Resampling Strategy in Sequential Monte Carlo for Constrained Sampling Problems

Chencheng Cai

Washington State University

Oct. 2024

State-Space Model and Sequential Monte Carlo

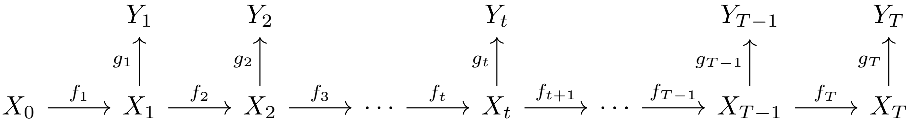
Constrained Sequential Monte Carlo and Pilot Methods

Examples

Notations

- $t = 0, 1, \ldots, T$: Time index
- X_t, Y_t: random variable at time t
- x_t, y_t: value of the random variables
- $X_{t:s}$: $(X_t, X_{t+1}, \ldots, X_s)$
- $\boldsymbol{X}_t$: $X_{0:t}$
- $p(\cdot)$: the general probability notation

State-Space Model (SSM)



- Latent Variables: $X_0, X_1, \ldots, X_T$.

$$\begin{aligned} p(X_0 = x_0) &= f_0(x_0), \\ p(X_t = x_t \mid \boldsymbol{X}_{t-1} = \boldsymbol{x}_{t-1}) &= f_t(x_0, \ldots, x_{t-1}, x_t) \\ &\equiv f_t(x_t \mid \boldsymbol{x}_{t-1}) \end{aligned}$$

- Independent Observations: $Y_1, Y_2, \ldots, Y_T$.

$$p(Y_t = y_t \mid X_t = x_t) = g_t(x_t, y_t) \equiv g_t(y_t \mid x_t)$$

State-Space Model

- The state-space model is a full probability model with the joint density

$$p(\boldsymbol{x}_T, \boldsymbol{y}_T) = f_0(x_0) \prod_{t=1}^{T} f_t(x_t \mid \boldsymbol{x}_{t-1}) g_t(y_t \mid x_t)$$

- Likelihood estimation problem:

$$\text{Compute } p(\boldsymbol{y}_T; \theta)$$

- Most likely path (MLP) problem:

$$\arg\max_{\boldsymbol{x}_T} \; p(\boldsymbol{x}_T \mid \boldsymbol{y}_T)$$

- Sampling problem:

$$\text{draw } \boldsymbol{x}_T \sim p(\boldsymbol{x}_T \mid \boldsymbol{y}_T)$$

Sampling from a State-Space Model

- Sequential Monte Carlo (SMC) is a set of methods of sampling from the state-space models.
- The key step is based on the following recursive importance sampling step:
  - if $\{(\boldsymbol{x}_t^{(i)}, w_t^{(i)})\}_{i=1}^N$ is a properly weighted sample for

$$p(\boldsymbol{X}_t \mid \boldsymbol{y}_t) \propto f_0(X_0) \prod_{s=1}^{t} f_s(X_s \mid \boldsymbol{X}_{s-1}) g_s(y_s \mid X_s)$$

  - draw $x_{t+1}^{(i)}$ from the proposal distribution $q_{t+1}(X_{t+1} \mid \boldsymbol{x}_t)$
  - let

$$x_{t+1}^{(i)} = (\boldsymbol{x}_t^{(i)}, x_{t+1}^{(i)})$$

$$w_{t+1}^{(i)} \propto w_t^{(i)} \frac{f_{t+1}(x_{t+1}^{(i)} \mid \boldsymbol{x}_t^{(i)}) g_{t+1}(y_{t+1} \mid x_t^{(i)})}{q_{t+1}(x_{t+1}^{(i)} \mid \boldsymbol{x}_t^{(i)})}$$

