KU LEUVEN

Adjoint Monte Carlo particle methods with reversible random number generators

E. Løvbak, G. Samaey

F. Blondeel, A. Lee, L. Vanroye, A. Van Barel, S. Vandewalle

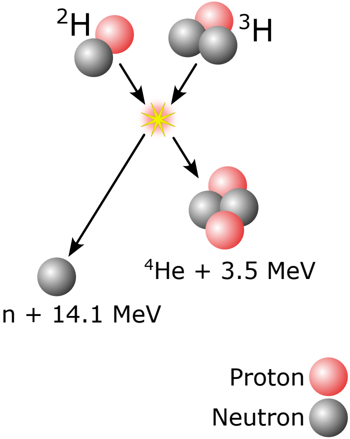
KU Leuven, Department of Computer Science, NUMA Section

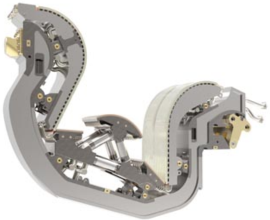
fwo Research Foundation Flanders Opening new horizons

NUMA

DEPARTMENT OF COMPUTER SCIENCE

Motivation: nuclear fusion in tokamaks





$$Q \geq 10 : \quad 50\text{MW} \rightarrow 500\text{MW}$$

china eu india japan korea russia usa

Research Foundation Flanders
Opening new horizons

NUMA

Didactical example: Heat equation in 1D

- **Macroscopic view**

$$\frac{\partial}{\partial t}\mathcal{T}(x,t) - \frac{\partial^2}{\partial x^2}\mathcal{T}(x,t) + u(x)\mathcal{T}(x,t) = 0$$

- **Particle view**

$$\{Q_{p,\tau}\}_{p=1}^{P} = \left\{ \begin{bmatrix} X_{p,\tau} \\ W_{p,\tau} \end{bmatrix} \right\}_{p=1}^{P}, \qquad \tau = 0, \ldots, T-1$$

  - Diffusion step

$$X_{p,\tau+1} = X_{p,\tau} + \sqrt{2\Delta t}\,\xi_{p,\tau} \quad \xi_{p,\tau} \sim \mathcal{N}(0,1)$$

  - Reweighting step

$$W_{p,\tau+1} = W_{p,\tau} \exp\left(-\Delta t\, \hat{u}\left(X_{p,\tau+1}\right)\right)$$

Optimization problem

- **Continuous**

$$\min_{u(x)} \mathcal{J}(\mathcal{T}(x,t),u(x,t)) = \int_0^\infty \int_0^L \frac{1}{2}\mathcal{T}(x,t)^2 \mathrm{d}x\,\mathrm{d}t + \kappa \int_0^L \frac{1}{2}u(x)^2 \mathrm{d}x$$

$$\text{subject to } \frac{\partial}{\partial t}\mathcal{T}(x,t) - \frac{\partial^2}{\partial x^2}\mathcal{T}(x,t) + u(x)\mathcal{T}(x,t) = 0,$$

$$\mathcal{T}(x,0) = \mathcal{T}_0(x), \quad \mathcal{T}(0,t) = \mathcal{T}(L,t)$$

- **Discrete**

$$\mathcal{J}(\mathcal{T}(x,t),u(x,t)) \approx \hat{\mathcal{J}}(\hat{\mathcal{T}},\hat{u}) = \Delta t \sum_{\tau=0}^{T}{}'' \Delta x \frac{1}{2}\hat{\mathcal{T}}_\tau^\top \hat{\mathcal{T}}_\tau + \nu \Delta x \frac{1}{2}\hat{u}^\top \hat{u}$$

$$\hat{\mathcal{T}}_{\tau,n} = \sum_{p=1}^{P} \frac{1}{\Delta x}\mathcal{I}_n\left(X_{p,\tau}\right) W_{p,\tau}$$

Adjoint-based optimization

- PDE-constrained optimization

$$\min_{u} \mathcal{J}(q,u), \quad \text{subject to} \quad \mathcal{B}(q;u) = 0$$

