A simulations-and-regressions algorithm with application to hydropower storage

Workshop on Electricity, Energy and Commodities Risk Management
Fields Institute

M. Denault, J.-G. Simonato and L. Stentoft

HEC Montréal

14-16 August 2013
Project Overview

- Initial motivation: the modeling and control of energy production and storage.
- Pure profit optimization: attempt to play storage and variable prices at their best. No risk management *per se*.
- Complex optimization problems: optionnality, stochastic state variables, multiscale seasonalities, long-term decisions.
Project Overview

- Development of a dynamic programming approach based on simulations and regressions.
- The techniques of DP with simulations and regression have become central in financial engineering to solve financial option problems.
- We’re valuing an energy instrument that is a set of dependent options; and to value it, we need to time the sale decisions at best.
Focus here on one application (hydropower) and two state variables: the exogenous spot price of power, and the endogenous water level.

- The endogenous (control-dependent) variable is a key point.
- Two main ideas:
  - Graft the endogenous state variable onto the simulation paths of the exogenous state variable, building paths of “optimal” water levels.
  - Apply a “backwash” technique to both deal with operational limits and avoid clustering in the endogenous variable space.
Related literature

Closely related literature, on gas storage.

- Boogert and De Jong (2008): probably the first simulations-and-regressions approach to gas storage, but the endogenous variable is discretized.
The **simple** setup, discussed today, includes

- a hydro power production facility which includes storage;
- the possibility to buy or sell a limited but fixed amount of power at each period;
- purchases of power increase the water level (see below);
- all transactions are at the spot price, which is stochastic;
- storage is of course bounded above and below.
A more complete problem setup would include

▶ a variable local demand at a constant price, which must be satisfied;
▶ purchases and sales are on a neighbour market, with stochastic prices;
▶ purchases of power help keep water behind the dam (but don’t actually increase the level);
▶ water inflows are stochastic.
Setting the problem as a dynamic program

- The goal is to maximize expected net profit over a finite horizon $[0, T]$.
- Natural setup for dynamic programming: knowing the optimal policy from $t + 1$ to $T$, find the optimal policy from $t$ to $T$ by identifying the best policy between $t$ and $t + 1$.
- Backward solution is then possible, from time $T$ to time 0, given the final boundary condition.
The water level is an endogenous state variable: the production decision at $t$ changes the state of the system at $t+1$.

Compare: the American option has only an exogenous state variable, the stock price.

The water levels follow the state equation

$$L_{t+1} = h(u_t; L_t);$$

where $u$ is a sales decision and $L_t$ is the water level at time $t$. 
Dynamic programming recursion

Thanks to the optimality principle of dynamic programming, we can compute the value function recursively as

\[
V_t(S_t, L_t) = \sup_{u_t \in U(S_t, L_t)} \left\{ \pi_t(u_t; S_t, L_t) + \mathbb{E}_t \left\{ V_{t+1}(S_{t+1}, h(u_t; L_t)) \right\} \right\}
\]

where \( u \) is the decision variable, \( \pi_t \) is the payoff function on the period from \( t \) to \( t + 1 \), the expectation is conditional on time \( t \) information.

(The value function is the (monetary) value of being in a certain state at a certain time, assuming that the best non-anticipative decisions will be made until the end of time.)
Traditional solution approach for the recursion equation

- The traditional way to solve the continuous time, continuous state variables DP is to discretize all state variables and time.
- This is the technique we use for benchmarking.
- Subject to the curse of dimensionality: beyond a few state variables, the technique is very time-consuming, or even untractable.
A simulations-and-regressions approach

For its simplicity, flexibility and ability to handle greater numbers of state variables, we prefer the dynamic programming approach of Monte Carlo simulations and (simple linear) regressions.

- Monte Carlo simulations are used to generate ahead of time a set of scenarios for the exogenous stochastic variable (e.g. spot price)
- Decisions are discretized.
- For each decision, the profit function is approximated by regressing the profits on the state variable values (for all paths).

What about the endogenous variable (water level)?
Solution through simulations-and-regressions

Let the value function be known at \( t + 1 \)

\[
V_{t+1}\left( S_{t+1}^{(k)}, L_{t+1}^{(k)} \right)
\]

for each spot price path \( k \in 1, \ldots, K \).

Define the \textit{backward} state equation

\[
\leftarrow h(u_t; L_{t+1}) = L_t
\]

and the water level at time \( t \) which \textit{depends on} \( u_t \)

\[
L_t^{(k)}(u_t) = \leftarrow h\left( u_t; L_{t+1}^{(k)} \right)
\]
We can regress

\[ \pi_t(u_t; S_t^{(k)}, L_t^{(k)}(u_t)) + V_{t+1}(S_{t+1}^{(k)}, L_{t+1}^{(k)}) \]

on \( (S_t^{(k)}, L_t^{(k)}(u_t)) \)

for each possible decision \( u_t \). Note that the “antecedent levels” \( L_t^{(k)}(u_t) \) are functions of the decision.