  - then $\{(\boldsymbol{x}_{t+1}^{(i)}, w_{t+1}^{(i)})\}_{i=1}^N$ is a properly weighted sample for

$$p(\boldsymbol{X}_{t+1} \mid \boldsymbol{y}_{t+1}) \propto f_0(X_0) \prod_{s=1}^{t+1} f_s(X_s \mid \boldsymbol{X}_{s-1}) g_s(y_s \mid X_s)$$

Sequential Importance Sampling (SIS)

1. Draw $x_0^{(i)}$ i.i.d. from $q_0(X_0)$.
2. $w_0^{(i)} \leftarrow f_0(x_0^{(i)})/q_0(x_0^{(i)})$.
3. For $t = 1, \ldots, T$,
   - 3.1 Draw $x_t^{(i)}$ from $q_t(X_t \mid \boldsymbol{x}_{t-1}^{(i)})$.
   - 3.2 Update weight

$$w_t^{(i)} \leftarrow w_{t-1}^{(i)} \frac{f_t(x_t^{(i)} \mid \boldsymbol{x}_{t-1}^{(i)}) g_t(y_t \mid x_t^{(i)})}{q_t(x_t^{(i)} \mid \boldsymbol{x}_{t-1}^{(i)})}.$$

4. Return $\{(\boldsymbol{x}_T^{(i)}, w_T^{(i)})\}_{i=1}^N$.

Sequential Importance Sampling (SIS)

The choice of q_t is arbitrary.

- bootstrap particle filter:

$$q_t(X_t \mid \boldsymbol{X}_{t-1}) = f_t(X_t \mid \boldsymbol{X}_{t-1})$$

- independent particle filter:

$$q_t(X_t \mid \boldsymbol{X}_{t-1}) \propto g_t(y_t \mid X_t)$$

- conditional optimal:

$$q_t(X_t \mid \boldsymbol{X}_{t-1}) \propto f_t(X_t \mid \boldsymbol{X}_{t-1}) g_t(y_t \mid X_t)$$

Major drawback: weight collapse.

- A small number of samples posses the majority of the weights.
- Small ESS. Large variance.
- Solution: resampling.

Resampling

1. At time t, assign a priority score $\beta_t^{(i)}$ to the sample $\boldsymbol{x}_t^{(i)}$.
2. Draw $j_1, \ldots, j_N$ i.i.d. from $\{1, 2, \ldots, N\}$ such that
$$P[j_k = i] = \beta_t^{(i)}.$$
3. relabel the samples
$$\boldsymbol{x}_t^{(i)} \leftarrow \boldsymbol{x}_t^{(j_i)}.$$
4. update weights
$$w_t^{(i)} \leftarrow \frac{w_t^{(j_i)}}{\beta_t^{(j_i)}}.$$

Resampling

- The choice of priority score β_t is arbitrary.
  - Conventional choice: $\beta_t \propto w_t$
  - Auxiliary particle filter: $\beta_t \propto p(y_{t+1} \mid \boldsymbol{X}_t)$
  - Delayed particle filter: $\beta_t \propto p(y_{t+\Delta} \mid \boldsymbol{X}_t)$

- Resampling algorithms:
  - Multinomial, residual, stratified, etc..

- Resampling schedule:
  - Fixed schedule: Do resampling at $t = \delta, 2\delta, 3\delta, \ldots$.
  - Adaptive schedule: Do resampling when ESS $< 0.3N$.

$$\text{Effective Sample Size(ESS)} = \frac{\left(\sum_{i=1}^{N} w^{(i)}\right)^2}{\sum_{i=1}^{N} \left(w^{(i)}\right)^2}$$

- Drawback: degeneracy.

