Sequential Monte Carlo methods

A not-so-theoretical introduction

Flávio Eler De Melo

School of Engineering and Physical Sciences
Heriot-Watt University

June 27th, 2017

Outline

Monte Carlo

- Experiment-based methods for solving physical and mathematical problems
- A sufficient number of experiments is realized to enable computing a physical quantity
- Characterizing real phenomena is hard (often impossible in the analytical sense)
- Amount and nature of uncertainty is generally unknown
- Convenient when computational power is available

An illustration: estimate π

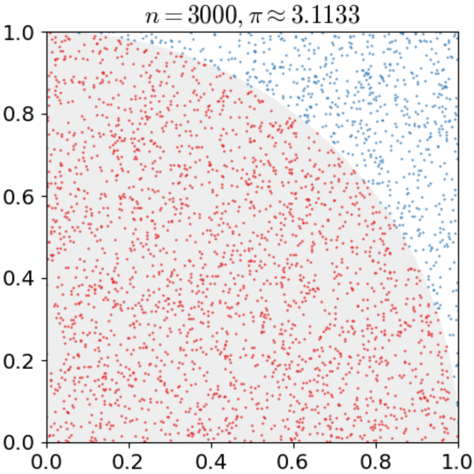
Experiment

- Take N i.i.d. samples $\{X^{(i)}\}_{i\in[1..N]}$ from the uniform distribution on a square with side ℓ
- Count the samples that fall inside a circle inscribed in the square (N_{circle})
- Estimate π as

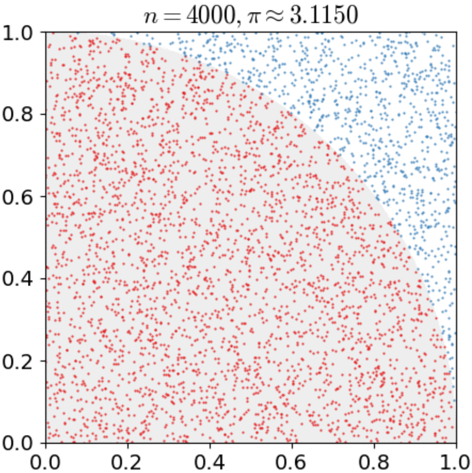
$$
\begin{aligned}
\Pr\{X \text{ in the circle}\} &= \frac{A_{\text{circle}}}{A_{\text{square}}} \approx \frac{N_{\text{circle}}}{N} \\
&= \frac{\pi\ell^2/4}{\ell^2} = \frac{\pi}{4} \approx \frac{N_{\text{circle}}}{N} \\
\therefore \pi &\approx \hat{\pi} = \frac{4N_{\text{circle}}}{N}
\end{aligned}
$$

- With $N = 100,000,000$ samples, $|\pi - \hat{\pi}| \sim 10^{-5}$. What is going on?

