For $t=T-1$ down to $t=0$ for each scenario we update the approximation by adding a cut, and then go backward in time. Thus at time t when we compute a new cut the approximation of V_{t+1} has already been updated.

SDDP and DOASA 2/2

SDDP fits into our framework as follows. Given N, we define the $T-1$-admissible selection process, $(y^{(T-1)k})_{k\in\mathbb{N}}$ by an i.i.d. sequence of random variables with y^0 selecting uniformly a set of N pre-leaves (i.e. nodes whose children are leaves) of the tree. Then for $\kappa \in \{1,\dots,T-2\}$, $k\in\mathbb{N}$, $n\in\mathcal{N}\backslash\mathcal{L}$, we define

$$y_n^{k(T-1)+\kappa} := \begin{cases} 1 & \text{if there exist } m \in r(n) \text{ such that } y_m^{k(T-1)+\kappa-1} = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Our theorem gives convergence of upper and lower bound in this case.

Contents

Conclusion

- The SDDP algorithms are well known and quite used in practice especially in a linear framework to be able to use powerful solver like CPLEX.
- Until now every convergence proof has been done in the linear case and most of the time for specific implementation of the algorithm.
- Our result extend the convergence to the convex, finite distribution case, and for a wide class of algorithm.
- It rely on an assumption sligthly stronger than the relatively complete recourse case, and of some independence in the selection process which forbid deterministic (Round-robin like) selection.
- Work is ongoing to extend to the continuous distribution case.

Bibliography

M. Pereira, L.Pinto (1991).
Multi-stage stochastic optimization applied to energy planning.
Mathematical Programming

Z.Chen, W. Powell (1999).
A convergent cutting plane and partial-sampling algorithm for multistage linear programs with recourse.
Journal of Optimization Theory and Applications

A.Philpott, Z. Guan (2008).
On the convergence of stochastic dual dynamic programming and related methods.
Operations research letters

P.Girardeau, V.Leclère, A. Philpott (2013).
On the convergence of decomposition methods for multi-stage stochastic convex programs.
Submitted - on Optimization Online.

The end

Thank you for your attention !