Sequential Importance Sampling with Resampling

1. Draw $x_0^{(i)}$ i.i.d. from $q_0(X_0)$.
2. $w_0^{(i)} \leftarrow f_0(x_0^{(i)})/q_0(x_0^{(i)})$.
3. For $t = 1, \ldots, T$,
   3.1 Draw $x_t^{(i)}$ from $q_t(X_t \mid \boldsymbol{x}_{t-1}^{(i)})$.
   3.2 Update weight
   $$w_t^{(i)} \leftarrow w_{t-1}^{(i)} \frac{f_t(x_t^{(i)} \mid \boldsymbol{x}_{t-1}^{(i)}) g_t(y_t \mid x_t^{(i)})}{q_t(x_t^{(i)} \mid \boldsymbol{x}_{t-1}^{(i)})}.$$
   3.3 **Resampling (Optional):**
   - **assign a priority score $\beta_t^{(i)}$ to the sample $\boldsymbol{x}_t^{(i)}$.**
   - **Draw $j_1, \ldots, j_N$ i.i.d. from $\{1, 2, \ldots, N\}$ such that**
     $$P[j_k = i] = \beta_t^{(i)}.$$
   - **relabel the samples**
     $$\boldsymbol{x}_t^{(i)} \leftarrow \boldsymbol{x}_t^{(j_i)}.$$
   - **update weights**
     $$w_t^{(i)} \leftarrow \frac{w_t^{(j_i)}}{\beta_t^{(j_i)}}.$$
4. Return $\{(\boldsymbol{x}_T^{(i)}, w_T^{(i)})\}_{i=1}^N$.

SISR

Sequential Importance Sampling with Resampling (SISR) is a flexible framework where the user decides

- the proposal function q_t,
- the priority score β_t,
- the resampling algorithm,
- the resampling schedule.

These factors are crucial for the performance of the Monte Carlo sample in estimation and optimization.

Strong Constraints

Information at time t:

$$\mathcal{I}_t$$

Cumulative information/constraints up to time t:

$$\mathcal{C}_0 \supset \mathcal{C}_1 \supset \cdots \supset \mathcal{C}_T,$$

such that

$$\mathcal{C}_{t+1} = \mathcal{C}_t \cap \mathcal{I}_t.$$

Strong constraints can affect the target distribution significantly.

$$G(t) = \chi^2 \left(p(\boldsymbol{X}_t \mid \mathcal{C}_{t-1}) || p(\boldsymbol{X}_t \mid \mathcal{C}_t) \right) = \mathrm{Var}_{p(\boldsymbol{X}_t \mid \mathcal{C}_{t-1})} \left[\frac{p(\boldsymbol{X}_t \mid \mathcal{C}_t)}{p(\boldsymbol{X}_t \mid \mathcal{C}_{t-1})} \right]$$

Notation: $t_+ \geqslant t$ is the next time a strong constraint is imposed after time t.

SMC with Strong Constraints

- The perfect intermediate sampling distribution:

$$\bar{p}_t(\boldsymbol{X}_t) := p(\boldsymbol{X}_t \mid \mathcal{C}_T)$$

  - The most efficient choice.
  - Difficult to draw from $p(X_{t+1} \mid \boldsymbol{X}_t, \mathcal{C}_T)$
- The current intermediate sampling distribution of SMC:

$$\tilde{p}_t(\boldsymbol{X}_t) := p(\boldsymbol{X}_t \mid \mathcal{C}_t)$$

  - Easy to sample sequentially.
  - May miss the constraint $\mathcal{I}_T$ in the future.
- We propose the following intermediate sampling distribution:

$$p_t^+(\boldsymbol{X}_t) := p(\boldsymbol{X}_t \mid \mathcal{C}_{t_+})$$

  - Consider potential future strong constraints.
  - Consider only one future constraint.

SMC with Strong Constraints

To incorporate $\mathcal{I}_{t_+}$, one can modify the SMC algorithm by

- Resampling method ✗
- Resampling schedule ✗
- Proposal distribution q_t ✓
- Priority score β_t ✓

SMC with Strong Constraints

Changing proposal distribution q_t:

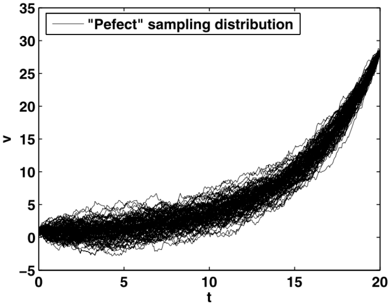
- Use linear interpolation q_t to make the trajectories more likely to satisfy $\mathcal{I}_{t_+}$.
- Easy to implement.
- Properly weighted.
- May break the underlying nature/shape/topology.