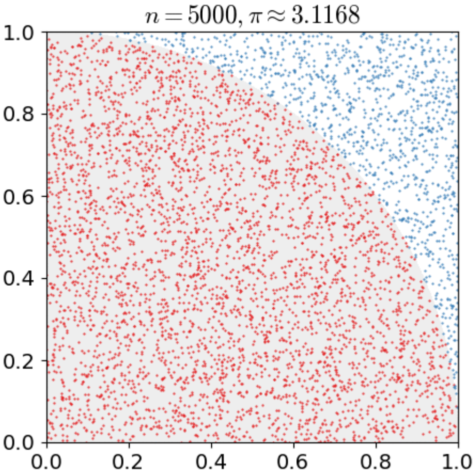
An illustration: estimate π



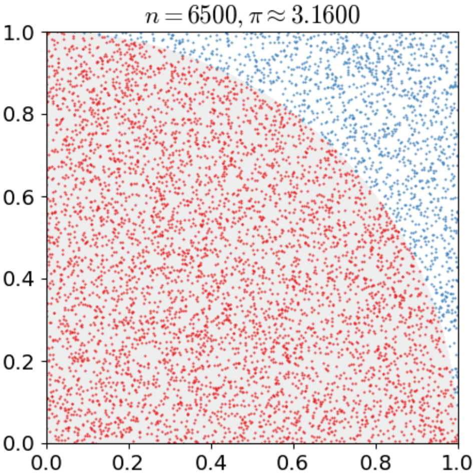
An illustration: estimate π



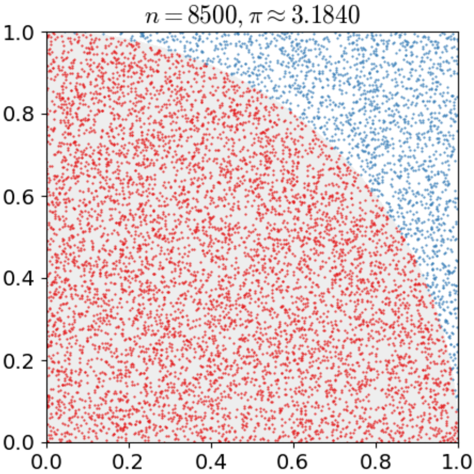
An illustration: estimate π



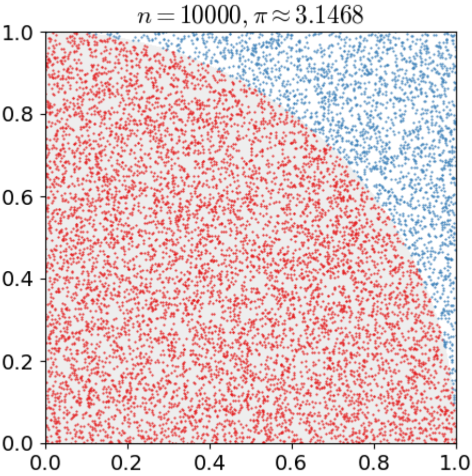
An illustration: estimate π



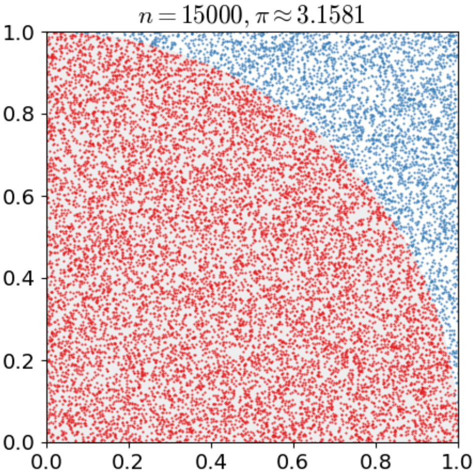
An illustration: estimate π



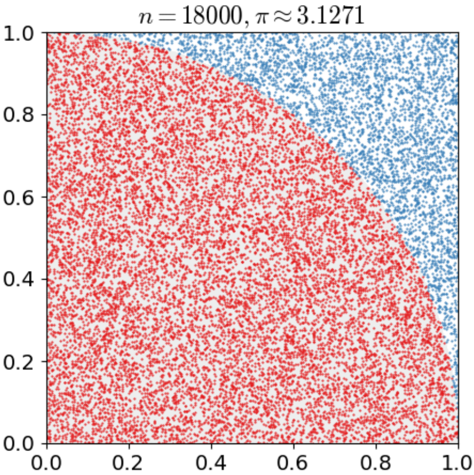
An illustration: estimate π



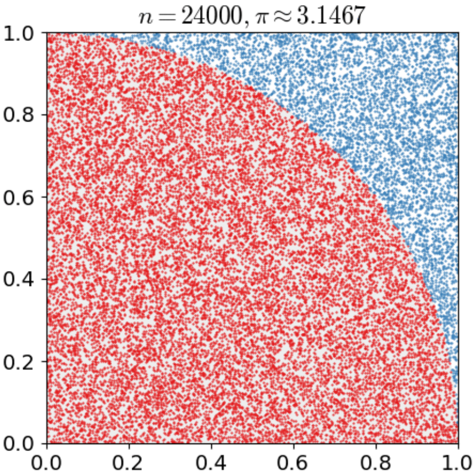
An illustration: estimate π



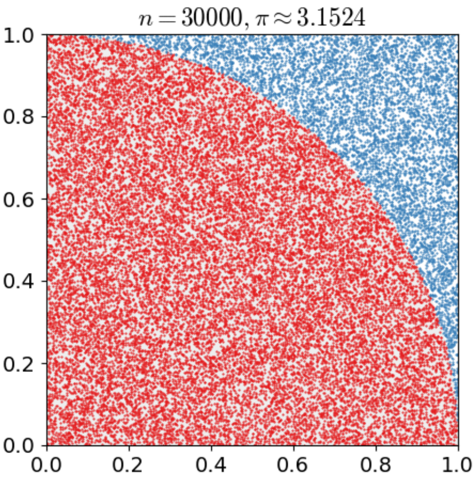
An illustration: estimate π



An illustration: estimate π



An illustration: estimate π



Monte Carlo

- Law of large numbers: an empirical average tends to the expected value as the number of experiments increases
- Regions:

$$\Omega_{\text{square}} := \{x \in \mathbb{R}^2 : x \text{ is in the square}\},$$
$$\Omega_{\text{circle}} := \{x \in \mathbb{R}^2 : x \text{ is in the circle}\},$$

- $X^{(i)} \sim \mathcal{U}(x; \partial A_{\text{square}})$, for $i = 1, \ldots, N$,

$$\mathcal{U}(x; \partial A_{\text{square}}) = \begin{cases} 1/A_{\text{square}}, & x \in \Omega_{\text{square}}, \\ 0, & \text{otherwise.} \end{cases}$$

