January 26, 2016

Abstract

1 Introduction

We look for solutions of partial differential equations of advection-diffusion-type

$$\frac{\partial \rho(x,t)}{\partial t} + \frac{\partial (a(x)\rho(x,t))}{\partial x} = D\frac{\partial^2 \rho(x,t)}{\partial x^2} \tag{1}$$

in two ways:

- By explicitly solving eq. (1) using the discretization scheme[1]

$$\rho_i^{n+1} = \rho_i^n + \Delta t \left(\frac{D}{\Delta x^2}\left(\rho_{i+1}^n - 2\rho_i^n + \rho_{i-1}^n\right) - \frac{a(x)}{\Delta x}(\rho_i^n - \rho_{i-1}^n)\right) \tag{2}$$

  for the value of the probabilty density ρ at position $x = i\Delta x$ and time $t = (n+1)\Delta t$.

- By simulating an ensemble of N particles evolving according to the corresponding stochastic differential equation

$$\mathrm{d}x = a(x)\mathrm{d}t + \sqrt{2D}\cdot \mathrm{d}W_t. \tag{3}$$

  The position x^{n+1} of each particle at time $t = (n+1)\Delta t$ is simulated using the Euler-Maruyama scheme

$$x^{n+1} = x^n + a(x^n)\Delta t + \sqrt{2D\Delta t}\cdot \xi^n \tag{4}$$

  with $\xi^n \sim \mathcal{N}(0,1)$

2 Coarse time stepper

To simulate the time evolution of the density $\boldsymbol{\rho}(t)$, we construct a coarse time stepper $\boldsymbol{\Phi}_T$ which allows the performance of time-steps at the macroscopic level, using only the stochastic simulation of the position vectors of the N particles at the microscopic level, generated by eq. (4).

[1] First-order upwind scheme for the advective part combined with the Forward-Time Central-Space-method for the diffusive part

To achieve this, we will define two operators (lifting in section 2.1 and restriction in section 2.2) that relate the microscopic and macroscopic levels of description. Once these lifting $\mathcal{L}$ and restriction operators $\mathcal{R}$ have been constructed, a coarse time-stepper $\boldsymbol{\Phi}_T$ to evolve the macroscopic state $\boldsymbol{\rho}$ over a time interval of length $n\Delta t$ is constructed as a three-step-procedure (lift–evolve–restrict):

$$\boldsymbol{\rho}(t + n\Delta t) = \boldsymbol{\Phi}_T(\boldsymbol{\rho}) = (\mathcal{R} \circ \mathcal{E}(n\Delta t) \circ \mathcal{L}(\omega))(\boldsymbol{\rho}(t)) \tag{5}$$

where $\mathcal{E}(n\Delta t)(\boldsymbol{\rho}(t))$ is the simulation of the SDE for N particles over n timesteps.

2.1 Lifting: $\boldsymbol{\rho} \to \mathbf{x}$

Given the density $\boldsymbol{\rho}$, we need to sample a position vector x_i for every particle $i \leq N$. We use the inverse transformation method for this. The particle does not only gets an inital position, but also a seed for generating random steps in the simulation.

2.2 Restriction: $\mathbf{x} \to \boldsymbol{\rho}$

The restriction operator $\mathcal{R} : \mathbb{Q}^N \to \mathbb{Q}^k$ maps the microscopic state $\mathbf{x}$ (determined by the position vectors of N particles) to a density $\boldsymbol{\rho}$, discretisized in k bins. This is done by counting the number of particles in every bin Δ_j for $1 \leq j \leq k$:

$$\frac{1}{N} \sum_{i=1}^{N} w^i \cdot \chi_{\Delta_j}(x^i) = \rho_j \tag{6}$$

with

$$\chi_{\Delta_j}(x) = \begin{cases} 1 & \text{if } x \in \Delta_j, \\ 0 & \text{if } x \notin \Delta_j. \end{cases} \tag{7}$$

and setting all weights $w_i = 1$ for $1 \leq i \leq N$.