Changing priority score β_t:

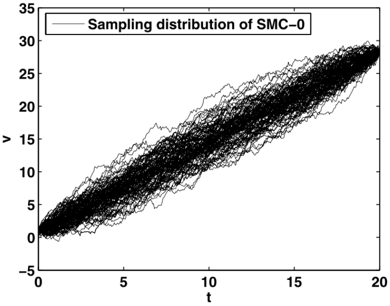
- Assign higher priority score to samples that are more likely to comply with $\mathcal{I}_{t_+}$.
- Properly weighted.
- Keep the underlying nature/shape/topology.
- Need to calculate/approximate/evaluate the optimal priority score.

SMC with Strong Constraints

$$dv_t = 0.2 v_t dt + dw_t$$



(a)



(b)

Ming Lin, Rong Chen, and Per Mykland. “On generating Monte Carlo samples of continuous diffusion bridges.” Journal of the American Statistical Association 105.490 (2010): 820-838.

Optimal Priority Score

We observe that

$$p_t^+(\boldsymbol{X}_t) \propto \tilde{p}_t(\boldsymbol{X}_t) p(\mathcal{C}_{t_+} \mid \boldsymbol{X}_t, \mathcal{C}_t)$$

If $\{(\boldsymbol{x}_t^{(i)}, w_t^{(i)})\}_{i=1}^N$ is properly weighted w.r.t. $\tilde{p}_t(\boldsymbol{X}_t)$,
then $\{(\boldsymbol{x}_t^{(i)}, w_t^{(i)} p(\mathcal{C}_{t_+} \mid \boldsymbol{x}_t^{(i)}, \mathcal{C}_t))\}_{i=1}^N$ is properly weighted w.r.t. $p_t^+(\boldsymbol{X}_t)$.

- We run SIS using $\tilde{p}_t(\boldsymbol{X}_t)$ as in the conventional SMC.
- The resampling step should be done w.r.t. $p_t^+(\boldsymbol{X}_t)$, that is, in the resampling step, we use **optimal priority score**

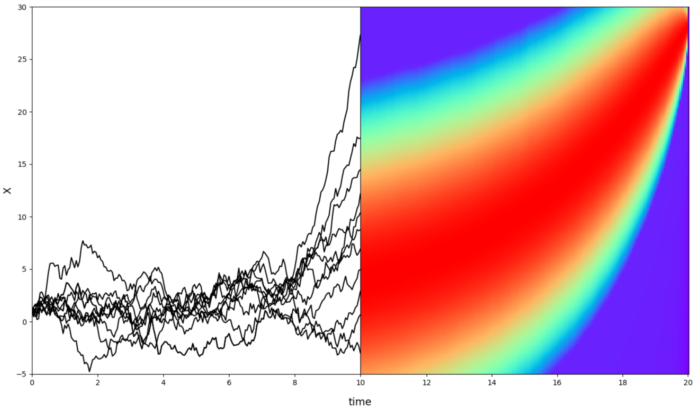
$$\beta_t^{(i)} \propto w_t^{(i)} p(\mathcal{C}_{t_+} \mid \boldsymbol{x}_t^{(i)}, \mathcal{C}_t)$$

- Sequential Importance Sampling (under $\tilde{p}_t$) with Resampling (under p_t^+)

Optimal Priority Score

SMC with constraints (SMCc):

$$\text{Use priority score } \beta_t \propto w_t p(\mathcal{C}_{t_+} \mid \boldsymbol{X}_t, \mathcal{C}_t)$$



Optimal Priority Score

$$p(\mathcal{C}_{t_+} \mid \boldsymbol{X}_t, \mathcal{C}_t) \propto \int \cdots \int \prod_{s=t+1}^{t_+} p(\mathcal{I}_s \mid X_s) p(X_s \mid \boldsymbol{X}_{s-1}) d\boldsymbol{X}_{t+1:t_+}$$

Two pilot methods to estimate $p(\mathcal{C}_{t_+} \mid \boldsymbol{X}_t, \mathcal{C}_t)$:

1. Parametric Approximation
   Use tractable parametric functions to approximate the integrand (e.g. multivariate normal)
2. Forward Pilot.
3. Backward Pilot.