- Probability:

$$\begin{aligned} \Pr\{X \in \Omega_{\text{circle}}\} &\triangleq \int_{\Omega_{\text{circle}}} \mathcal{U}(x; \partial A_{\text{square}}) dx \\ &= A_{\text{square}}^{-1} \int_{\Omega_{\text{circle}}} dx = A_{\text{circle}}/A_{\text{square}}. \end{aligned}$$

Monte Carlo

- Monte Carlo enables estimates of $\mathbb{E}[\varphi] = \int_{\mathcal{X}} \varphi(x) p_\pi(x) dx$, for locally integrable functions $\varphi(\cdot)$ of $x \in \mathcal{X}$, by computing

$$\hat{\varphi} = \frac{1}{N} \sum_{i=1}^{N} \varphi(x^{(i)}), \quad x^{(i)} \sim p_\pi(x).$$

- π estimate: $\varphi(x) = \mathbb{1}_{\Omega_{\text{circle}}}(x)$, where $\mathbb{1}_B(x) = 1$ if $x \in B$ and zero otherwise.
- In our example we sample from a uniform density: prior knowledge about the phenomenon is required
- Often prior knowledge is available but sampling is difficult: we know how to sample from a limited number of probability densities.
- How to sample? We know how to generate samples from $\mathcal{U}([0,1])$, for instance, $Z^{(i)} = \mod (aZ^{(i-1)} + c, m)$ and $Z^{(i)}/m \sim \mathcal{U}([0,1])$.