The reason why we explicitly introduced these weights in the restriction operator will be clarified in section 10 where we will need to evaluate the coarse time stepper $\boldsymbol{\Phi}_T(\boldsymbol{\rho}+\varepsilon\mathbf{v})$, now applied to the density shifted with a certain perturbation $\varepsilon\mathbf{v}$. To evaluate the perturbated restriction-operator we will use the weights w^i_ε, determined such that

$$\frac{1}{N} \sum_{i=1}^{N} w^i_\varepsilon \cdot \chi_{\Delta_j}(x^i) = \rho_j + \varepsilon v_j. \tag{8}$$

We do this by computing the weight per bin as $w^j_\varepsilon = 1 + \frac{\varepsilon v_j}{\rho_j}$ and assign this value to each particle in Δ_j. So, small perturbations on the density lead to small perturbations in the weights. The advantage of this weighted restriction operator lies in the possibility to use the same realizations $\mathbf{x}$ in the unperturbated (6) and the perturbated (8) restriction-operator.

3 Convergence of variance-reduced Jacobian-vector-products

3.1 Why do we need variance reduction of the Jacobian-vector products?

If we want to compute steady states for the density $\boldsymbol{\rho}_*$ (without direct simulation), we can find them by solving the non-linear system

$$\boldsymbol{\rho}_* - \boldsymbol{\Phi}_T(\boldsymbol{\rho}_*) = 0. \tag{9}$$

In each Newton iteration, one needs to solve a linear system involving the Jacobian of $\boldsymbol{\Phi}_T$, denoted as $D(\boldsymbol{\Phi}_T)$. Since we do not have an explicit formula for $D(\boldsymbol{\Phi}_T)$ we are forced to use an iterative method (such as GMRES) that only requires Jacobian-vector products. The Jacobian $D(\boldsymbol{\Phi}_T)$ applied to a vector $\mathbf{v}$ (with unit norm) will be estimated by a finite difference approximation

$$\begin{aligned} D(\boldsymbol{\Phi}_T)\cdot\mathbf{v} &\approx \frac{\boldsymbol{\Phi}_T(\boldsymbol{\rho}+\varepsilon\mathbf{v},\boldsymbol{\omega_1}) - \boldsymbol{\Phi}_T(\boldsymbol{\rho},\boldsymbol{\omega_2})}{\varepsilon} \\ &\approx \frac{\boldsymbol{\Phi}_T(\boldsymbol{\rho},\boldsymbol{\omega_1}) + \varepsilon D(\boldsymbol{\Phi}_T)(\boldsymbol{\rho},\boldsymbol{\omega_1})\cdot\mathbf{v} - \boldsymbol{\Phi}_T(\boldsymbol{\rho},\boldsymbol{\omega_2})}{\varepsilon}. \end{aligned} \tag{10}$$

If we use the solution of the PDE, eq. (2), the time stepper is deterministic and the calculation of the Jacobian-vector products is straightforward. If we use the solutions of the SDE however, we have to deal with numerical noise in evaluating eq. 10. Because the coarse time-stepper is stochastic, repeating $\boldsymbol{\Phi}_T$ with two sets of random numbers $\boldsymbol{\omega_1}, \boldsymbol{\omega_2}$ will give different results. For $\varepsilon \ll 1$ this will result in an $\mathcal{O}(1/(\varepsilon^2 N))$ variance. Consequently the variance on the Jacobian-vector-products will grow unboundedly as ε tends to zero and $D(\boldsymbol{\Phi}_T)\mathbf{v}$ completely loses the structure of the perturbation $\mathbf{v}$.

3.2 How do we reduce the variance of the Jacobian-vector products?

This numerical noise can be reduced by using the same random numbers $\boldsymbol{\omega}$ for the unperturbed and perturbed simulations. If we apply the weighted restriction operator (8), we get the same microscopic realizations in the lifting step - the only difference is in the computation of the weights. As such, we impose $\boldsymbol{\omega_1} = \boldsymbol{\omega_2}$ in eq. (10) and consequently the variance of $D(\boldsymbol{\Phi}_T)\cdot\mathbf{v}$ is bounded and of $\mathcal{O}(1/N)$. Fig. 1 shows that the variance on the stochastic solution for the Jacobian-vector-product converges to zero with $\mathcal{O}(1/N)$ and that it does not depend on the value of ε.