Forward Pilot

Suppose there is a low-dimensional summary statistics $S(\boldsymbol{X}_t)$ such that

$$p(X_{t+1:t+d}, \mathcal{C}_{t+d} \mid \boldsymbol{X}_t, \mathcal{C}_t) = p(X_{t+1:t+d}, \mathcal{C}_{t+d} \mid S(\boldsymbol{X}_t), \mathcal{C}_t)$$

Also, assume there is a function ϕ such that

$$S(\boldsymbol{X}_{t+1}) = \phi(S(\boldsymbol{X}_t), X_{t+1})$$

Then $p(\mathcal{C}_{t_+} \mid \boldsymbol{X}_t, \mathcal{C}_t) = p(\mathcal{C}_{t_+} \mid S(\boldsymbol{X}_t), \mathcal{C}_t)$.
This can be estimated by a kernel estimation based on a forward pilot sample (without resampling).

Forward Pilot

The forward pilot algorithm (part I): (for $t_1 < t \leq t_2$)

- Initialization: For $j = 1, \ldots, m$, draw samples $\widetilde{S}_{t_1}^{(j)}$ from a proposal distribution $\varphi(S)$ that covers the support of $S(x_{0:t_1})$.
- For $t = t_1 + 1, \ldots, t_2$, draw pilot samples forwardly as follows.
  - Generate samples $\widetilde{x}_t^{(j)}$ from a proposal distribution $\varphi(\widetilde{x}_t \mid \widetilde{S}_{t-1}^{(j)})$, and calculate $\widetilde{S}_t^{(j)} = \phi(\widetilde{S}_{t-1}^{(j)}, \widetilde{x}_t^{(j)})$ for $j = 1, \ldots, m$.
  - Calculate the incremental weights

$$\widetilde{u}_t^{(j)} = \frac{p(\widetilde{x}_t^{(j)}, C_t \mid S(\widetilde{x}_{0:t-1}^{(j)}) = \widetilde{S}_{t-1}^{(j)}, C_{t-1})}{\varphi(\widetilde{x}_t^{(j)} \mid \widetilde{S}_{t-1}^{(j)})}, \quad j = 1, \ldots, m.$$

Forward Pilot

We observe that

$$\mathbb{E}\left[\prod_{s=t+1}^{t_2} \tilde{u}_t^{(j)} \mid S_t^{(j)} = S\right] = p(\mathcal{C}_{t_2} \mid S(\boldsymbol{x}_t) = S, \mathcal{C}_t)$$

A very rough kernel estimation would work.
The forward pilot algorithm (part II):

- For $t = t_2 - 1, t_2 - 2, \ldots, t_1 + 1$:
  - Compute $U_t^{(j)} = \prod_{s=t+1}^{t_2} \widetilde{u}_s^{(j)}$ for $j = 1, \ldots, m$.
  - Let $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_D$ be a partition of the support of $S(x_{0:t})$. Estimate $p(C_{t_+} \mid x_{0:t}, C_t) = p(C_{t_+}|S(x_{0:t}), C_t)$ by

$$f_t(S(x_{0:t})) = \sum_{d=1}^{D} \gamma_{t,d}\mathbb{I}\big(S(x_{0:t}) \in \mathcal{S}_d\big) \tag{3.3}$$

with $\gamma_{t,d} = \sum_{j=1}^{m} U_t^{(j)}\mathbb{I}(\widetilde{S}_t^{(j)} \in \mathcal{S}_d)/\sum_{j=1}^{m}\mathbb{I}(\widetilde{S}_t^{(j)} \in \mathcal{S}_d)$, where $\mathbb{I}(\cdot)$ is the indicator function.