Importance sampling

- How to proceed if we do not know how sample from $p_\pi(x)$ (target measure) but we know how to evaluate it?
- Suppose a distribution $q(x)$ (proposal) from which it is easy to sample, and is somewhat "close" to $p_\pi(x)$
- By doing

$$\begin{aligned}\mathbb{E}[\varphi] &= \int_{\mathcal{X}} \varphi(x) p_\pi(x) dx = \int_{\mathcal{X}} \overbrace{\frac{p_\pi(x)}{q(x)}}^{\breve{w}(x)} \varphi(x) q(x) dx \\ &= \int_{\mathcal{X}} \breve{w}(x) \varphi(x) q(x) dx,\end{aligned}$$

- We take samples $x^{(i)} \sim q(x)$, and compute the estimate as

$$\hat{\varphi} = \sum_{i=1}^{N} w^{(i)} \varphi(x^{(i)}), \quad w^{(i)} = \breve{w}(x^{(i)})/N.$$

Sequential Monte Carlo

- What if:
  - x_t now varies with time, i.e., the sequence $\{x_t\}_{t\geq 0}$ is a stochastic process
  - Evidence about the process is given by an observation process $\{y_k\}_{k\in\mathbb{N}}$, realized at time steps $t = t_k$.
- Can we estimate $\mathbb{E}[\varphi_t|y_1,\dots,y_k] = \int_{\mathcal{X}} \varphi(x_t)p(x_t|y_1,\dots,y_k)dx_t$?
- Solution: sequential Monte Carlo methods.

Sequential importance sampling (SIS)

- For simplicity we write sequences of states and observations as $x_{0:k} = (x_0, x_1, \dots, x_k)$ and $y_{1:k} = (y_1, y_2, \dots, y_k)$.
- Sequential importance sampling performs inference as

$$\mathbb{E}[\varphi_k | y_{1:k}] = \int_{\mathcal{X}} \breve{w}(x_{0:k}|y_{1:k})\varphi(x_k)q(x_{0:k}|y_{1:k})dx_k,$$
$$\breve{w}(x_{0:k}|y_{1:k}) \triangleq \frac{p_\pi(x_{0:k}|y_{1:k})}{q(x_{0:k}|y_{1:k})}.$$

- At time step $k-1$, we possess a set of weights and samples (particles) $\{w_{k-1}^{(i)}, x_{k-1}^{(i)}\}$
- In the standard SIS setting $x_{k-1}^{(i)}$ is a path sample, i.e., $x_{k-1}^{(i)} \equiv x_{0:k-1}^{(i)} = x_{k-1}^{(i)}, x_{k-2}^{(i)}, \dots x_0^{(i)}$
- The weights are given by $w_{k-1}^{(i)} \propto \breve{w}(x_{0:k-1}^{(i)}|y_{1:k-1})$

Sequential importance sampling (SIS)

- When a new observation y_k becomes available, new samples extend the path of previous samples, i.e.,

$$x_k^{(i)} \sim q(x_{0:k}|y_{1:k}) = q(x_k|x_{0:k-1}, y_k)q(x_{0:k-1}|y_{1:k-1}),$$
$$x_k^{(i)} \sim q(x_k|x_{k-1}^{(i)}, y_k) \equiv q(x_k|x_{0:k-1}^{(i)}, y_k),$$

- The new weights are updated as

$$\breve{w}(x_{0:k}|y_{1:k}) := \frac{p_\pi(x_{0:k}|y_{1:k})}{q(x_{0:k}|y_{1:k})} = \frac{\frac{p_\pi(y_k|x_k)p(x_k|x_{k-1})}{p(y_k|y_{1:k-1})}}{q(x_k|x_{k-1}, y_k)} \frac{p_\pi(x_{0:k-1}, y_{1:k-1})}{q(x_{0:k-1}|y_{1:k-1})}$$
$$= \frac{1}{p(y_k|y_{1:k-1})} \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q(x_k|x_{k-1}, y_k)} \breve{w}(x_{0:k-1}|y_{1:k-1}),$$

- Estimates are given as

$$\hat{\varphi} = \frac{\frac{1}{N}\sum_{i=1}^N \breve{w}_k^{(i)} \varphi(x_k^{(i)})}{\frac{1}{N}\sum_{i=1}^N \breve{w}_k^{(i)}}, \quad \breve{w}_k^{(i)} = \breve{w}(x_{0:k}^{(i)}|y_{1:k}).$$

Sequential importance sampling (SIS)