3.3 Results

From now on, let us abbreviate the Jacobian-vector product $D(\boldsymbol{\Phi}_T)\cdot\mathbf{v}$ estimated from the for particle-based time-stepper as $\widehat{\mathbf{J}\mathbf{v}}$. We also chose the perturbation vector $\mathbf{v}$ to be sinusiodal (the convergence plots will be independent from $\mathbf{v}$). If we calculate $\widehat{\mathbf{J}\mathbf{v}}$ from the stochastic simulations with different number of particles N, we can test two things: the convergence to zero of the variance on $\widehat{\mathbf{J}\mathbf{v}}$ and the convergence to zero of the estimated bias on $\widehat{\mathbf{J}\mathbf{v}}$. We will also plot the mean squared error (MSE) which incorporates both

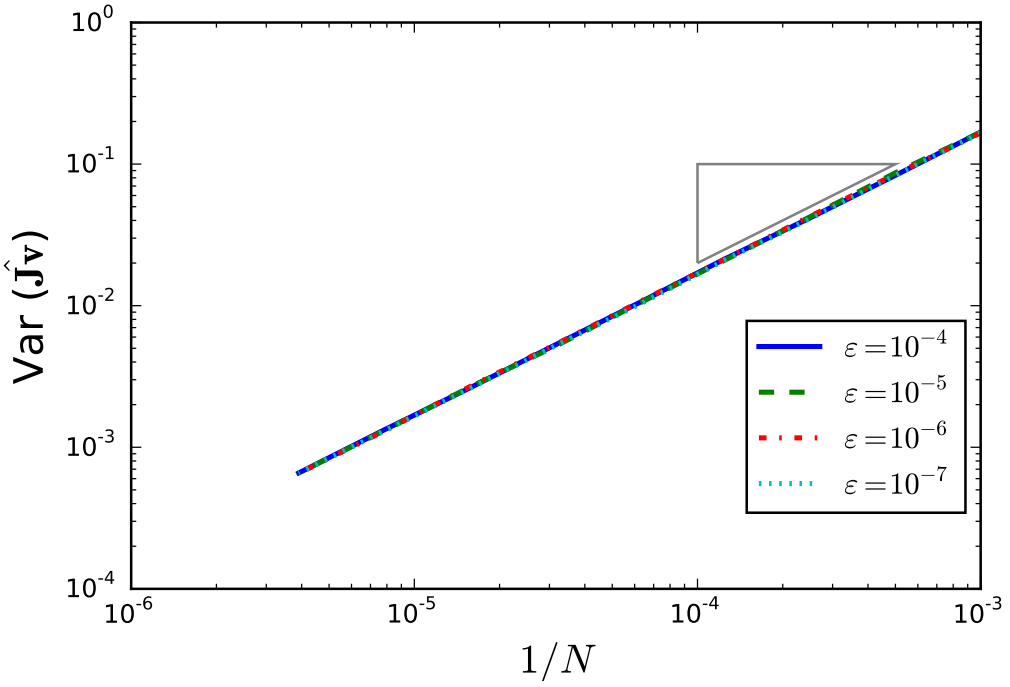

Figure 1: The variance on the stochastic solution for the Jacobian-vector-product converges to zero with $\mathcal{O}(1/N)$ and it does not depend on the value of ε.

effects (see eq. (11). The bias can be estimated by comparing the expectation value $\left\|\hat{\mathbf{J}}\mathbf{v}\right\|$ with the Jacobian-vector product for the PDE solved on a fine grid, denoted as $\mathbf{J}\mathbf{v}_{FP}$. The estimated expectation value $\mathbb{E}(\hat{\mathbf{J}}\mathbf{v})$ is the Jacobian-vector averaged over M stochastic simulations.

$$\texttt{MSE}(\hat{\mathbf{J}}\mathbf{v}, \mathbf{J}\mathbf{v}_{FP}) = \mathbb{E}\left[\left(\hat{\mathbf{J}}\mathbf{v} - \mathbf{J}\mathbf{v}_{FP}\right)^2\right] = \texttt{Var}(\hat{\mathbf{J}}\mathbf{v}) + \left(\mathbf{Bias}(\hat{\mathbf{J}}\mathbf{v}, \mathbf{J}\mathbf{v}_{FP})\right)^2 \tag{11}$$

with

$$\texttt{Var}(\hat{\mathbf{J}}\mathbf{v}) = \mathbb{E}\left[\left(\hat{\mathbf{J}}\mathbf{v} - \mathbb{E}[\hat{\mathbf{J}}\mathbf{v}]\right)^2\right] \tag{12}$$

and

$$\mathbf{Bias}(\hat{\mathbf{J}}\mathbf{v}, \mathbf{J}\mathbf{v}_{FP}) = \mathbb{E}[\hat{\mathbf{J}}\mathbf{v}] - \mathbf{J}\mathbf{v}_{FP}, \tag{13}$$

where we defined the inproduct of a n_x-dimensional vector $\mathbf{v}$ as $\mathbf{v}^2 = \frac{\mathbf{v}^T \cdot \mathbf{v}}{n_x}$. Thus, in eq. (11) and (12) we divide by the number of discretization steps n_x [2]. This allows us to make a meaningful comparison with solutions on finer or coarser grids.