Backward Pilot

If the system is Markovian:

$$p(X_t, \mathcal{I}_t \mid \boldsymbol{X}_{t-1}, \mathcal{C}_{t-1}) = p(X_t, \mathcal{I}_t \mid X_{t-1}, \mathcal{C}_{t-1})$$

then the optimal priority score is

$$p(\mathcal{C}_{t_+} \mid X_t, \mathcal{C}_t) \propto \int \cdots \int \prod_{s=t+1}^{t_+} p(\mathcal{I}_s \mid X_s) p(X_s \mid X_{s-1}) d\boldsymbol{X}_{t+1:t_+}$$

We may draw samples in a backward fashion from

$$p(\boldsymbol{X}_{t:t_+}) \propto \prod_{s=t+1}^{t_+} p(\mathcal{I}_s \mid X_s) p(X_s \mid X_{s-1})$$

And $p(\mathcal{C}_{t_+} \mid X_t, \mathcal{C}_t)$ as a function of X_t is proportional to the marginal density of X_t.

Backward Pilot

The backward pilot algorithm contains (1) backward SIS (2) kernel estimation.

- Initialization: For $j = 1, \ldots, m$, draw samples $\tilde{x}_{t_2}^{(j)}$ from a proposal distribution $r(x_{t_2})$ approximately proportional to $p(I_{t_2} \,|\, x_{t_2})$ and set $\tilde{w}_{t_2}^{(j)} = 1/r(\tilde{x}_{t_2}^{(j)})$.
- For $t = t_2 - 1, \ldots, t_1 + 1$, draw pilot samples backward as follows.
  - Generate samples $\tilde{x}_t^{(j)}$, $j = 1, \ldots, m$, from a proposal distribution $r(\tilde{x}_t \,|\, \tilde{x}_{t+1}^{(j)})$.
  - Update weights by

$$\tilde{w}_t^{(j)} = \tilde{w}_{t+1}^{(j)} \frac{p(\tilde{x}_{t+1}^{(j)}, I_{t+1} \,|\, \tilde{x}_t^{(j)})}{r(\tilde{x}_t^{(j)} \,|\, \tilde{x}_{t+1}^{(j)})}, \quad j = 1, \ldots, m.$$

  - Let $\mathcal{X}_1 \cup \cdots \cup \mathcal{X}_D$ be a partition of the support of x_t. Estimate $p(C_{t_+} \,|\, x_{0:t}, C_t) = p(\mathcal{I}_{t+1:t_+} \,|\, x_t)$ by

$$f_t(x_t) = \sum_{d=1}^{D} \eta_{t,d} \mathbb{I}\big(x_t \in \mathcal{X}_d\big), \tag{4.1}$$

where $\eta_{t,d} = (1/m|\mathcal{X}_d|) \sum_{j=1}^m \tilde{w}_t^{(j)} \mathbb{I}(\tilde{x}_t^{(j)} \in \mathcal{X}_d)$, and $|\mathcal{X}_d|$ denotes the volume of the subset $\mathcal{X}_d$.

Long-Run Marginal Expected Shortfall (LRMES)

- Let $x_{f,t}$ and $x_{m,t}$ be the daily log-price of the firm and the market, respectively.
- The long-run marginal expected shortfall (LRMES) is defined as

