Particle Filtering Approaches for Dynamic Stochastic Optimization

John R. Birge
The University of Chicago Booth
School of Business

Joint work with Nicholas Polson, Chicago Booth.
Themes

- Dynamic stochastic optimization models face the curse of dimensionality
- Randomization can overcome some dimensionality effects
- With sufficient symmetry, the results can be quite effective
- Keys are conditioning properly on outcomes, appropriate re-sampling and split sampling, and uses of Gaussian mixtures
Outline

• Basic problem
• Problems of estimation
• Particle filters and sequential methods for simulation
• Optimization by slice sampling
• Dynamic Gaussian problems
• Extensions to Gaussian mixtures
General Problem:

Find a set of controls, $u_t, t = 0, \ldots, T$, to

$$
\min_{u_t \in U_t \cap A_t; x_{t+1} = g_t(x_t, u_t)} \mathbb{E}\left[ \sum_{t=0}^{T-1} f_t(x_{t+1}, u_t) \right]
$$

where $A_t$ is $\Sigma_t$-measurable (nonanticipative),
$U_t$ represents the possibility of additional constraints on the controls,
$g_t$ represents state dynamics,
$f_t$ represents the cost of the current action and next state transition.

Challenges:
Rapid expansion of states in dimension of $x_t$ in each period,
Exponential growth of trees over time.
First Problem: What is $x_t$?

- Example: hidden Markov model
- Example: We begin with inventory $x_0$ (which may be uncertain)
- We make observations $y_1, \ldots, y_T$ based on reported sales and new inputs
- What is the distribution of actual inventory $x_t$ at any time $t$?
Bayesian Interpretation

• Suppose a distribution on \( x_0: f(x_0) \)
• Assume some transition: \( p(x_t|x_{t-1}) \)
• Observations: \( q(y_t|x_t) \)
• Goal: find \( P(x_t|y_1,\ldots,y_t (\ldots y_T)) \)
• Idea: Use Bayes Rule to find the conditional probability:

\[
P(x_t|y^t) = P(x_t,y^t)/P(y^t) \\
\propto \prod_s q(y_s|x_s)p(x_s|x_{s-1})f(x_0)
\]
Particle Methods

- Idea: Keep a limited number ($N$) of particles in each period
- Propagate forward to obtain new points
- Re-sample (re-weight) backwards to get consistent weights
- Re-balance if weights are low

General Processes

• If the observation and transition functions, \( q \) and \( p \) are quite general, then \( P(x_t|y_t) \) may not be available analytically (E.g., no sales if stocked out, maximum limit, bulky demand)

• Alternatives:
  – Markov chain Monte Carlo (MCMC)
  – Particle methods
Sequential Importance Resampling (SIR) Method

1. **Propagate.** Draw $x_{t+1,j}, j = 1, \ldots, N$ from $f(x_{t+1,j}|x_{t,j})$.

2. **Re-weight.** Update weights, $\hat{w}_{t,j}, j = 1, \ldots, N$, to
   \[
   \hat{w}_{t+1,j} = \frac{w_{t,j} g(y_{t+1}|x_{t+1})}{\sum_{j=1}^{N} \hat{w}_{t,j} g(y_{t+1}|x_{t+1})}
   \]
   for $j = 1, \ldots, N$.

3. **Re-sample.** If a re-sampling condition is satisfied, draw $N$ new samples $x'_{t+1,j}, j = 1, \ldots, N$ from $x_{t+1,j}, j = 1, \ldots, N$ with probability distribution given by $\hat{w}_{t+1,j}, j = 1, \ldots, N$; let $x_{t+1,j} = x'_{t+1,j}$ and $\hat{w}_{t+1,j} = 1/N$ for $j = 1, \ldots, N$. 
Tractable Case: LQG

- Linear, quadratic Gaussian (LQG) Model
- Implications: All of the distributions are normal
- Can derive analytical formulas (Kalman filter):
  \[ y_t = H x_t + v_t; \quad x_t = F x_{t-1} + u_t; \quad v \sim N(0,Q); \quad u \sim N(0,R) \]

Iterations:
  \[ w_t = y_t - Hx'_t; \quad S = HP'HT + R; \]
  \[ K = P'HTS^{-1}; \quad x''_t = x'_t + Kw_t \]
  \[ P'' = (I-KH)P'; \quad x'_t = Fx''_{t-1}; \quad P' = FP''F^T + Q \]
Kalman Prediction

Inventory over 10 periods:
- Actual (*)
- Observed (+)

Kalman estimate:
\[ x_t \sim N(x''_t, P''_t) \]