[2]If the number of discretisations steps for solving the PDE happens to be higher than the number of bins n_x, we rescale the dimension of $\mathbf{J}\mathbf{v}_{FP}$ to n_x

Table 1: Parameter values

Discretization parameters		PDE	SDE
Discretization step	Δx	10^{-4}	10^{-2}
Number of discretization steps	n_x	34000	340
Time step	Δt	10^{-8}	10^{-4}
Number of timesteps	n	10^6	100

System parameters		
Diffusion coefficient	D	0.5
Drift coefficient	a	1
Simulation parameters		
Number of realizations	M	100
Perturbation size	ε	10^{-5}

4 Convergence of the bias

We calculate the expectation value of $\hat{\mathbf{J}}\mathbf{v}$ by substituting eq. (**??**) in the finite-difference approximation.

$$\mathbb{E}[\hat{\mathbf{J}}\mathbf{v}] = \mathbb{E}\left[\frac{\mathbf{\Phi}_T(\hat{\boldsymbol{\rho}} + \varepsilon\mathbf{v}) - \mathbf{\Phi}_T(\hat{\boldsymbol{\rho}})}{\varepsilon}\right] \tag{14}$$

with

$$\mathbf{\Phi}_T(\hat{\boldsymbol{\rho}}) = \frac{1}{N}\sum_{i=1}^{N} \chi_{\Delta_j}\left[\mathcal{E}(x_i(t)\right] \tag{15}$$

and

$$\mathbf{\Phi}_T(\hat{\boldsymbol{\rho}} + \varepsilon\mathbf{v}) = \frac{1}{N}\sum_{i=1}^{N}(1 + \frac{\varepsilon\mathbf{v}}{\boldsymbol{\rho}(x_i(t))}) \cdot \chi_{\Delta_j}\left[\mathcal{E}(x_i(t)\right]. \tag{16}$$

$$\begin{aligned}
\mathbb{E}[\hat{\mathbf{J}}\mathbf{v}] &= \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\frac{\chi_{\Delta_j}\left[\mathcal{E}(x_i(t))\right]\mathbf{v}}{\boldsymbol{\rho}(x_i(t))}\right] && (17)\\
&= \frac{\mathbf{v}}{N}\mathbb{E}\left[\sum_{i=1}^{N}\frac{\chi_{\Delta_j}\left[\mathcal{E}(x_i(t))\right]}{\boldsymbol{\rho}(x_i(t))}\right] && (18)\\
&= \frac{\mathbf{v}}{N}\sum_{i=1}^{N}\frac{\mathbb{E}\left[\chi_{\Delta_j}\left[\mathcal{E}(x_i(t))\right]\right]}{\boldsymbol{\rho}(x_i(t))} && (19)\\
&= \frac{\mathbf{v}}{N}\sum_{i=1}^{N}\frac{\chi_{\Delta_j}\left[\mathbb{E}\left[\mathcal{E}(x_i(t))\right]\right]}{\boldsymbol{\rho}(x_i(t))} && (20)
\end{aligned}$$