- Usual choices of one-step proposals:
  - Bootstrap filter: $q(x_k|x_{k-1}, y_k) = p(x_k|x_{k-1})$, resulting in

    $$\breve{w}(x_{0:k}|y_{1:k}) \propto p(y_k|x_k)\breve{w}(x_{0:k-1}|y_{1:k-1})$$

    .
  - Optimal proposal: $q(x_k|x_{k-1}, y_k) = \frac{p(y_k|x_k)p(x_k|x_{\mathbf{k-1}})}{p(y_k|x_{\mathbf{k-1}})}$, resulting in

    $$\breve{w}(x_{0:k}|y_{1:k}) \propto p(y_k|x_{k-1})\breve{w}(x_{0:k-1}|y_{1:k-1})$$

Problems

- Weight degeneracy: if proposed particles are too far from the region of high probability under the target distribution, only a few particles will have significant weight, which causes the other weights to become irrelevant for the estimate.
- Particle degeneracy: a direct consequence of the curse of dimensionality. Recall that the particles extend stochastic paths, which in turn occupy a space with increasing dimension as $x_{0:k}^{(i)} \in \mathcal{X}^{k+1}$. As the number of dimensions increases, a finite number of realizations can only populate the space to an increasingly sparse extent.

Sequential Monte Carlo samplers

- In Markov Chain Monte Carlo literature, estimates can be generated by simulating an event according to a transition Markov kernel (reversible) that corresponds to an invariant (stationary) distribution $p_\pi(dx)$.
- Convergence to the invariant distribution is only guaranteed by using an accept-reject step.
- Sample a candidate $x_k^{\star(i)} \sim q(x_k | x_{k-1}^{(i)})$
- Compute acceptance probability
$\alpha^{(i)}(x_k^{\star(i)} | x_{k-1}^{(i)}) = \min\left(\frac{p_\pi(x_k^{\star(i)}) q(x_{k-1}^{(i)} | x_k^{\star(i)})}{p_\pi(x_{k-1}^{(i)}) q(x_k^{\star(i)} | x_{k-1}^{(i)})}, 1 \right)$
- Sample a test variable $u^{(i)} \in \mathcal{U}([0,1]$, if $u^{(i)} \leq \alpha^{(i)}$ then accept the candidate $x_k^{(i)} \leftarrow x_k^{\star(i)}$, else reject the move $x_k^{(i)} \leftarrow x_{k-1}^{(i)}$.

Sequential Monte Carlo samplers

- MCMC acknowledges and corrects for the fact that a single-step proposal can lead the chain to the wrong direction, and so convergence is guaranteed by accept-reject step.
- When doing particle filtering (SIS), once a candidate is sampled the move is made, such that convergence to the target distribution is not enforced.
- Particle filtering degenerates when unlikely moves are made and the weights lose relevance.
- Sequential Monte Carlo samplers introduces a weight compensation to account for possibly bad moves. This is done via introduction of a backward Kernel.

Sequential Monte Carlo samplers

- Sequential Monte Carlo samplers provide estimates for

$$\mathbb{E}[\varphi_k|y_{1:k}] = \int_{\mathcal{X}} \breve{w}(x_{0:k}|y_{1:k})\varphi(x_k)q(x_{0:k}|y_{1:k})dx_k,$$

$$\breve{w}(x_{0:k}|y_{1:k}) \triangleq \frac{p_\pi(x_{0:k}|y_{1:k})}{q(x_{0:k}|y_{1:k})}.$$

