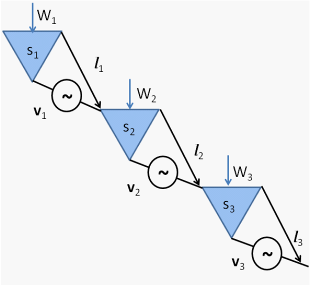
Mixed Integer Dynamic Approximation Scheme

Andy Philpott
Electric Power Optimization Centre
University of Auckland.
www.epoc.org.nz

(Joint work with Faisal Wahid, Frederic Bonnans: Funded by PGMO, EDF, Meridian Energy.)

Motivation: daily hydro generation in a river chain



Given initial storage of water, and hourly electricity prices $\pi(t)$, electricity generator arranges releases $v_i(t)$ and spill $l_i(t)$ of water to maximize revenue $\sum_t \pi(t) \sum_i g_i(v_i(t))$ while respecting the water flow constraints of the river chain. Here g_i converts water flow into power.

Stochastic dynamic programming formulation

Given initial state $\mathbf{x}_0$, we seek an optimal policy yielding $V_1(\mathbf{x}_0)$, where

$$\begin{aligned} V_t(\mathbf{x}) &= \mathbb{E}_{\xi_t}\left[\max_{u \in U(x)} \{r_t(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}_t) + V_{t+1}(f_t(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}_t))\}\right], \\ V_{T+1}(\mathbf{x}) &= R(\mathbf{x}). \end{aligned}$$

Our goal was to develop a SDDP-type algorithm.

Model might require discrete actions and states

Given initial state $\mathbf{x}_0$, we seek an optimal policy yielding $V_1(\mathbf{x}_0)$, where

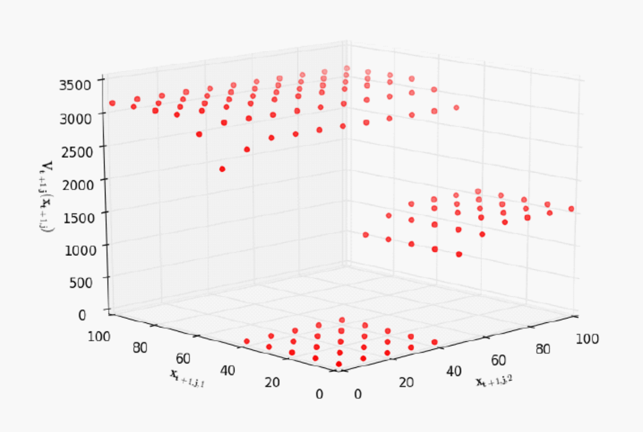
$$\begin{aligned} V_t(\mathbf{x}) &= \mathbb{E}_{\xi_t}\left[\max_{u\in U(x)}\{r_t(\mathbf{x},\mathbf{u},\xi_t)+V_{t+1}(f_t(\mathbf{x},\mathbf{u},\xi_t))\}\right], \\ V_{T+1}(\mathbf{x}) &= R(\mathbf{x}) \end{aligned}$$

Here release values $\mathbf{u}$ and inflow ω are assumed to take integer values, and satisfy network dynamics:

$$f_t(\mathbf{x},\mathbf{u},\omega,) = \mathbf{x}_t - A\mathbf{u}_t + \omega_t.$$

This gives integer-valued state variables, so $V_t(\mathbf{x})$ is not concave.

Example optimal value function



Exact optimal value function for two-reservoir problem with minimum feasible release.

Stochastic dual dynamic programming (SDDP)

[Pereira and Pinto, 1991]

Assume $V_t(\mathbf{x})$ is concave.

Represent an approximately optimal policy by an outer approximation of $V_t(\mathbf{x})$ using cutting planes.

Evaluate the approximation by simulating sample paths of the states under the policy by solving linear programming stage problems.

Improve the policy at the states visited by simulation using a cutting-plane approximation defined by subgradients of the stage optimal value function.

