Piecewise deterministic Markov processes for Monte Carlo

Markov lecture
INFORMS, Nashville, 14th November 2016

Gareth Roberts, University of Warwick

Joint work with Paul Fearnhead, Joris Bierkens, Murray Pollock, Adam Johansen, and Krys Latuszynski

WARWICK
THE UNIVERSITY OF WARWICK

CRiSM

i-like.org.uk

Talk outline

Markov was an ingenious mathematician fascinated by the mathematical properties of the chains he introduced.

Talk outline

Markov was an ingenious mathematician fascinated by the mathematical properties of the chains he introduced.

Did he ever envisage how useful they would become in modelling and simulation?

Talk outline

- Introduce PDMP
- The zig-zag as an alternative to MCMC. "The Zig-Zag Process and Super-Efficient Sampling for Bayesian Analysis of Big Data" arXiv:1607.03188
- The scale algorithm. "The Scalable Langevin Exact Algorithm: Bayesian Inference for Big Data" arXiv: 1609.03436
- CIS (Continuous time importance sampling) Joint work with Krys Latuszynski and Paul Fearnhead.

Currently writing a review paper covering all this material.

Piecewise-deterministic Markov processes

Continuous time stochastic process, denote by Z_t.

The dynamics of the PDP involves random events, with deterministic dynamics between events and possibly random transitions at events.

(i) The deterministic dynamics. eg specified through an ODE

$$\frac{\mathrm{d}z_t}{\mathrm{d}t} = \Phi(z_t), \tag{1}$$

So

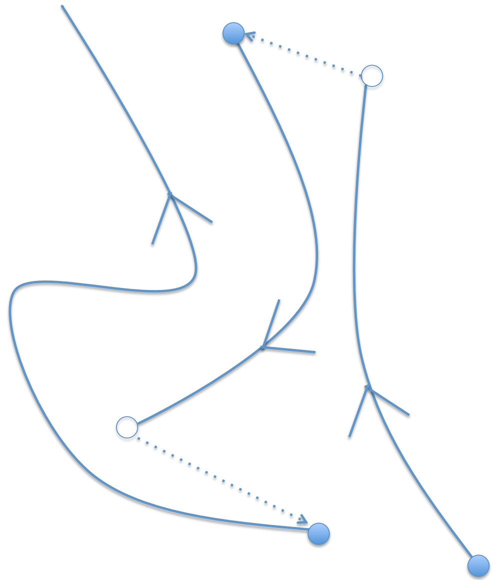
$$z_{s+t} = \Psi(z_t, s)$$

for some function Ψ.

(ii) The event rate. Events occur at rate, $\lambda(z_t)$,

(iii) The transition distribution at events. At each event time τ, Z changes according to some transition kernel

PDMP



PDMP

Date back to 1951 paper by Mark Kac on the telegraph process.

Mathematical foundations: Davis (1984, JRSS B)

Instrinsically continuous in time unike (almost all) algorithms. Why would they ever be useful for simulation?

Unlike diffusion processes they are comparatively understudied, and underused (either for models or in stochastic simulation).

PDMP

Date back to 1951 paper by Mark Kac on the telegraph process.

Mathematical foundations: Davis (1984, JRSS B)

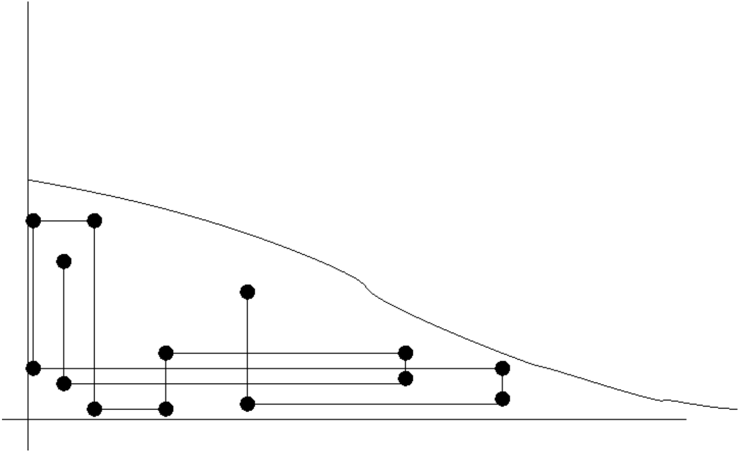
Instrinsically continuous in time unike (almost all) algorithms. Why would they ever be useful for simulation?