Here:
\[ x_{10} \sim N(13.8, 0.62) \]
Comparison to MCMC/Particle Filter

- Idea: construct a sample distribution that fits the data using only information about one-step transitions
- Goal: maintain a constant number of samples across time instead of explosively growing tree
- Can it work? Why?
Method Structure

**Propagate**
- $w_t^1, x_t^1$
- $w_t^2, x_t^2$
- $\ldots$
- $w_t^N, x_t^N$

**Re-sample:**
- $x_{t+1}^1$
- $x_{t+1}^2$
- $\ldots$
- $x_{t+1}^N$
- $y_{t+1}$
- $w_t^1$
- $w_t^2$
- $\ldots$
- $w_t^N$
Cum. Results

• $N=50$
• Compare to Kalman Cumulative
Particle Frequency

- $N=50$
- Particle (+)
- Kalman (*)
Other Uses of Particles

- Particle filtering: finding $x_t$ from $y^t = (y_1, \ldots, y_t)$
- Particle smoothing: finding $x_t$ from $y^T = (y_1, \ldots, y_T)$
- Particle learning: (Polson et al.)

Find parameters $\theta_t$ at the same time to explain the model.

- Simulation
Extensions to Optimization

**Problem:** find $x$ to

$$\min_x f(x) \text{subject to } g(x) \geq 0$$

Pincus (1968): enough to sample from:

$$\pi_\kappa(x) \propto \exp\{-\kappa(f(x))\}$$

with constraints (e.g., Geman/Geman (1985)):

$$\pi_{\kappa,\lambda}(x) \propto \exp\{-\kappa(f(x) + \lambda g(x))\}$$

**Extension:** Augment with latent variables $(\lambda, \omega)$ then we can write

$$\pi_{\kappa,\lambda}(x) \propto \exp\{-\kappa(f(x) + \lambda g(x))\}$$

$$= e^{ax} E_\omega \{ e^{-\frac{x^2}{2}} \} e^{bx} E_\lambda \{ e^{-\frac{x^2}{2} \lambda} \}$$

where the parameters $(a,b)$ depend on $(\kappa, \lambda \kappa)$ and the functional forms of $(f(x), g(x))$. 
Basic Optimization via Simulation

• Pincus results:
• Model: min $F(x)$
• Create a distribution $P_{\lambda}(x) \propto \exp(-\lambda F(x))$
• Then:

$$\lim_{\lambda \to \infty} E_{P_{\lambda}}(x) = x^*$$
Markov Chain Simulation

- From $x_t$ to $x_{t+1}$
- Sample over
  \[ \exp(-\lambda F(\mu x_t + (1-\mu) x_{t+1})) \]
- Choose directions to obtain mixing in the region
Example

• Rosenbrock function:

\[ \phi(x = (x_1, x_2)) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2, \]

which has a global minimum at \( x^* = (1, 1) \).

Specification for \( f \):

\[ x_t = \{ u \text{ if } v \preceq e^{-\kappa(\phi(u)/\phi(x_{t-1}))}; x_{t-1} \text{ otherwise}, \} \]

where \( u \sim U([-4, 4]^2) \), a uniform distribution over \([-4, 4] \times [-4, 4] \), and \( v \sim U([0, 1]) \), a standard uniform draw.

Note: corresponds to the Metropolis-Hastings method of Markov Chain Monte Carlo using the distribution proportional to \( e^{-\kappa \phi(x)} \) for the acceptance criterion.
Test for Rosenbrock

- $T=5000$ iterations
- $N=500$ replications
- Re-sample when:
  - Effective Sample Size $< 250$
- $\kappa=2, 10, 100$
Rosenbrock Graphs

\[ \kappa = 2 \quad 10 \quad 100 \]
Other Examples: Rastigrin
Himmelblau

Himmelblau: kappa=1

Himmelblau: kappa=2

Himmelblau: kappa=5

Himmelblau: kappa=10
Shubert

- Shubert: kappa=0.1
- Shubert: kappa=0.5
- Shubert: kappa=1
- Shubert: kappa=5
Combine Particles and Simulation

• Suppose $N$ particles are used
• Consider a 2-stage problem
• $f(x, \xi) = c(x) + Q(x, y, \xi)$
• Distribution on $x, y_i, \xi$:
  
  $P(x, y_1, \ldots y_N, \xi) \propto \prod_{i=1}^{N} (c(x) + Q(x, y_i, \xi_i))$

Result: $E_P (x) \rightarrow x^*$
Implementation

• Propagation forward on y’s:

\[ q(\xi_j | x_1) \propto \{ c(x) + Q(\xi_j, y_j, x) \} p(\xi_j) \]

where \( y_j = \text{argmin}(c(x) + Q(\xi_j, y_j, x)) \)