Mixed integer dynamic approximation scheme (MIDAS)
[Wahid et al, 2016]

Assume $V_t(\mathbf{x})$ is monotonic.

Represent an approximately optimal policy by an outer approximation of $V_t(\mathbf{x})$ using step functions.

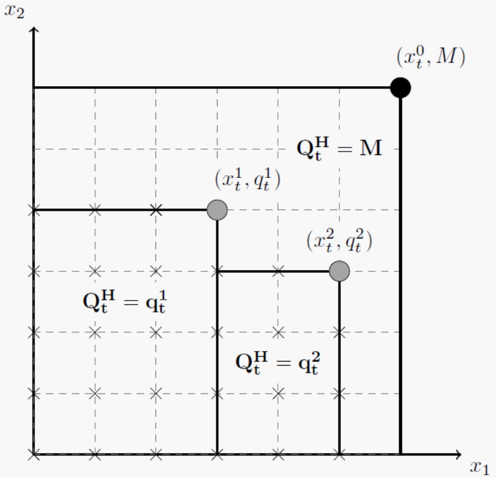
Evaluate the approximation by simulating sample paths of the states under the policy by solving mixed-integer programming stage problems.

Improve the policy at the states visited by simulation using a step-function approximation defined by the optimal values of the MIP stage problems.

Summary

Summary

Approximation of value function



$V_t(x)$ is (outer) approximated by a piecewise constant function $Q^H(x)$. Here $q_t^1 < q_t^2$.

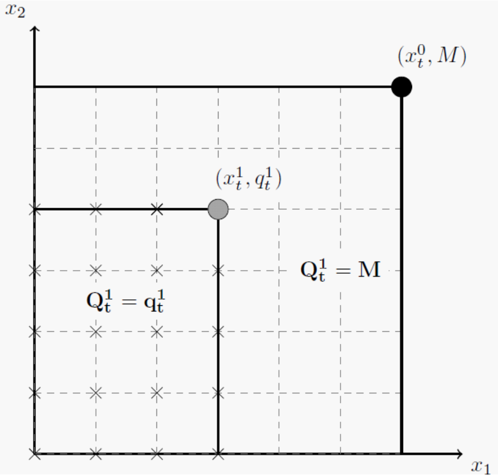
Approximation represented by a mixed integer program

$$
\begin{array}{llcll}
Q^H(x) = \max & \varphi & & & \\
\text{s.t.} & \varphi & \leq & q^h + (M - q^h) y^h, & h = 1, 2, \ldots, H, \\
& x_k & \geq & (x_k^h + 1) z_k^h, & k = 1, 2, \ldots, n, \\
& & & & h = 1, 2, \ldots, H, \\
& \sum_{k=1}^n z_k^h & \geq & y^h, & h = 1, 2, \ldots, H, \\
& y^h & \in & \{0, 1\}, & h = 1, 2, \ldots, H, \\
& z_k^h & \in & \{0, 1\}, & k = 1, 2, \ldots, n, \\
& & & & h = 1, 2, \ldots, H.
\end{array}
$$