Unlike diffusion processes they are comparatively understudied, and underused (either for models or in stochastic simulation).

.... until recently

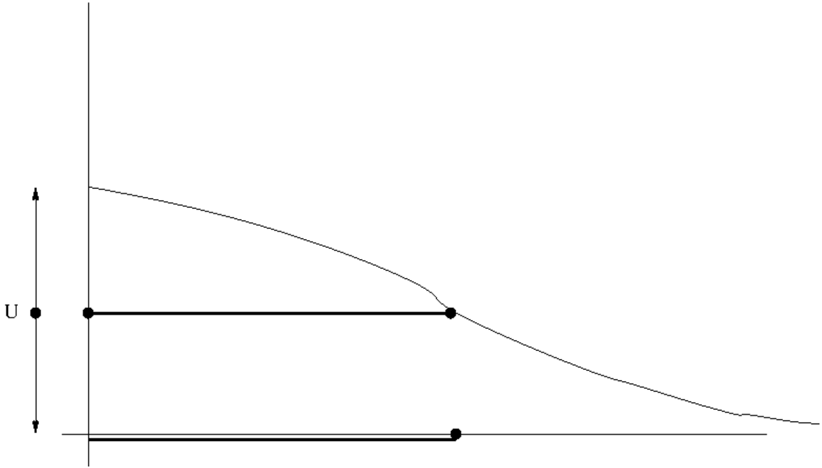
The slice sampler

An MCMC method for simulating from a target density by introducing an auxiliary variable.



Well understood theoretically, eg R + Rosenthal (1999)

An alternative



Metropolis-Hastings [Metropolis et al. 1953, Hastings 1970]

- S finite set (*state space*)
- $Q(x, y)$ transition probabilities on S (*proposal chain*)
- $\pi(x)$ a probability distribution on S (*target distribution*)

Define acceptance probabilities

$$A(x, y) = \min\left(1, \frac{\pi(y)Q(y, x)}{\pi(x)Q(x, y)}\right).$$

The Metropolis-Hastings (MH) chain is

$$P(x, y) = \begin{cases} Q(x, y)A(x, y) & y \neq x, \\ 1 - \sum_{z \neq x} Q(x, z)A(x, z) & y = x. \end{cases}$$

The MH chain is reversible:

$$\pi(x)P(x, y) = \pi(y)P(y, x) \quad \forall x, y \in S.$$

In particular, π is invariant for P.

Non-reversibility for MCMC?

The fact that MH is reversible is good because

- Beautiful clean mathematical theory: Markov chain transition operator is self-adjoint, spectrum is real, if geometrically ergodic then CLTs hold for all L^2 functions ...

Non-reversibility for MCMC?

The fact that MH is reversible is good because

- Beautiful clean mathematical theory: Markov chain transition operator is self-adjoint, spectrum is real, if geometrically ergodic then CLTs hold for all L^2 functions ...
- Detailed balance is a local condition - crucial for implementability, don't need to do an integral to decide whether to accept or reject.

Non-reversibility for MCMC?

The fact that MH is reversible is good because

- Beautiful clean mathematical theory: Markov chain transition operator is self-adjoint, spectrum is real, if geometrically ergodic then CLTs hold for all L^2 functions ...
- Detailed balance is a local condition - crucial for implementability, don't need to do an integral to decide whether to accept or reject.

So why should we bother to look further?

Non-reversibility for MCMC?

BUT it has long been known in probability that non-reversible chains can sometimes converge much more rapidly than reversible ones (see for instance Hwang, Hwang-Ma and Sheu (1993), Chen Lovasz and Pak (1999), Diaconis, Holmes and Neal (2000).

Non-reversibility for MCMC?

BUT it has long been known in probability that non-reversible chains can sometimes converge much more rapidly than reversible ones (see for instance Hwang, Hwang-Ma and Sheu (1993), Chen Lovasz and Pak (1999), Diaconis, Holmes and Neal (2000).

Hamiltonian MCMC (Hybrid Monte Carlo) tries to construct chains with non-reversible character, but ultimately it is also reversible because of the accept/reject step.