- Naive application of chain rule

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}u}(q(u),u) = \frac{\partial \mathcal{J}}{\partial u}(q,u) + \frac{\partial \mathcal{J}}{\partial q}(q,u)\frac{\mathrm{d}q}{\mathrm{d}u}(u)$$

Adjoint-based optimization

- PDE-constrained optimization

$$\min_{u} \mathcal{J}(q,u), \quad \text{subject to} \quad \mathcal{B}(q;u) = 0$$

- Naive application of chain rule

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}u}(q(u),u) = \frac{\partial \mathcal{J}}{\partial u}(q,u) + \frac{\partial \mathcal{J}}{\partial q}(q,u)\frac{\mathrm{d}q}{\mathrm{d}u}(u)$$

Adjoint-based optimization

- PDE-constrained optimization

$$\min_u \mathcal{J}(q,u), \quad \text{subject to} \quad \mathcal{B}(q;u) = 0$$

- Naive application of chain rule

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}u}(q(u),u) = \frac{\partial \mathcal{J}}{\partial u}(q,u) + \frac{\partial \mathcal{J}}{\partial q}(q,u)\frac{\mathrm{d}q}{\mathrm{d}u}(u)$$

- Solution: Lagrangian $\mathcal{L}(q,q^*,u) = \mathcal{J}(q,u) + (q^*, \mathcal{B}(q,u))$

$$\mathcal{B}(q,u) = 0 \qquad \text{State equation}$$

$$\frac{\partial \mathcal{J}}{\partial q}^*(q,u) + \frac{\partial \mathcal{B}}{\partial q}^*(q,u)q^* = 0 \qquad \text{Adjoint equation}$$

$$\frac{\partial \mathcal{J}}{\partial u}^*(q,u) + \frac{\partial \mathcal{B}}{\partial u}^*(q,u)q^* = 0 \qquad \text{Design equation}$$

Discrete adjoint

- Linearized discretization around $\hat{q}' = \hat{q}(\hat{u})$

$$\hat{\mathcal{B}}(\hat{q},\hat{u}) \approx \hat{\mathcal{B}}(\hat{q}',\hat{u}) + \frac{\partial \hat{\mathcal{B}}}{\partial \hat{q}}(\hat{q}',\hat{u})(\hat{q}-\hat{q}') = 0$$

is a matrix-vector system

Discrete adjoint

- Linearized discretization around $\hat{q}' = \hat{q}(\hat{u})$

$$\hat{\mathcal{B}}(\hat{q}, \hat{u}) \approx \qquad \frac{\partial \hat{\mathcal{B}}}{\partial \hat{q}}(\hat{q}', \hat{u})(\hat{q} - \hat{q}') = 0$$

  is a matrix-vector system
- Adjoint equation

$$\frac{\partial \hat{\mathcal{J}}}{\partial \hat{q}}^{\top}(\hat{q}', \hat{u}) + \frac{\partial \hat{\mathcal{B}}}{\partial \hat{q}}^{\top}(\hat{q}', \hat{u})\hat{q}^* = 0$$

Monte Carlo as a matrix

- For each particle p and time step τ

$$B_{p,\tau+1}(Q_{p,\tau+1}, Q_{p,\tau}; \hat{u}) = \begin{bmatrix} X_{p,\tau+1} - X_{p,\tau} - \sqrt{2\Delta t}\xi_{p,\tau} \\ W_{p,\tau+1} - W_{p,\tau} \exp\left(-\Delta t\, \hat{u}\left(X_{p,\tau+1}\right)\right) \end{bmatrix} = 0$$

- Particle Jacobian

$$\frac{\partial \hat{\mathcal{B}}_p}{\partial \hat{q}_p}(Q_p, \hat{u}) = \begin{bmatrix} \frac{\partial B_{p,1}}{\partial Q_{p,1}} & & & \\ \frac{\partial B_{p,2}}{\partial Q_{p,1}} & \frac{\partial B_{p,2}}{\partial Q_{p,2}} & & \\ & \ddots & \ddots & \\ & & \frac{\partial B_{p,T}}{\partial Q_{p,T-1}} & \frac{\partial B_{p,T}}{\partial Q_{p,T}} \end{bmatrix}$$

$$\frac{\partial B_{p,\tau+1}}{\partial Q_{p,\tau+1}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \frac{\partial B_{p,\tau+1}}{\partial Q_{p,\tau}} = \begin{bmatrix} -1 & 0 \\ 0 & -\exp\left(-\Delta t\, \hat{u}\left(X_{p,\tau+1}\right)\right) \end{bmatrix}$$