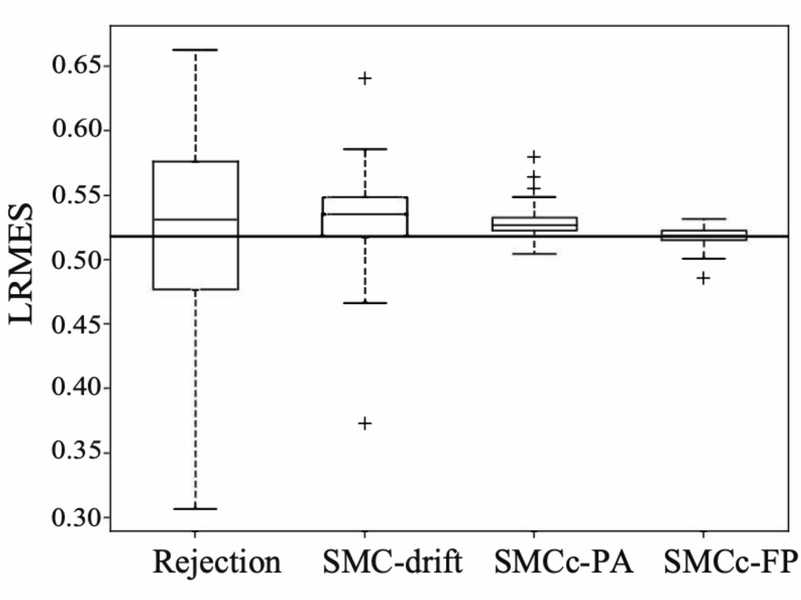
$$\text{LRMES} = \mathbb{E}[1 - e^{x_{f,T} - x_{f,0}} \mid e^{x_{m,T} - x_{m,0}} < 60\%]$$

- The dynamics of $(x_{f,t}, x_{m,t})$ is assumed to follow the Glosten-Jagannathan-Runkle generalized autoregressive conditional heteroskedasticity model (GJR-GARCH):

$$\begin{bmatrix} x_{m,t} \\ x_{f,t} \end{bmatrix} = \begin{bmatrix} x_{m,t-1} \\ x_{f,t-1} \end{bmatrix} + \begin{bmatrix} \sigma^2_{m,t} & \rho_t \sigma_{m,t} \sigma_{f,t} \\ \rho_t \sigma_{m,t} \sigma_{f,t} & \sigma^2_{f,t} \end{bmatrix}^{1/2} \begin{bmatrix} \epsilon_{m,t} \\ \epsilon_{f,t} \end{bmatrix}$$

with $\sigma^2_{m,t}$ and $\sigma^2_{f,t}$ evolves over time as well.

Long-Run Marginal Expected Shortfall (LRMES)



Long-Run Marginal Expected Shortfall (LRMES)

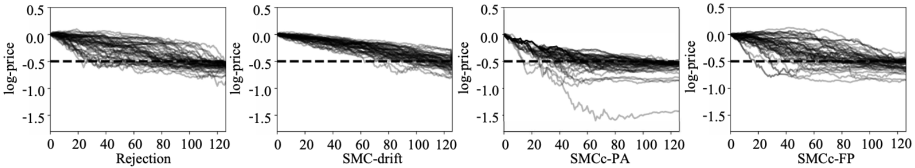


Figure 2. Sample paths of $X_{m,0:T}$ generated by different methods before weight adjustment. The horizontal line denotes a 40% price decrease.

Robot Control

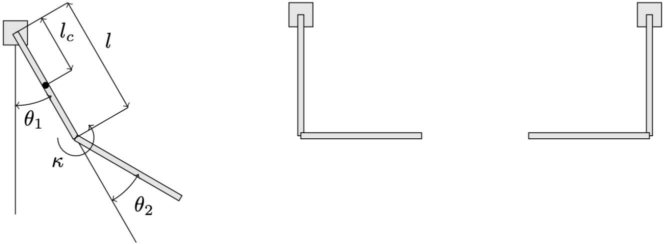


Figure 5. Acrobot with two arms (left panel), starting position $\boldsymbol{\theta}_0$ at $(0, \pi/2, 0, 0)$ (middle panel), and target position $\boldsymbol{\theta}_*$ at $(0, -\pi/2, 0, 0)$ (right panel) .