Deterministic MIDAS

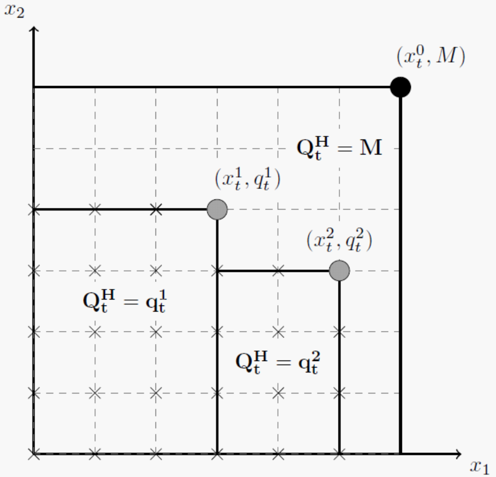
1. Set $H = 1$, $Q_t^H(x) = M$, an upper bound on $V_t(x)$, $t = 1, 2, \ldots, T+1$
2. Forward pass: Set $x_1^H = \bar{x}$. For $t = 1$ to T,
   1. Solve $\max_{u \in U(x_t^H)} \left\{ r_t(x_t^H, u) + Q_{t+1}^H(f_t(x_t^H, u)) \right\}$ to give u_t^H.
3. If there is some $h < H$ with $x_t^H = x_t^h$ for $t = 1, 2, \ldots, T+1$ then stop.
4. Backward pass: Update $Q_{T+1}^H(x)$ to $Q_{T+1}^{H+1}(x)$ by adding $q_{T+1}^H = R(x_{T+1}^H)$ at point x_{T+1}^H. Then for every $t = T$ down to 1 do
   1. Solve $\varphi = \max_{u \in U(x_t^H)} r_t(x_t^H, u) + Q_{t+1}^{H+1}(f_t(x_t^H, u))$;
   2. Update $Q_t^H(x)$ to $Q_t^{H+1}(x)$ by adding $q_t^H = \varphi$ at point x_t^H;
5. Increase H by 1 and go to step 2.

Example in two dimensions



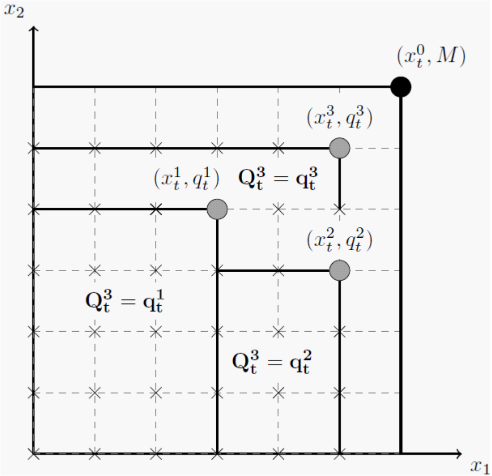
New point added at x_t^1 with value q_t^1.

Example in two dimensions



New point added at x_t^2 with value q_t^2 (Here $q_t^2 > q_t^1$).

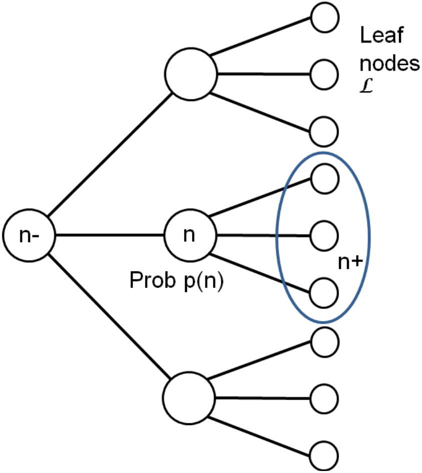
Example in two dimensions



New point added at x_t^3 with value q_t^3 (Here $q_t^3 > q_t^2 > q_t^1$).

Summary

Multistage stochastic algorithm uses a scenario tree



Tree formulation of stochastic control problem

$$
\begin{array}{lll}
\text{MSPT:} & \max & \sum_{n\in\mathcal{N}\setminus\{0\}} p(n) r_n(x_{n-}, u_n) + \sum_{n\in\mathcal{L}} p(n) R(x_n) \\
& \text{s.t.} & x_n = f_{n-}(x_{n-}, u_n, \xi_n), \\
& & x_0 = \overline{x}, \\
& & u_n \in U(x_n), \\
& & x_n \in X_n.
\end{array}
$$

DP recursion is:

$$
\begin{aligned}
V_n(x_n) &= \sum_{m\in n+} \frac{p(m)}{p(n)} \max_{u\in U(x_n)} \left\{ r_m(x_n, u) + V_m\left(f_n(x_n, u, \xi_m)\right)\right\} \\
V_n(x_n) &= R(x_n),\ n \in \mathcal{L},
\end{aligned}
$$

where we seek a policy that maximizes $V_0(x_0)$.