Metropolis-Hastings

[Metropolis et al. 1953, Hastings 1970]

- S finite set (*state space*)
- $Q(x, y)$ transition probabilities on S (*proposal chain*)
- $\pi(x)$ a probability distribution on S (*target distribution*)

Define acceptance probabilities

$$A(x, y) = \min\left(1, \frac{\pi(y)Q(y, x)}{\pi(x)Q(x, y)}\right).$$

The Metropolis-Hastings (MH) chain is

$$P(x, y) = \begin{cases} Q(x, y)A(x, y) & y \neq x, \\ 1 - \sum_{z \neq x} Q(x, z)A(x, z) & y = x. \end{cases}$$

The MH chain is reversible:

$$\pi(x)P(x, y) = \pi(y)P(y, x) \quad \forall x, y \in S.$$

In particular, π is invariant for P.

Non-Reversible Metropolis-Hastings [Bierkens, 2015]

- S finite set (*state space*)
- $Q(x, y)$ transition probabilities on S (*proposal chain*)
- $\pi(x)$ a probability distribution on S (*target distribution*)
- $\Gamma \in \mathbb{R}^{S \times S}$: skew-symmetric matrix with zero row-sums (vorticity matrix)

Define acceptance probabilities

$$A(x, y) = \min\left(1, \frac{\pi(y)Q(y, x) + \Gamma(x, y)}{\pi(x)Q(x, y)}\right).$$

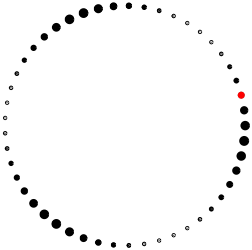
The non-reversible Metropolis-Hastings (MH) chain is

$$P(x, y) = \begin{cases} Q(x, y)A(x, y) & y \neq x, \\ 1 - \sum_{z \neq x} Q(x, z)A(x, z) & y = x. \end{cases}$$

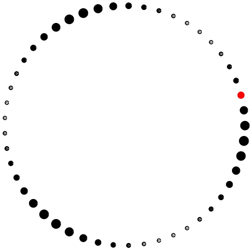
The NRMH chain is non-reversible:

$$\pi(x)P(x, y) \neq \pi(y)P(y, x) \quad \exists x, y \in S.$$

But π is invariant for P.



: Metropolis-Hastings



: Non-reversible Metropolis-Hastings

Cycles and lifting

Recall Γ skew-symmetric with zero row sums.

Also want acceptance probability

$$A(x, y) = \min\left(1, \frac{\pi(y)Q(y,x) + \Gamma(x,y)}{\pi(x)Q(x,y)}\right)$$

to be non-negative.

A 4-state example illustrates: No cycles $\Rightarrow$ no non-reversible Markov chain.

Cycles and lifting

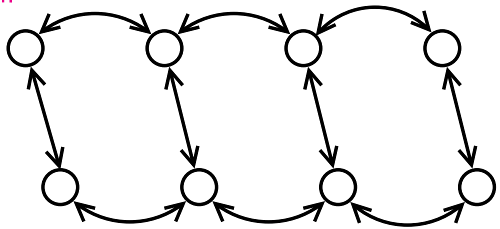
Recall Γ skew-symmetric with zero row sums.

Also want acceptance probability

$$A(x,y) = \min\left(1, \frac{\pi(y)Q(y,x) + \Gamma(x,y)}{\pi(x)Q(x,y)}\right)$$

to be non-negative.

A 4-state example illustrates: No cycles $\Rightarrow$ no non-reversible Markov chain.

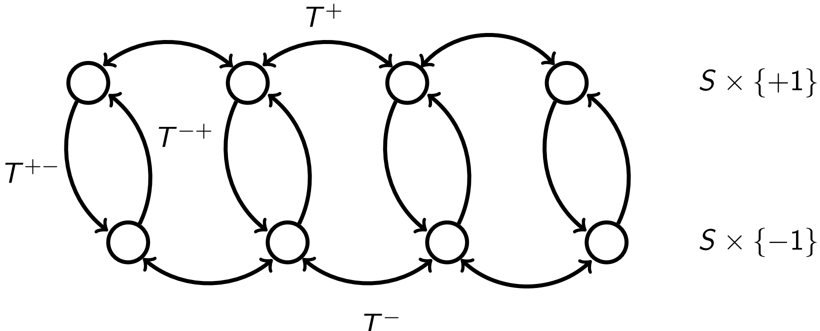


How to construct lifted MCMC algorithms?