• \( p(x | \xi_1, y_1, \ldots, \xi_N y_N) \)

\[ \propto \prod_{j=1}^N \{ c(x) + Q(\xi_j, y_j, x) \} p(\xi_j) \]

The particles concentrate the distribution on \( x^* \) as in the deterministic optimization.
Example

• From B/Louveaux, Chapter 1 (Farmer):
Dynamic Models

• General Idea:

Create a distribution over decision variables and random variables

Use MCMC to sample – with sufficient number of particles or “annealing parameter” – marginal mean on optimum

• Extension for dynamics:

Keep constant sample particle numbers

Convergence in fixed number of particles per stage?
Example: Linear-Quadratic Gaussian

• All distributions are available in closed form

\[ x_t = Ax_{t-1} + Bu_{t-1} + v_t \]

• All normal distributions with risk-sensitive objective:

\[ -\log E(\exp(-(\theta/2)(x_2^TQx_2 + u_1^TRu_1))) \]

where \( u_1 \) is control in the first period

Note: Equivalent to robust formulation

• Now, all can be found in closed form
Results for LQG

• Mean of \( u \): \( m_{u_1} \)

\[ m_{u_1} = L_1(\theta) x_1, \]

where \( L_1 \) is the result from the Kalman filter

• Variance of \( u \): \( V_u(N) \) where

\[ V_{u_1}(N) \rightarrow 0 \text{ as } N \rightarrow \infty \]

\[ m_{u_1}(\theta) \rightarrow u_1^* \text{ as } \theta \rightarrow 0 \]

where \( u_1^* \) is the two-stage risk-neutral solution.
Extensions to T Stages

• Can derive $S_t^\theta$ recursively to show that:

$$p(x_t|x_{t-1}, u_{t-1}) = K \exp\left(-\frac{1}{2} \times \left(\sum_{s=1}^{t-1} (x_t^T Q_s x_s + u_s^T R_s u_s) + x_t^T S_t^\theta x_t\right)ight)$$

$$+ \left(x_t-Ax_{t-1}-Bu_{t-1}\right)^T V_{t-1}\left(x_t-Ax_{t-1}-Bu_{t-1}\right)$$

For particle methods, $j=1 \ldots N$:

Each $x_t^j$ drawn consistently from this distribution
General Method Structure

Propagate:
- $w_t^1, x_t^1$ → $x_{t+1}^1$
- $w_t^2, x_t^2$ → $x_{t+1}^2$
- $\vdots$
- $w_t^N, x_t^N$ → $x_{t+1}^N$

Re-sample:
- $w_t^1, x_t^1$ → $u_t^1$
- $w_t^2, x_t^2$ → $u_t^2$
- $\vdots$
- $w_t^N, x_t^N$ → $u_t^N$

© JR Birge
I-Sim Workshop, Purdue U., July 2015
General Result

- If the propagation step is unbiased, then
  \[ E(u_l) \rightarrow u_l^* \]
- The unbiased result is possible in LQG from the recursive structure.
- Harder to obtain in more general systems (without additional future sampling)
Two-Stage: Compare Direct MC

Low Risk Aversion           High Risk Aversion

© JRBirge  
I-Sim Workshop, Purdue U., July 2015
Three Stage Example
Generalizing Objectives

• Direct: risk-sensitive exponential utility:
  \[ e^{-\theta Q(x,y,\xi)} - Q \text{ quadratic} \]

• Objective: \( e^{-\theta |y-d|} \)

\[ E_\lambda [\exp((-\theta/2\lambda_j)(y_j-d)^2)] \]

where \( \lambda \sim \text{Exp}(2) \)

Also, use:

\[ E[ e^{-\theta |y-d|} ] \approx 1 + \theta |y-d| + O(1) \]
Further Generalizations

• Use objective approximations that preserve normal forms
• Extend LQG-like results to this general framework
• Find other frameworks where unbiased future samples are available
• Obtain sensitivity results on bias from “imperfect” future sampling
• Use optimization to pick ML instead
Conclusions

• Techniques from MCMC can be used effectively for optimizing to reduce dimension effects

• Results work well in situations with sufficient symmetry

• Keys are to effectively use re-sampling, split (and slice) sampling, and latent variables
Other Questions to Consider

• How can learning and parameter estimation be combined with optimization?
• What is the range of problems that can be approached in this manner?
• What are the best uses of optimization technology as part of this process?
Thank you!

• Questions?