MIDAS algorithm for stochastic dynamic programming

1. Set $Q_n^1(x) = M$, for every $n \in \mathcal{N} \setminus \mathcal{L}$;
2. For $H = 1, 2, \ldots,$
   - set $Q_n^H(x) = R(x)$, for every $n \in \mathcal{L}$;
   - perform a forward pass then a backward pass.

Forward pass

Set $x_0^H = \overline{x}$, and $n = 0$. While $n \notin \mathcal{L}$:

1. Sample $m \in n+$ to give ξ_m^H;
2. Solve $\left\{ \max_{u \in U(x_n^H)} \left\{ r_m(x_n^H, u) + Q_m^H(f_n(x_n^H, u, \xi_m^H)) \right\} \right\}$ to give u_m^H;
3. Set $n = m$.

Backward pass

For the particular node $n \in \mathcal{L}$ at the end of step 1 update $Q_n^H(x)$ to $Q_n^{H+1}(x)$ by adding $q_n^{H+1} = R(x_n^H)$ at point x_n^{H+1}. While $n > 0$

1. Set $n = n-$;
2. Compute

$$\varphi = \sum_{m \in n+} \frac{p(m)}{p(n)} \max_{u \in U(x_n^H)} \{ r_m(x_n^{H+1}, u) + Q_m^{H+1}(f_n(x_n^{H+1}, u, \xi_m)) \}$$

3. Update $Q_n^H(x)$ to $Q_n^{H+1}(x)$ by adding $q_n^{H+1} = \varphi$ at point x_n^{H+1};
4. Increase H by 1 and begin a new forward pass.

Sampling property

FPSP: For each $n \in \mathcal{L}$, with probability 1

$$\left|\left\{H : \xi_n^H = \xi_n\right\}\right| = \infty.$$

Theorem

If step 1 of forward pass satisfies FPSP then sampled MIDAS converges almost surely to an optimal solution.

Summary

Outer approximation of continuous monotonic functions

Given a continuous nondecreasing function $Q(x) \leq M$, and a finite set of values

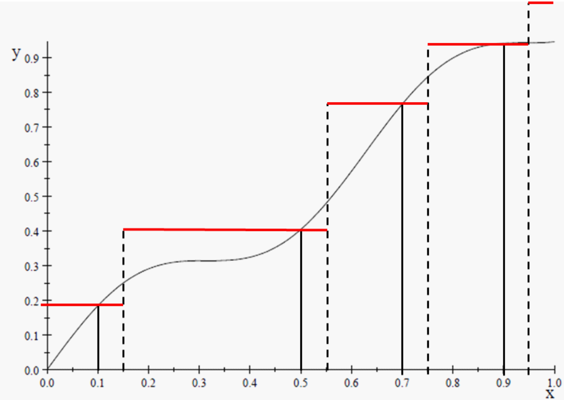
$$Q(x^h) = q^h,\ h = 1, 2, \ldots, H,$$

approximate $Q(x)$ by a piecewise constant function $Q^H(x)$ so that for every x

$$Q(x) \leq Q^H(x) + \varepsilon$$

- q^h is a real number and Q^H is a function;
- $Q(x)$ is assumed monotonic to guarantee that $Q(x) \leq Q^H(x) + \varepsilon$ for every x.

Example



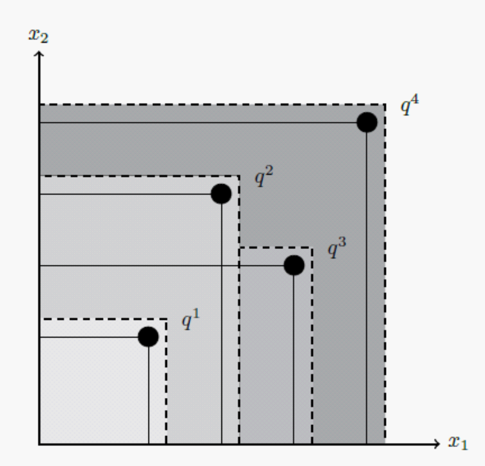
Approximation of $Q(x)$ at points $x^h = 0.1, 0.5, 0.7, 0.9$, and $\delta = 0.05$. $Q^H(x)$ shown in red is upper semicontinuous, and is an upper bound on $Q(x) - \varepsilon$.