A general lifted Markov chain [Turitsyn, Chertkov, Vucelja, 2011]

- State space S augmented to $S^\sharp = S \times \{-1, +1\}$.
- T^+, T^- are sub-Markov transition matrices on S.
- $T^\pm$ satisfy skew-detailed balance: for all $x, y \in S$, $\pi(x) T^+(x, y) = \pi(y) T^-(y, x)$.
- T^{-+}, T^{+-} transitions between replicas, e.g.

$$T^{-+}(x) = \max\left(0, \sum_{y \neq x} (T^+(x, y) - T^-(x, y))\right).$$



Lifted Metropolis-Hastings [Turitsyn, Chertkov, Vucelja, 2011]

How to choose T^+ and T^-?

Introduce a quantity of interest: $\eta : S \to \mathbb{R}$

Take (Q, π) reversible, e.g. Metropolis-Hastings chain.

Define

$$T^+(x,y) := \begin{cases} Q(x,y) & \text{if } \eta(y) \geq \eta(x) \\ 0 & \text{if } \eta(y) < \eta(x). \end{cases}$$

$$T^-(x,y) := \begin{cases} Q(x,y) & \text{if } \eta(y) \leq \eta(x) \\ 0 & \text{if } \eta(y) > \eta(x). \end{cases}$$

Then skew-detailed balance is satisfied:

$$\pi(x) T^+(x,y) = \pi(y) T^-(y,x) \quad \text{for all } x, y.$$

In practice, Lifted Metropolis-Hastings algorithm:

- Propose according to proposal chain Q
- If move is allowed, accept with MH acceptance probability
- If move is not allowed, possibly switch replica.

Does lifting solve the non-reversible MCMC problem?

The problem is that we need to know the switching probabilities, eg

$$T^{-+}(x) = \max\left(0, \sum_{y \neq x} (T^{+}(x,y) - T^{-}(x,y))\right).$$

This will typically be difficult to calculate, usually impossible in continuous state spaces.

Does lifting solve the non-reversible MCMC problem?

The problem is that we need to know the switching probabilities, eg

$$T^{-+}(x) = \max\left(0, \sum_{y \neq x} (T^{+}(x,y) - T^{-}(x,y))\right).$$

This will typically be difficult to calculate, usually impossible in continuous state spaces.

So lifting is not generally applicable

But, mathematically we can take a limit of smaller proposed moves and speed up the process to obtain a continuous time limit.

We initially did this for the Curie-Weiss model in statistical physics (http://arxiv.org/abs/1509.00302. to appear in *Annals of Applied Probability*).

But, mathematically we can take a limit of smaller proposed moves and speed up the process to obtain a continuous time limit.

We initially did this for the Curie-Weiss model in statistical physics (http://arxiv.org/abs/1509.00302. to appear in *Annals of Applied Probability*).

This was purely for mathematical reasons to understand lifting for the Curie-Weiss model.

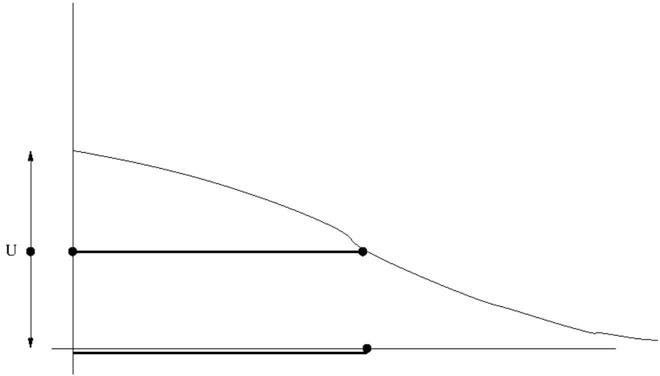
But, mathematically we can take a limit of smaller proposed moves and speed up the process to obtain a continuous time limit.

We initially did this for the Curie-Weiss model in statistical physics (http://arxiv.org/abs/1509.00302. to appear in *Annals of Applied Probability*).

This was purely for mathematical reasons to understand lifting for the Curie-Weiss model.

But the continuous-time limit argument extends easily to general target densities.

Another look at our initial example ...