Research Foundation Flanders
Opening new horizons
NUMA
DEPARTMENT OF COMPUTER SCIENCE

Adjoint Monte Carlo

- Simulate particles from final state

$$Q^*_{p,T} = \begin{bmatrix} X^*_{p,T} \\ W^*_{p,T} \end{bmatrix} = -\frac{\partial \hat{\mathcal{J}}}{\partial Q_{p,T}}^\top = -\begin{bmatrix} 0 \\ \Delta t \hat{\mathcal{T}}\left(X_{p,T}\right) \end{bmatrix}$$

- Reverse time stepping

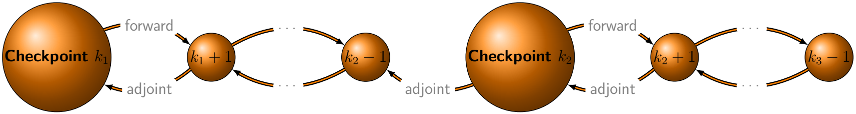
$$\begin{aligned} Q^*_{p,\tau} &= -\frac{\partial B_{p,\tau+1}}{\partial Q_{p,\tau}}^\top Q^*_{p,\tau+1} - \frac{\partial \hat{\mathcal{J}}}{\partial Q_{p,\tau}}^\top \\ &= \begin{bmatrix} X^*_{p,\tau+1} \\ \exp\left(-\Delta t\, \hat{u}\left(X_{p,\tau+1}\right)\right) W^*_{p,\tau+1} \end{bmatrix} - \begin{bmatrix} 0 \\ \Delta t \hat{\mathcal{T}}(X_{p,\tau}) \end{bmatrix} \end{aligned}$$

- Note that $X^*_{p,\tau} = 0$ for all τ

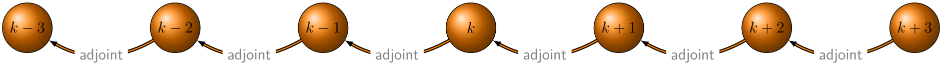
$$W^*_{p,\tau} = \exp\left(-\Delta t\, \hat{u}\left(X_{p,\tau+1}\right)\right) W^*_{p,\tau+1} - \Delta t \hat{\mathcal{T}}(X_{p,\tau})$$

Matching forward and adjoint simulations

- Same paths in forward/backward simulation
- Challenge: $P \times T$ large
- Solutions:
  - Checkpointing: 2 forward simulations + backward simulation



  - Generating the paths in reverse (this work)



Reversing a random number generator

- PCG: permuted congruential generator[1]
  - internal state ζ_k and constant vectors a, c, m

$$\zeta_{k+1} = a\zeta_k + c \mod m,$$

  - 1-way (permutation) function generates output from ζ_k
  - Passes `TestU01` with flying colors
- Reversing modular operations $\rightarrow$ reversed uniform sequence

$$\zeta_k = a^{-1}(\zeta_{k+1} - c) \mod m,$$

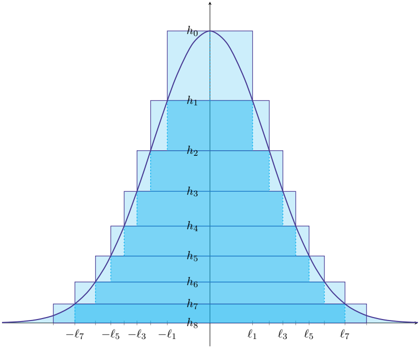
$$a^{-1} \equiv a^{m-2} \mod m$$

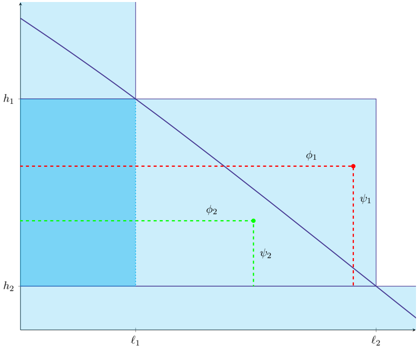
- Exponential distribution through inverse transform:

$$u \sim \mathcal{U}([0,1]) \Rightarrow -\lambda \ln(1-u) \sim \mathcal{E}(\lambda)$$