MIP approximates a continuous monotonic function

Assume that

$$
\begin{aligned}
Q^H(x) = \max \quad & \varphi \\
\text{s.t.} \quad & \varphi \leq q^h + (M - q^h) y^h, && h = 1, 2, \ldots, H, \\
& x_k \geq (x_k^h + \delta) z_k^h, && k = 1, 2, \ldots, n, \\
& && h = 1, 2, \ldots, H, \\
& \textstyle\sum_{k=1}^n z_k^h \geq y^h, && h = 1, 2, \ldots, H, \\
& y^h \in \{0, 1\}, && h = 1, 2, \ldots, H, \\
& z_k^h \in \{0, 1\}, && k = 1, 2, \ldots, n, \\
& && h = 1, 2, \ldots, H.
\end{aligned}
$$

Example in two dimensions



Contour plot of $Q^H(x)$ when $H = 4$. Circled points are x^h, $h = 1, 2, 3, 4$. Darker shading indicates increasing values of $Q^H(x)$ that equals $Q(x^h)$ in each region containing x^h, $h = 1, 2, 3, 4$.

Forward pass

Set $x_0^H = x_0$, and $n = 0$. While $n \notin \mathcal{L}$:

1. Sample $m \in n+$ to give ξ_m^H;
2. Solve $\max_{u \in U(x_n^H)} \left\{ r_m(x_n^H, u) + Q_m^H(f_n(x_n^H, u, \xi_m^H)) \right\}$ to give u_m^H;
3. If $\left\| f_n(x_n^H, u_m^H, \xi_m^H) - x_m^h \right\|_\infty < \delta$ for $h < H$ then set $x_m^{H+1} = x_m^h$, else set $x_m^{H+1} = f_n(x_n^H, u_m^H, \xi_m^H)$;
4. Set $n = m$.

Backward pass

For the particular node $n \in \mathcal{L}$ at the end of forward pass update $Q_n^H(x)$ to $Q_n^{H+1}(x)$ by adding $q_n^{H+1} = R(x_n^{H+1})$ at point x_n^{H+1}.
While $n > 0$

1. Set $n = n-$;
2. Compute
$$\varphi_n = \sum_{m \in n+} \frac{p(m)}{p(n)} \max_{u \in U(x_n^H)} \left\{ r_m(x_n^{H+1}, u) + Q_m^{H+1}(f_n(x_n^{H+1}, u, \xi_m)) \right\}$$
3. Update $Q_n^H(x)$ to $Q_n^{H+1}(x)$ by adding $q_n^{H+1} = \varphi_n$ at point x_n^{H+1};

Convergence

FPSP: For each $n \in \mathcal{L}$, with probability 1

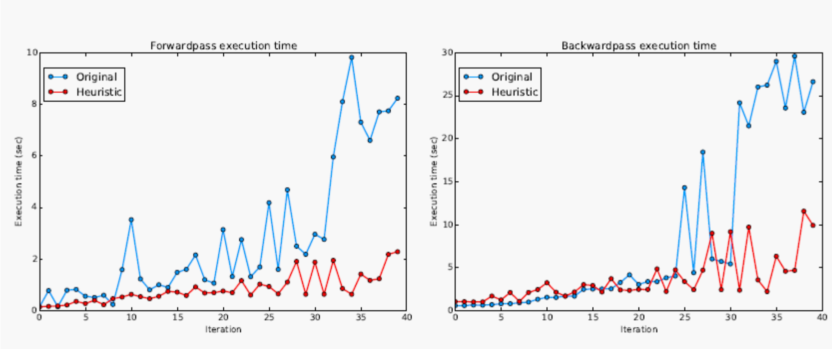
$$\left|\left\{H : \xi_n^H = \xi_n\right\}\right| = \infty.$$