Instead of apriori drawing the uniform random variable, change direction with hazard rate

$$\max\{0, -(\log \pi)'(x)\}$$

One-dimensional zig zag process

Langevin diffusion generator (for comparison):

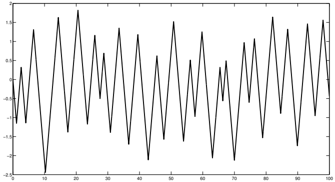
$$Lf(y) = (\log \pi)'(y)\frac{df}{dy} + \frac{d^2 f}{dy^2}, \quad y \in \mathbb{R}.$$

Invariant density $\pi(y)$

Zig zag process generator:

$$Lf(y,j) = aj\frac{df}{dy} + \lambda(y,j)(f(y,-j) - f(y,j)), \quad y \in \mathbb{R}, j \in \{-1,+1\}.$$

Here speed $a > 0$ and switching rate $\lambda(y,j) \geq 0$.



Relation between speed, switching rate and potential

$$Lf(y,j) = aj\frac{df}{dy} + \lambda(y,j)(f(y,-j) - f(y,j)), \quad y \in \mathbb{R}, j \in \{-1,+1\}.$$

speed a, switching rate λ and target density $\pi(x)$

$$\lambda(y,j) - \lambda(y,-j) = -aj(\log \pi)'(y).$$

Equivalently

$$\lambda(y,j) = \gamma(y) + \max\left(0, -aj(\log \pi)'(y)\right), \quad \gamma(y) \geq 0.$$

Related work

- [Goldstein, 1951],[Kac, 1974]: constant jump rate λ, relation to *telegraph equation*
- [Peters, de With, Physical Review E, 2012]: first occurrence of the zig zag process for sampling from general targets, with multi-dimensional, non-ergodic extension.
- [Fontbana, Guérin, Malrieu, 2012, 2015]: convergence analysis of zig zag process under convex potential.
- [Monmarché, 2014]: simulated annealing using "run-and-tumble" process.
- [Bouchard-Côté et al., 2015]: bouncy particle sampler.

Implementation

How do we simulate continuous time stochastic process like this?

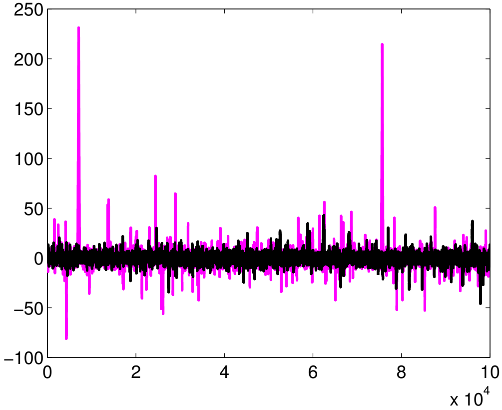
By using thinned poisson processes

For example, if $|(\log \pi)'(x)| < c$, simulate a Poisson process of rate c (by simulating the exponential inter-arrival times). Then at each poisson time, we accept as a direction change with probability $\max(-(\log \pi)'(x), 0)/c$.

This makes the algorithm inexpensive to implement as we only need to calculate $(\log \pi)'(x)$ occasionally.

There are many other details .... though the method is not so complicated.

Zig zag process for sampling the Cauchy distribution



$T = 10,000$

Multi-dimensional zig zag process

Multi-dimensional zig zag process: here we have a multi-dimensional binary velocity, eg $(1, -1, -1, 1, 1, -1, 1, 1)$.

Efficient sampling (currently for potentials with locally Lipschitz gradients in multiple dimensions, but with obvious ways to extend).

Subsampling

Motivation: intractable likelihood problems where calculating π at any one fixed location is prohibitively expensive (given that very many evaluations will be required to run the algorithm. For this talk, concentrate on the Bayesian setting:

$$\pi(x) = \prod_{i=1}^{N} \pi_i(x)$$

Eg we have N observations (but this method is not in any way restricted to the independent data case).

Subsampling

Motivation: intractable likelihood problems where calculating π at any one fixed location is prohibitively expensive (given that very many evaluations will be required to run the algorithm. For this talk, concentrate on the Bayesian setting:

$$\pi(x) = \prod_{i=1}^{N} \pi_i(x)$$

Eg we have N observations (but this method is not in any way restricted to the independent data case).

Aim to be lazy and only use a small number of the terms in the product.

Subsampling

Motivation: intractable likelihood problems where calculating π at any one fixed location is prohibitively expensive (given that very many evaluations will be required to run the algorithm. For this talk, concentrate on the Bayesian setting:

$$\pi(x) = \prod_{i=1}^{N} \pi_i(x)$$

Eg we have N observations (but this method is not in any way restricted to the independent data case).

Aim to be lazy and only use a small number of the terms in the product.