1: M.E. O'Neill, *PCG: A Family of Simple Fast Space-Efficient Statistically Good Algorithms for Random Number Generation.* Technical report HMC-CS-2014-0905, Harvey Mudd College (2014)

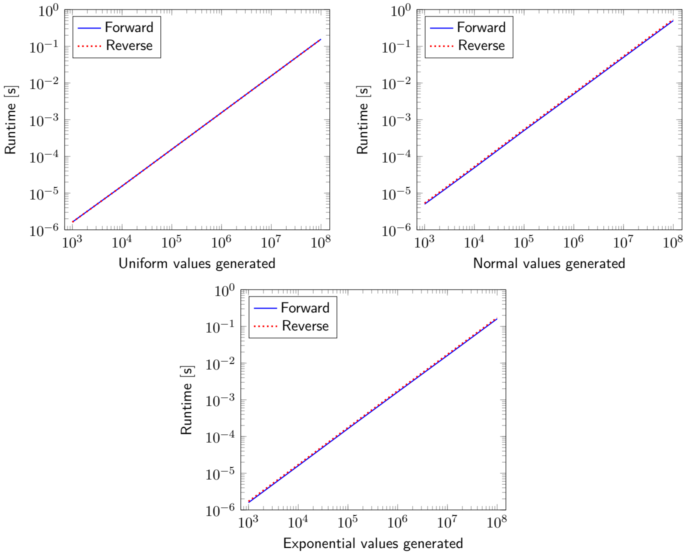
Normal distribution through Ziggurat



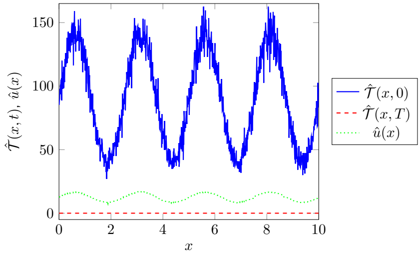


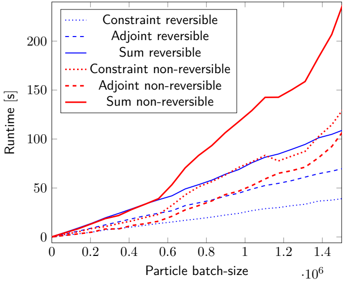
- Uses either 1, 2 or $1 + 2n,\ n = 1, 2, \ldots$ uniform values
- How many depends on the first value
- Solution: Seed second generator

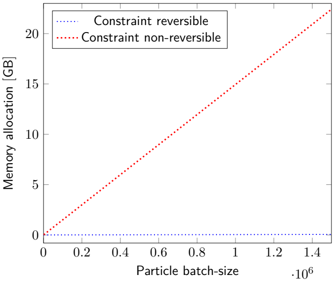
Generator timings



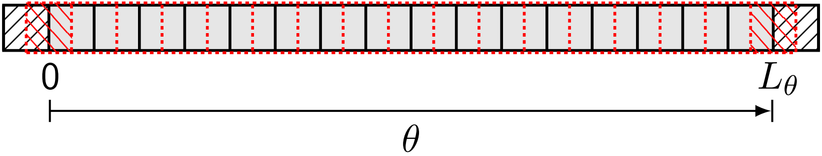
Optimal cooling







Fusion example: domain length optimization[2]



$$\mathcal{J}(q, L_\theta) = \frac{1}{2}\left(n_{\mathbf{i}} b_\theta u_{\|} - \Gamma_d \right)^2 \Big|_{L_\theta} + \frac{\kappa}{2}\left(L_\theta - L_0\right)^2$$

$$q = \left(n_{\mathbf{i}}, u_{\|}, f_{\mathbf{n}} \right)^\top$$

$n_{\mathbf{i}}$	plasma density
$u_{\|}$	plasma velocity
$f_{\mathbf{n}}$	neutral position-velocity