- And introduces the backward kernel $L^{(k)}(x_{k-1}|x_k)$ such that

$$p_\pi(x_{0:k}|y_{1:k}) = p_\pi(x_k|y_{1:k})L^{(k)}(x_{k-1}|x_k)L^{(k-1)}(x_{k-2}|x_{k-1})\ldots L^{(1)}(x_0|x_1),$$

$$p_\pi(x_k|y_{1:k}) = \int p_\pi(x_{0:k}|y_{1:k})dx_{0:k-1}.$$

Sequential Monte Carlo samplers

- The new weights are updated as

$$\begin{aligned}
\breve{w}(x_{0:k}|y_{1:k}) &:= \frac{p_\pi(x_{0:k}|y_{1:k})}{q(x_{0:k}|y_{1:k})} = \frac{p_\pi(x_k|y_{1:k})L^{(k)}(x_{k-1}|x_k)L^{(k-1)}(x_{k-2}|x_{k-1})\dots}{q(x_k|x_{k-1},y_k)q(x_{0:k-1}|y_{1:k-1})} \\
&= \frac{p_\pi(x_k|y_{1:k})L^{(k)}(x_{k-1}|x_k)}{q(x_k|x_{k-1},y_k)q(x_{0:k-1}|y_{1:k-1})} \frac{p_\pi(x_{k-1}|y_{1:k-1})}{p_\pi(x_{k-1}|y_{1:k-1})} L^{(k-1)}(x_{k-2}|x_{k-1})\dots \\
&= \frac{p_\pi(x_k|y_{1:k})L^{(k)}(x_{k-1}|x_k)}{p_\pi(x_{k-1}|y_{1:k-1})q(x_k|x_{k-1},y_k)} \frac{p_\pi(x_{0:k-1}|y_{1:k-1})}{q(x_{0:k-1}|y_{1:k-1})} \\
&= \frac{p_\pi(x_k|y_{1:k})L^{(k)}(x_{k-1}|x_k)}{p_\pi(x_{k-1}|y_{1:k-1})q(x_k|x_{k-1},y_k)} \breve{w}(x_{0:k-1}|y_{1:k-1}) \\
&\equiv \underbrace{\alpha_L(x_k|x_{k-1},y_k)}_{\text{analogue of } \alpha} \breve{w}(x_{0:k-1}|y_{1:k-1}).
\end{aligned}$$

Optimal Transport

Monge-Kantorovich problem:

- Two densities $p_0(x)$ and $p_\Lambda(x)$, with total mass $\int_{\mathcal{X}} p_0(x)dx = \int_{\mathcal{X}} p_\Lambda(x)dx = 1$
- Find a smooth one-to-one map $M : \mathcal{X} \to \mathcal{X}$, $M : p_0 \mapsto p_\Lambda$, where $\int_{x \in A} p_0(x)dx = \int_{M(x) \in A} p_\Lambda(M(x))dM(x)$, that achieves

$$d(p_0, p_\Lambda)^r = \inf_M \int \|M(x) - x\|^r p_0(x)dx, \; r \geq 0.$$

- The map means that $\det(\nabla M) \cdot p_\Lambda(M(x)) = p_0(x)$
- When $r = 2$, the problem is a continuum mechanics classical problem:

$$\partial_\lambda p = -\nabla \cdot (p\mu), \; \lambda \in [0, \Lambda], \; p(0, \cdot) = p_0, \; p(\Lambda, \cdot) = p_\Lambda.$$

Optimal Transport

- $\det(\nabla M) \cdot p_\Lambda(M(x)) = p_0(x)$ is highly nonlinear, and becomes a second-order elliptic equation for "potential maps" as $M = \nabla \Psi$
- Solving $\partial_\lambda p = -\nabla \cdot (p\mu)$ by a Monte Carlo method only requires propagating samples according to $\dot{x} = \mu(\lambda)$
- Reich, 2011: Parametrize p as a sequence of $N = \Lambda / \Delta\lambda$ intermediate densities, $(p_j)_{j \in [0..N]}$, which arise by applying the likelihood progressively

$$\ell_y(x) = \frac{1}{\sqrt{2\pi \det R}} e^{-\frac{1}{2}(y-Hx)^T R^{-1}(y-Hx)} \propto e^{-L_y(x)},$$

$$\ell_y^N(x) \propto e^{-\frac{L_y(x)}{N}} = e^{-\frac{L_y(x)}{\Lambda/\Delta\lambda}} \implies \ell_y(x) \propto \prod_{j=1}^{N} \ell_y^N(x).$$