We obtain a regression surface for each possible discrete decision.
What is an adequate path-k, time-t water level $L^{(k)}_t$?
Creating Paths of Water Levels

- Endogenous water level variable cannot be simulated ahead of time, like the spot prices.
- However, the value expectations must rely on optimal paths.
- We build water level paths backwards, using the regression surfaces.
- These **water level paths** are not actual simulations, but each is *matched* with a **spot price path**.
Creating Paths of Water Levels: forward-optimal paths

- So, which time-t water level $L_t$ is the “right” level, given a time-$(t+1)$ water level $L_{t+1}$?
- Certainly not the level with the highest value, that would be “backward optimal”.
- We need a time-t level that “forward optimally” leads to the (known) time-$(t+1)$ level.
- The regression surfaces are computed already, so just use them repeatedly (small numerical overhead)
What is an adequate path-k, time-t water level $L^{(k)}_{t}$?
Two problems crop up with this “fausse-simulation” technique.

- Problem 1: little control over the building of the water level paths, so water levels can go out-of-bounds (leakage) and can cluster. (And they do!)
- Problem 2: need to take account of the water level operational bounds wisely. We *do* need information about crossing the bounds.

Solution:

- Add a penalty term for violations of the dam upper and lower levels.
- Let water level paths go out-of-bounds, and use that info in the regressions.
- When a path goes too far out-of-bounds, *backwash* it randomly to the feasible area, thereby smoothing clusters.
Clusters, Bounds and Backwash

Solution:

- Add a penalty term for violations of the dam upper and lower levels.
- Let water level paths go out-of-bounds, and use that info in the regressions. This takes care of problem 2.
- When a path goes too far out-of-bounds, backwash it randomly to the feasible area, thereby smoothing clusters. This takes care of problem 1.
Summary of the algorithm

Initialization:

1. Choose a set of basis functions for the state variables, $S_t$ and $L_t$;
2. Randomly generate $K$ paths for the exogenous variable $S_t$, $(t = 0, 1, \ldots, T)$;
3. Randomly generate $K$ time-$T$ levels of the endogenous variable $L_T$, within the range $[L_{\text{min}}, L_{\text{max}}]$;
4. Compute time-$T$ values according to a boundary condition.
**Backward recursion**: for all times from \( t = T - 1 \) to \( t = 0 \):

1. Compute the regression surfaces \( \tilde{V}^u(S, L) \), \( u \in \{+1, 0, -1\} \) using a payoff with penalty for going out of bounds.

2. For each of the \( 3K \) candidate levels \( L_t^{(k)} \), compute a forward optimal decision.

3. Associate a level \( L_t^{(k)} \) for each path \( k \), according to the decisions in the above step. If \( L_t^{(k)} \) is too far out of bounds, randomly reassign it to a random, acceptable water level (the backwash technique).

4. Compute the \( K \) values \( V_t(S_t^{(k)}, L_t^{(k)}) \) as a sum of payoffs until time \( T \) along path \( (k) \). In the case of paths whose level has been reassigned in step 3, use instead the value on the regression surface.

**Out-of-sample tests**: retain solely the regression parameters.
Sources of approximations come from the simulation, the regression on basis, the backwash.

Tsitsiklis-Van Roy vs Longstaff-Schwartz approaches.

Given the backwash procedure, this algorithm is in fact hybrid of TVR and LS.
Illustration with a simple example

We consider a simple but interesting case of four half-days.

- Average price is 50$, except during the 2nd and 5th periods (average of 30$). Prices are serially independent.
- Three regimes: buy, sell, do nothing.
- Number of simulations in the learning phase: 50 000.
- Comparison is done with a fully discretized DP as benchmark.
Regression surfaces (g=sell, k=wait, r=buy) for period 3
Regression surfaces \((g=\text{sell}, k=\text{wait}, r=\text{buy})\) for period 3
Benchmark policies vs Simulation and Regression policies
Spot prices on x-axis; water levels on y-axis.
Numerical results on a larger model

- Algorithm is run on a problem with the same state variables (price and water level) but 16 weeks long at two periods per day (224 time steps)
- Spot price follows a geometric brownian motion (we want to benchmark!)
- Daily, weekly, monthly seasonalities on the spot prices.
- We do out-of-sample testing against a finely discretized dynamic program. The benchmark value is 247 500 $.
- Obtain results within two percent of the optimal value:

<table>
<thead>
<tr>
<th>Npath</th>
<th>Mean</th>
<th>Stdv</th>
</tr>
</thead>
<tbody>
<tr>
<td>25000</td>
<td>242 765 $</td>
<td>277 $</td>
</tr>
<tr>
<td>50000</td>
<td>242 897 $</td>
<td>161 $</td>
</tr>
<tr>
<td>75000</td>
<td>242 900 $</td>
<td>128 $</td>
</tr>
</tbody>
</table>
Numerical results: pretty good or pretty bad?

Pretty good or pretty bad? Well, both...

- The quality of the results (sim-and-reg vs benchmark) is influenced by the bases and by the backwash procedure.
- Polynomial bases do their best, but are clearly imperfect. This is however could be rather good news for the backwash technique.
- Note that sim-and-reg and benchmark results are similarly impacted by the discretization of the decision.
Conclusions

- The classical simulations-and-regressions technique is extended to a more general problem with an endogenous (control-dependent) state variable.
- Neither the exogenous nor the endogenous variables are discretized.
- Simulation based, so very flexible with respect to the modeling of the exogenous, stochastic variables.
Future and on-going work

- Introduce more exogenous and endogenous variables.
- Introduce decisions that kick in only after a number of periods.
- Risk management.
- Non-energy applications.