For instance we might try pseudo-marginal MCMC (Beaumont, 2003, Andrieu and Roberts, 2009).But that would require an unbiased non-negative estimate of $\pi(x)$ with variance which is stable as a function of N.

Subsampling

Motivation: intractable likelihood problems where calculating π at any one fixed location is prohibitively expensive (given that very many evaluations will be required to run the algorithm. For this talk, concentrate on the Bayesian setting:

$$\pi(x) = \prod_{i=1}^{N} \pi_i(x)$$

Eg we have N observations (but this method is not in any way restricted to the independent data case).

Aim to be lazy and only use a small number of the terms in the product.

For instance we might try pseudo-marginal MCMC (Beaumont, 2003, Andrieu and Roberts, 2009).But that would require an unbiased non-negative estimate of $\pi(x)$ with variance which is stable as a function of N. But this is not possible for a product without computing cost which is at least $O(N)$.

Subsampling within PDMP

PDMP for the exploration of high-dimensional distributions (such as zig-zag or the ScaLE algorithm, Fearnhead, Johansen, Pollock and Roberts, 2016) typically use $\log \pi(x)$ rather than $\pi(x)$ and

$$\log \pi(x) = \sum_{i=1}^{N} \log \pi_i(x)$$

for which there are well-behaved $O(1)$ cost, $O(1)$ variance (or sometime a little worse). Can we use this?

Zig zag switching rate $\max\left(0, -j \sum_{i=1}^{N} (\log \pi)'_i(x)\right) \rightsquigarrow O(N)$ calculation at every switch

Subsampling for zig-zag

Sub-sampling

- Determine global upper bound M for switching rate
- Simulate Exponential(M) random variable T
- Generate $I \sim \text{discrete}(\{1, \dots, N\})$
- Accept the generated T as a "switching time" with probability $N \max\left(0, -j(\log \pi_I)'(Y(T))\right)/M$

Theorem: This works! (invariant distribution π)

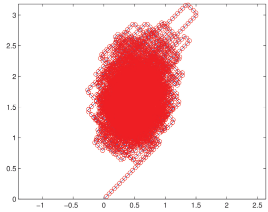
Subsampling + control variates

Crudely, for an $O(1)$ update in state space:

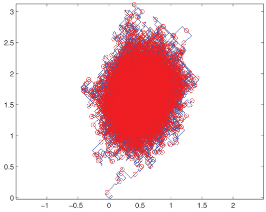
- Without subsampling, $O(N)$ computations required
- Using subsampling, gain factor $N^{1/2}$ $\rightsquigarrow$ complexity $O(N^{1/2})$ per step
- Using control variates, gain additional factor $N^{1/2}$ $\rightsquigarrow$ complexity $O(1)$ per step

Superefficiency We call an epoch the time taken to make one function evaluation of the target density π. The control variate subsampled zig-zag is superefficient in the sense that the effective sample size from running the algorithm per epoch diverges.

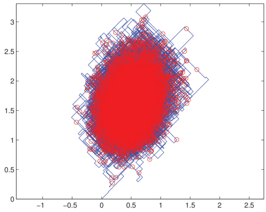
Subsampling + control variates – Logistic growth



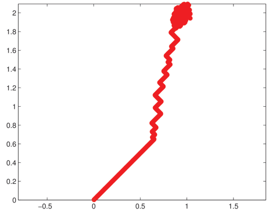
: $N = 100$



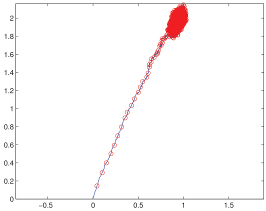
: $N = 100$



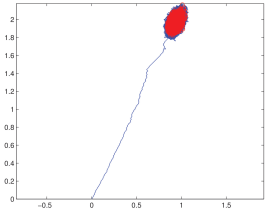
: $N = 100$



: $N = 10,000$



: $N = 10,000$



: $N = 10,000$

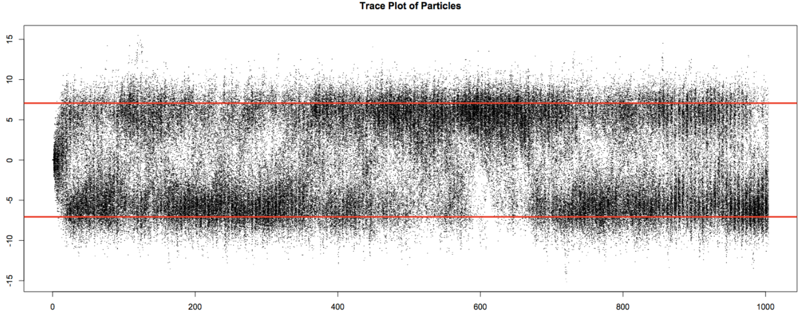
[Bierkens, Roberts, 2015, `http://arxiv.org/abs/1509.00302`]

The ScaLE Algorithm

The Scalable Langevin Exact algorithm.