- A two-arm robot system with a controllable torque at the joint.
- The status of the system is described by $\boldsymbol{\theta} = (\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2)'$.
- The physical law governs the dynamic.
- $\kappa \sim \text{Unif}[-5, 5]$ generates the probability space.
- We generate $\theta_{0:T} \propto p(\boldsymbol{\theta}_{0:T} \mid \mathcal{C}) e^{-\alpha\tau}$ with $\tau = \min\{t : \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_*\|_\infty < 0.01\}$

Robot Control

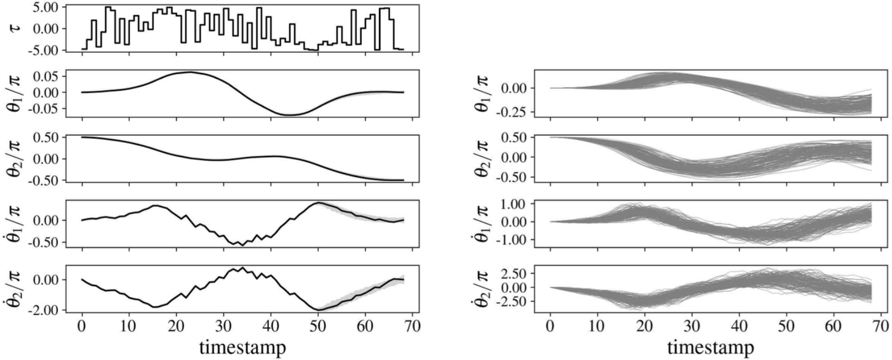
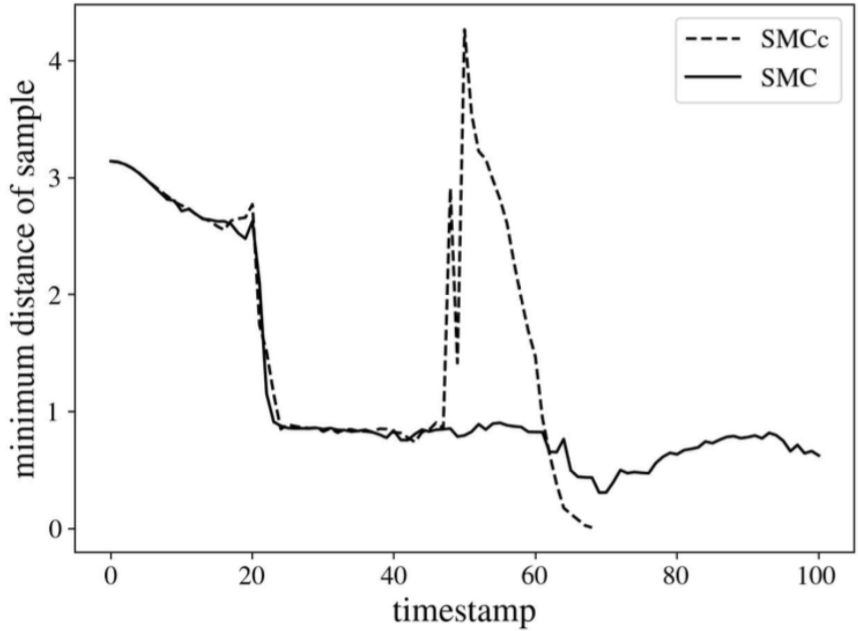


Figure 6. Left panel: Sample paths generated using the SMCc-BP method (in gray), with the "optimal" path (in black) that reaches the target state $\boldsymbol{\theta}_*$ at $t = 68$. The control sequence for the "optimal" path is shown in the top panel. Right panel: Sample paths generated using the random search method for $t = 0, 1, \ldots, 68$.

Robot Control



Summary

Joint work with Ming Lin (Xiamen University) and Rong Chen (Rutgers University).

Paper:
Resampling Strategy in Sequential Monte Carlo for Constrained Sampling Problems.
Statistica Sinica. **34** (2024), 1-18.

A related work:
State Space Emulation and Annealed Sequential Monte Carlo for High Dimensional Optimization. *Statistica Sinica.* To appear (2025).