Theorem

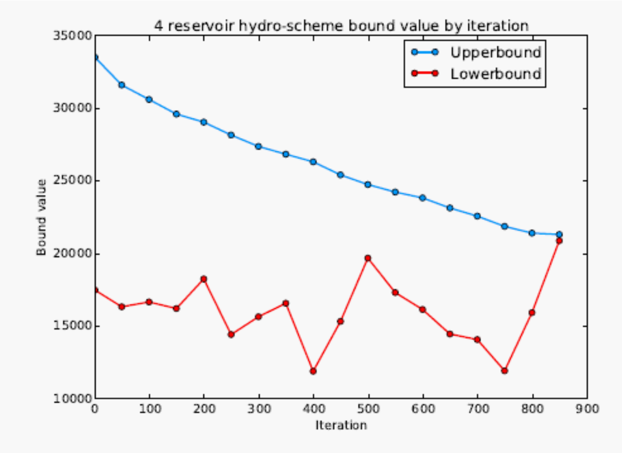
If step 1 of forward pass satisfies FPSP then sampled MIDAS converges almost surely to a $(T+1)\varepsilon$-optimal solution.

Summary

Computational results (4 reservoir river chain)



MIPS become expensive as points added. “Heuristic” eliminates dominated points and uses integer L-shaped method to solve subproblems.



MIDAS bounds with 4 periods, 4 reservoirs with levels 0-200, and 3 price states.

Example: Continuous state model for river chain

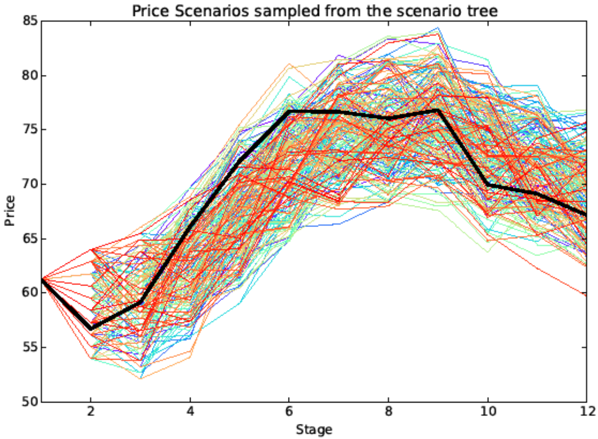
$$\begin{bmatrix} s_{t+1} \\ \pi_{t+1} \end{bmatrix} = \begin{bmatrix} s_t - v_t - l_t + \omega_t \\ \alpha_t \pi_t + (1 - \alpha_t) b_t + \eta_t \end{bmatrix},$$

Here ω_t is (random) reservoir inflow, η_t is error term for price model, so $\boldsymbol{\xi}_t = [\ \omega_t \quad \eta_t\ ]^\top$ and $\mathbf{u} = [\ v \quad l\ ]^\top$ release and spill. Reward in stage t is

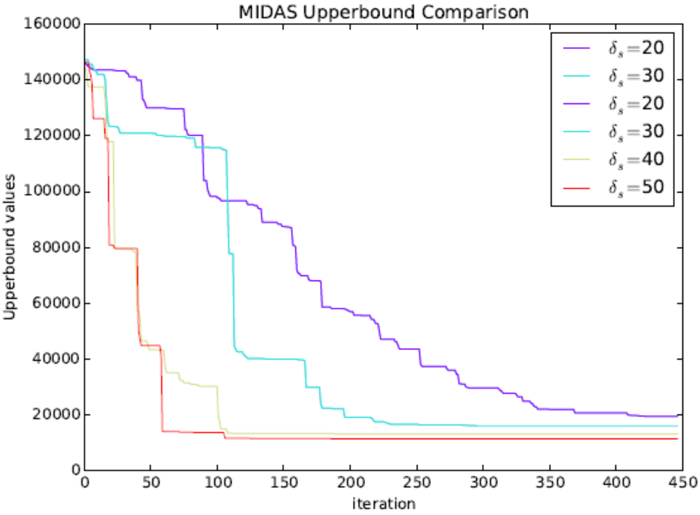
$$r_t(s, \pi, v, l, \omega_t, \eta_t) = \pi \sum_i g_i(v),$$

from released energy $g(v)$ sold at price π, and $U(\mathbf{x}) = \{(v, l) : v \in U_0,\ v + l \in [0, s]\}$.