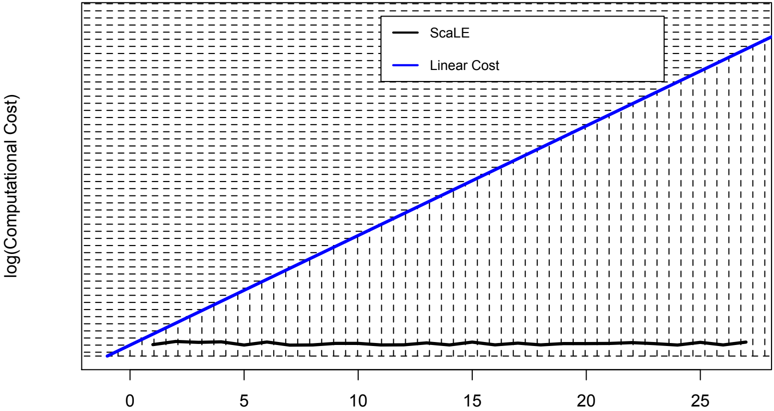
Uses quasi-stationary Monte Carlo which requires Sequential Monte Carlo to implement.

Many trajectories of weighted particles which regenerate from the other particles once they die.



Scaling

Much more complicated than zig-zag, however has perfect scaling properties for big data (also relies on subsampling and control variate strategies).



CIS

Continuous-time importance sampling.

Not an MCMC method to explore a fixed target distribution.

Gives unbiased estimators of functionals of diffusions at any fixed timepoints.

Outputs a weighted particle. The weight stays constant till some random time, upon which the particle jumps to a new location. (Hence can be written as a PDMP.)

Can be used for an arbitrary multi-dimensional diffusion process

Final remarks

- PDMPs have many uses for simulation of stochastic processes (even those very different from PDMPs) as well as steady state simulation.

Final remarks

- PDMPs have many uses for simulation of stochastic processes (even those very different from PDMPs) as well as steady state simulation.
- Subsampling and control-variate tweaks greatly improve efficiency in certain situations. PDMP are particularly amenable to this.

Final remarks

- PDMPs have many uses for simulation of stochastic processes (even those very different from PDMPs) as well as steady state simulation.
- Subsampling and control-variate tweaks greatly improve efficiency in certain situations. PDMP are particularly amenable to this.
- More work is needed on studying the theoretical and empirical properties of these algorithms, and exploiting their flexibility.

Final remarks

- PDMPs have many uses for simulation of stochastic processes (even those very different from PDMPs) as well as steady state simulation.
- Subsampling and control-variate tweaks greatly improve efficiency in certain situations. PDMP are particularly amenable to this.
- More work is needed on studying the theoretical and empirical properties of these algorithms, and exploiting their flexibility.
- Zigzag is a flexible and usually easy-to-implement method for simulating from a target distribution.

Final remarks

- PDMPs have many uses for simulation of stochastic processes (even those very different from PDMPs) as well as steady state simulation.
- Subsampling and control-variate tweaks greatly improve efficiency in certain situations. PDMP are particularly amenable to this.
- More work is needed on studying the theoretical and empirical properties of these algorithms, and exploiting their flexibility.
- Zigzag is a flexible and usually easy-to-implement method for simulating from a target distribution.
- Can zigzag be a competitor to Hamiltonian MCMC?

Final remarks

- PDMPs have many uses for simulation of stochastic processes (even those very different from PDMPs) as well as steady state simulation.
- Subsampling and control-variate tweaks greatly improve efficiency in certain situations. PDMP are particularly amenable to this.
- More work is needed on studying the theoretical and empirical properties of these algorithms, and exploiting their flexibility.
- Zigzag is a flexible and usually easy-to-implement method for simulating from a target distribution.
- Can zigzag be a competitor to Hamiltonian MCMC?