2: W. Dekeyser, *Optimal Plasma Edge Configurations for Next-Step Fusion Reactors.* PhD thesis (2014)

Plasma edge model

- Plasma $\Rightarrow$ Finite volume

$$\frac{\partial}{\partial\theta}\left(n_{\mathbf{i}} b_\theta u_\parallel\right) = S_{n_{\mathbf{i}}} - K_{\mathsf{d}} n_{\mathbf{i}}$$

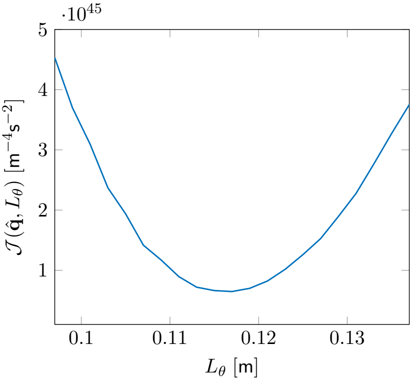
$$\frac{\partial}{\partial\theta}\left(m n_{\mathbf{i}} b_\theta {u_\parallel}^2 - \nu_{\mathsf{d}} \frac{\partial u_\parallel}{\partial\theta}\right) = S_{u_\parallel} - b_\theta \frac{\partial p}{\partial\theta}$$

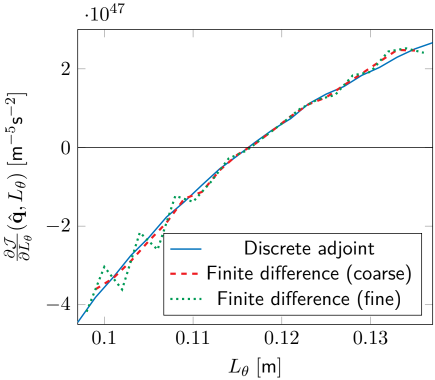
$$S_\psi = \int f_{\mathbf{n}}(\theta, v)\Psi(\theta, v)\mathrm{d}v, \quad \psi \in \{n_{\mathbf{i}}/u_\parallel\} \quad \Rightarrow \quad \text{Low-dimensional}$$

- Neutrals $\Rightarrow$ Monte Carlo

$$v\frac{\partial}{\partial\theta} f_{\mathbf{n}}(\theta, v) + K_{\mathsf{i}} f_{\mathbf{n}}(\theta, v) = S_{f_{\mathbf{n}}}\left(n_{\mathbf{i}}, u_\parallel\right) + K_{\mathsf{cx}} \int f_{\mathbf{n}}(\theta, v') C(v' \rightarrow v)\mathrm{d}v'$$

Preliminary results

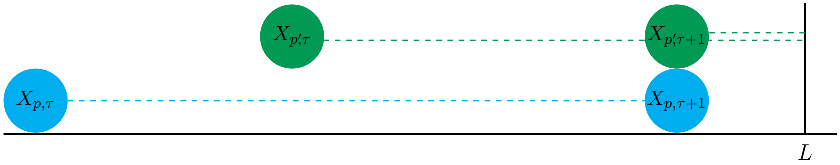




	$n_\mathbf{i}$-solver	$u_\parallel$-solver	Neutral solver
State	3.5×10^{-5} s	4.1×10^{-5} s	6.6 s
Adjoint	3.1×10^{-5} s	3.6×10^{-5} s	9.8 s

Can all simulations be reversed?

- In theory yes!
- ... but at what cost?
  - Reflections
  - Collisions in batched simulations
  - ...



Løvbak, E., Blondeel, F., Lee, A., Vanroye, L., Van Barel, A., Samaey, G., *Reversible random number generation for adjoint Monte Carlo simulation of the heat equation.* Monte Carlo and Quasi-Monte Carlo Methods - MCQMC 2022. **Submitted** (2023) *arXiv:2302.02778*

Løvbak, E., *Multilevel and adjoint Monte Carlo methods for plasma edge neutral particle models.* PhD thesis (2023)