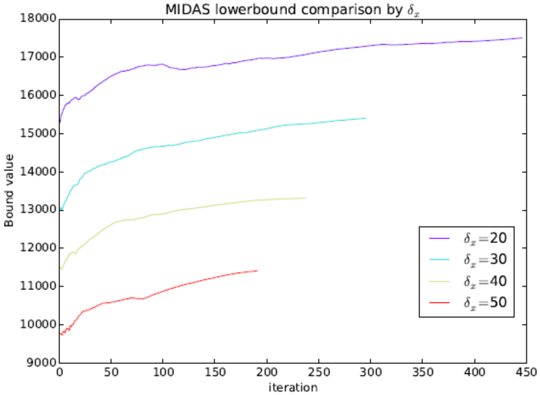
Price scenarios sampled from AR1 model



Epsilon upper bounds for two-reservoir problem



Epsilon lower bounds for two-reservoir problem



Summary

Conclusions

- SDDP has proved very successful in hydrothermal scheduling;
- Sampled trajectories reduce DP computation;
- Convexity limits applications;
- MIDAS is an attempt to extend these features to more general stochastic dynamic programs in the hope of making them tractable by solving small MIPs;
- Computational challenge is generation of large MIPS before convergence;
- Convergence wp1 can be shown for continuous monotonic value functions, and integer-valued state variables;

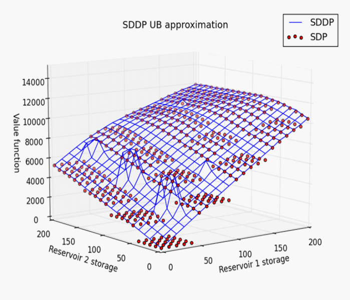
References

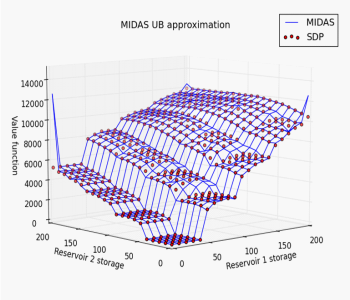
- Boomsma, T.K., Juul N. and Fleten S.E. Bidding in sequential electricity markets: The Nordic case *EJOR*, 238(3):797–809, 2014.
- Braathen, J and Eriksrud, A.L. Hydropower Bidding Using Linear Decision Rules. *Institutt for industriell økonomi og teknologiledelse*, 2013.
- Fleten, S.E. and Kristoffersen, T.K. Stochastic programming for optimizing bidding strategies of a Nordic hydropower producer. *EJOR*, 181(2):916–928, 2007.
- Löhndorf, N., Wozabal, D. and Minner, S. Optimizing Trading Decisions for Hydro Storage Systems Using Approximate Dual Dynamic Programming. *INFORMS*, 61(4):801-823, 2013.

References

- Pereira M.V.F. and Pinto L.M.V.G. Multi-stage stochastic optimization applied to energy planning. Math Prog, 52(1):359–375, 1991.
- Philpott, A.B., Wahid, F. and Bonnans, J.F. Mixed Integer Dynamic Approximation Scheme. Optimization Online May 2016.
- Pritchard, G. and Zakeri, G. Market offering strategies for hydroelectric generators. *Oper. Res*, 51(4):602–612, 2003.
- Zhou, J. Ahmed, S. and Sun, A. Nested decomposition of multistage stochastic integer programs with binary state variables, Optimization Online May 2016.

SDDP and MIDAS for stochastic MIPs





True value function shown in red. SDDP and MIDAS give different outer approximations when true value function is not concave.