Optimal Transport

$$p_{j+1}(x) = \frac{\ell_y^N(x)p_j(x)}{\int \ell_y^N(x)p_j(x)dx} = \frac{\left(1 - \Delta\lambda \frac{L_y(x)}{\Lambda}\right) p_j(x)}{\int \left(1 - \Delta\lambda \frac{L_y(x)}{\Lambda}\right) p_j(x)dx} + \mathcal{O}(\Delta\lambda^2),$$

$$p_{j+1}(x) = \frac{p_j(x) - \Delta\lambda \frac{L_y(x)}{\Lambda} p_j(x)}{1 - \frac{\Delta\lambda}{\Lambda}\mathbb{E}\left[L_y(x)\right]} + \mathcal{O}(\Delta\lambda^2),$$

Optimal Transport

$$p_{j+1}(x) = \frac{p_j(x) - \Delta\lambda \frac{L_y(x)}{\Lambda} p_j(x)}{1 - \frac{\Delta\lambda}{\Lambda}\mathbb{E}\left[L_y(x)\right]} + \mathcal{O}(\Delta\lambda^2),$$

$$\frac{p_{j+1}(x) - p_j(x)}{\Delta\lambda} = -\frac{1}{\Lambda}\left[L_y(x)p_j(x) - \mathbb{E}\left[L_y(x)\right]p_{j+1}(x)\right] + \mathcal{O}(\Delta\lambda^2),$$

Take the limit as $\Delta\lambda \to 0$ to give

$$\frac{\partial p(x,\lambda)}{\partial \lambda} = -\frac{1}{\Lambda}\left[L_y(x) - \mathbb{E}\left[L_y(x)\right]\right]p(x,\lambda),$$

where $p_j(x) \to p_{j+1}(x)$ and so

$$\nabla \cdot (p(x,\lambda)\mu) = \frac{1}{\Lambda}\left[L_y(x) - \mathbb{E}\left[L_y(x)\right]\right]p(x,\lambda)$$

.

Particle flow

- Increasing number of papers on a technique called Particle Flow.
  - These papers report remarkable performance:
    - No resampling
    - No proposal distribution (no sampling!?)
    - High dimensions (traditionally requiring frequent resampling)
    - Impressive RMSE
  - Particle flow does not propose an explicit method to approximate filtering distributions.

Particle flow

- Given a family of distributions:
  - $p_0(x)$, which is easy to sample from
  - $p_\Lambda(x)$, which is what we are interested in
  - $p_\lambda(x)$, which is between the two
- The intermediate distribution is defined as

$$p_\lambda(x) = \frac{p_0(x)\left[\frac{p_\Lambda(x)}{p_0(x)}\right]^{\lambda/\Lambda}}{\int p_0(x')\left[\frac{p_\Lambda(x')}{p_0(x')}\right]^{\lambda/\Lambda} dx'}$$

- Key idea: λ evolves continuously between $\lambda = 0$ and $\lambda = \Lambda$.