-

•

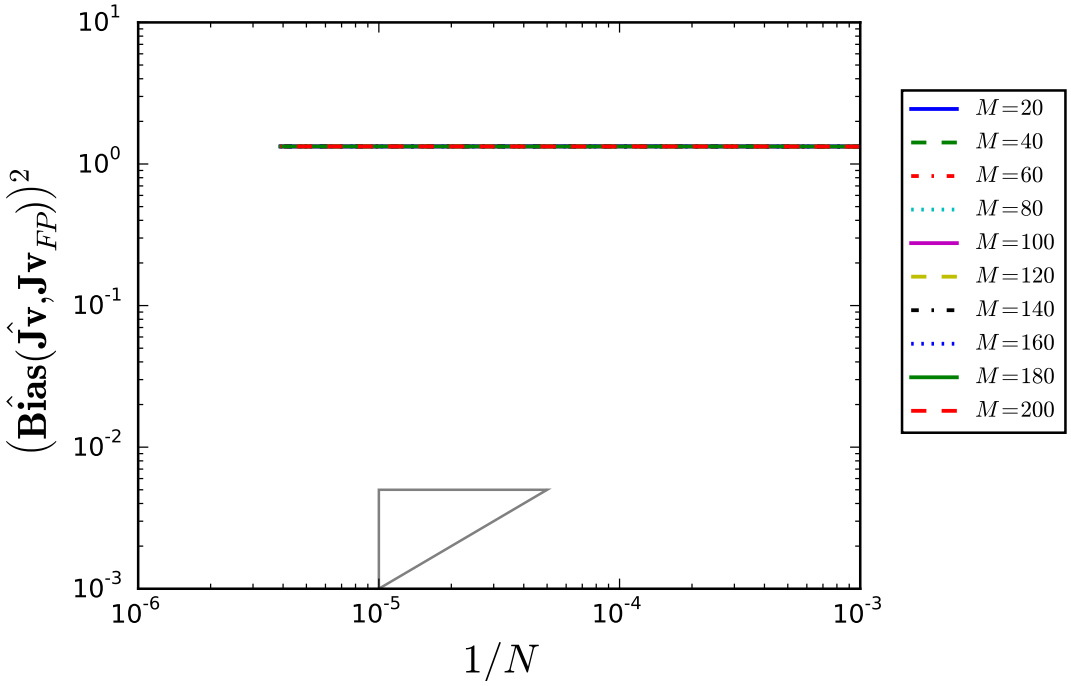

Figure 2: The expectation value of the stochastic solution of the Jacobian-vector-product is calculated by averaging over M stochastic realizations. The estimated bias between this solution and the deterministic one is squared and plotted as a function of the number of particles N for different values of M. For large particle numbers, the estimated bias becomes independent of M and converges to the real bias. This real bias is the numerical error in the solution of the PDE [Need to check this last statement].

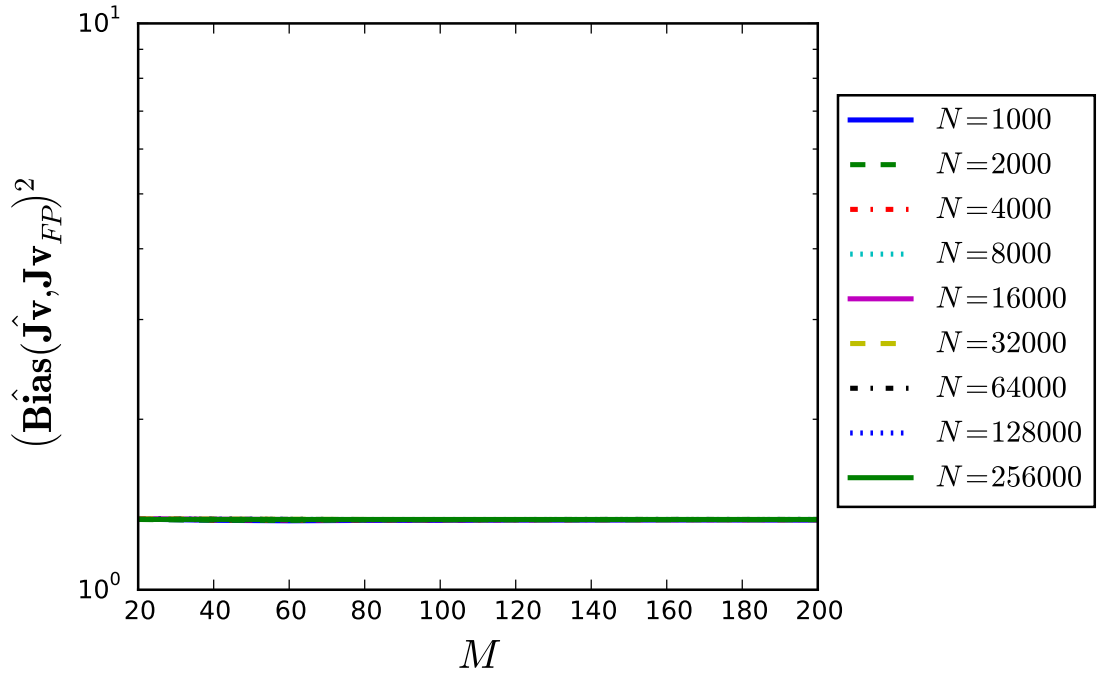

Figure 3: The expectation value of the stochastic solution of the Jacobian-vector-product is calculated by averaging over M stochastic realizations. The estimated bias between this solution and the deterministic one is squared and plotted as a function of the number of realizations M for simulations with different particle number N. For large particle numbers, the estimated bias becomes independent of M and converges to the real bias. This real bias is the numerical error in the solution of the PDE [Need to check this last statement].