Particle flow

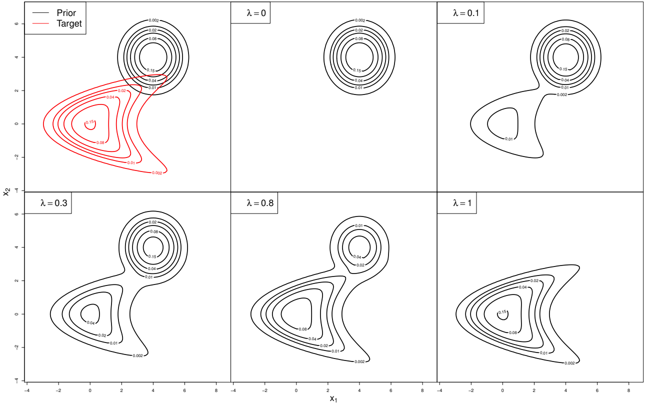


Figure 1: Intermediate distributions for particle flow

Stochastic Particle flow

- Stochastic version of the particle flow, by solving a stochastic differential equation (SDE) that describes the evolution w.r.t. $\lambda \in [0, \infty)$ of the samples $x(\lambda)^{(i)} \sim p_\lambda(x)$.
- If one starts with samples from $p_0(x)$ and propagates them through $0 \leq \lambda < \infty$ by simulating from the SDE, the samples become approximately $x(\lambda)^{(i)} \sim p_\Lambda(x) = \pi(x)$ for $\lambda \to \infty$.
- It is easy to demonstrate that the SDE that provides the described process can be achieved by the Langevin diffusion process

$$dx = \frac{1}{2} D(x) \nabla_x \log[\pi(x)] \, d\lambda + D(x)^{1/2} \, dw_\lambda,$$

where $\{w_\lambda\}$ is a standard Wiener process, $D(x)$ is the diffusion matrix, and $\pi(x)$ is the target distribution.

Stochastic Particle flow

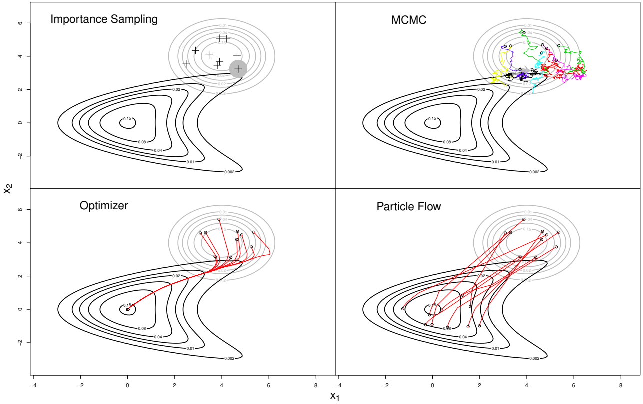


Figure 2: New particle flow in the context of other methods

Exemplar run

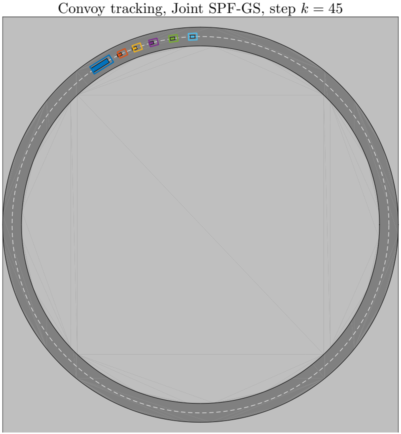


Figure 3: Exemplar run for the convoy tracking problem

Log RMSE x number of vehicles

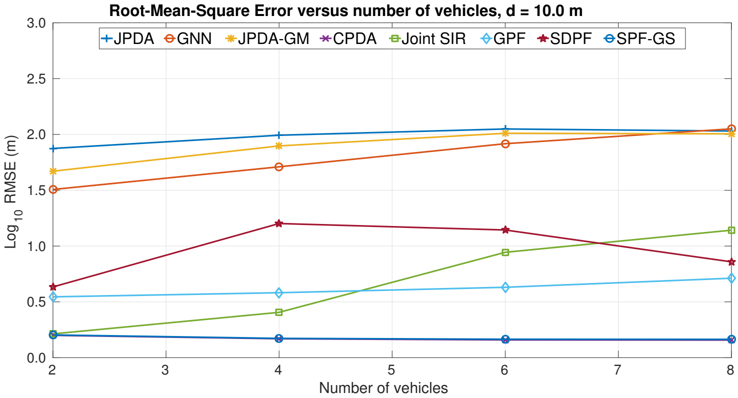


Figure 4: Logarithm of RMSE versus number of vehicles

Questions