

## Weighted-ensemble Brownian dynamics simulation: Sampling of rare events in nonequilibrium systems

Justus A. Kromer,<sup>1,\*</sup> Lutz Schimansky-Geier,<sup>1</sup> and Raul Toral<sup>2</sup>

<sup>1</sup>*Department of Physics, Humboldt-Universität zu Berlin, Newtonstr. 15, 12489 Berlin, Germany*

<sup>2</sup>*IFISC, Instituto de Física Interdisciplinar y Sistemas Complejos, CSIC-UIB, E-07122 Palma de Mallorca, Spain*

(Received 12 March 2013; revised manuscript received 17 June 2013; published 28 June 2013)

We provide an algorithm based on weighted-ensemble (WE) methods, to accurately sample systems at steady state. Applying our method to different one- and two-dimensional models, we succeed in calculating steady-state probabilities of order  $10^{-300}$  and reproduce the Arrhenius law for rates of order  $10^{-280}$ . Special attention is paid to the simulation of nonpotential systems where no detailed balance assumption exists. For this large class of stochastic systems, the stationary probability distribution density is often unknown and cannot be used as preknowledge during the simulation. We compare the algorithm's efficiency with standard Brownian dynamics simulations and the original WE method.

DOI: [10.1103/PhysRevE.87.063311](https://doi.org/10.1103/PhysRevE.87.063311)

PACS number(s): 07.05.Tp, 05.40.-a

### I. INTRODUCTION

Rare events are ubiquitous in many biological, chemical, and physical processes [1,2]. Whereas the density of states is known in systems at thermal equilibrium, interesting phenomena often occur in nonequilibrium systems [3]. Unfortunately, many such problems are inaccessible for analytic methods. Therefore computer simulations are a widely used tool to estimate the density of states or transition rates between them [4,5]. Since standard Brownian dynamics simulation (BDS) [6,7] provides computational costs that are inversely proportional to the state's probability, specialized methods [8–10] have to be used to adequately sample rare events, i.e., states with low probability or low transition rates.

In the last decades, flat histogram algorithms [11] have been developed, allowing one to evenly sample states with highly different probabilities. These algorithms are implementations of umbrella sampling [12], where each state is sampled according to a given probability distribution, the so-called umbrella distribution. Within nonequilibrium umbrella sampling [13] the space of interest is divided into different but almost evenly sampled subregions. The interaction between different regions occurs solely due to probability currents between them, whereby the probability distribution within a region is then calculated by performing Monte Carlo simulations.

In order to calculate low rates between a starting and a final state, forward flux sampling methods can be used (for a review see [14]). These methods consider a sequence of surfaces between these states and introduce walkers (copies of the system) to perform weighted trajectories according to the underlying dynamics. If walkers cross one of the surfaces, getting closer to the final state, new walkers with lower weights are introduced. Finally, many walkers with particular low weights reach the final state. Consideration of the particular weights allows one to calculate very low rates in a finite simulation time. Recently, extensions to these methods have been developed to calculate both transition rates using umbrella sampling [15] and probability distributions using forward flux sampling [16] algorithms.

In this work, we present an algorithm, based on the previously developed weighted-ensemble (WE) Brownian dynamics simulation [17–20], which allows one to calculate the stationary probability density function (SPDF) as well as transition rates between particular states. As in WE simulations the space of interest is divided into several subregions and the probability of finding the system in them is calculated by generating equally weighted walkers in each region. By moving to the underlying dynamics, the walkers transport probability between the subregions. Thus, WE methods are usually applied to systems of Brownian particles moving in a potential landscape [18,21].

We are interested in an algorithm which allows simulations of stochastic dynamical systems that *a priori* do not obey detailed balance for probability fluxes or suppose some special topology of the flow [22–26]. As a test bed, however, we use two classes of systems for which the stationary PDF can be computed analytically. The first is that of Brownian particles in conservative force fields under the influence of additive noise. The second is the class of canonic dissipative systems (see Appendix A) which are able to enhance self-oscillatory oscillations. If transformed to the energy variable and phase, the corresponding probability flow in these variables vanishes [24,27,28]. Beyond these two simple examples, our goal is to develop an algorithm which does not assume that neither the deterministic nor the stochastic items [see Eq. (1)] underlie such conditions. Thus, no information on the SPDF can be used for the simulations.

In general the algorithm can be applied to arbitrary dynamical systems of the form

$$\dot{x}_n = f_n(\mathbf{x}) + g_n(\mathbf{x})\xi_n(t), \quad n = 1, \dots, d, \quad (1)$$

where  $d$  is the number of stochastic time-dependent degrees of freedom  $x_n(t)$ ; the functions  $f_n(\mathbf{x})$  describe the deterministic velocities for the  $n$ th direction; and  $\xi_n(t)$  represents zero-mean Gaussian noise with  $\delta$ -like correlation function  $\langle \xi_n(t)\xi_m(t') \rangle = \delta_{nm}\delta(t-t')$ . The noise intensity along the  $n$ th direction is scaled by the functions  $g_n(\mathbf{x})$ , which in general depend on the vector  $\mathbf{x} = (x_1, x_2, \dots, x_d)^T$ . We are interested in high-precision sampling of the stationary probability current  $J_{\text{st}}(\mathbf{x})$  and the SPDF  $p_{\text{st}}(\mathbf{x})$  of finding the system in the

\*Electronic address: justuskr@physik.hu-berlin.de

$d$ -dimensional cube  $[x_1, x_1 + dx_1], \dots, [x_d, x_d + dx_d]$  with a finite resolution. We specify the resolution by the number  $M_{n,\text{res}}$  of evenly spaced supporting points along the  $n$ th direction, for which we determine  $p_{\text{st}}(\mathbf{x})$ .

This article is organized as follows: In Sec. II we introduce the new algorithm, based on WE methods [17], that allows one to calculate low probabilities and rates. In Sec. III, we study one- and two-dimensional model systems and analyze the algorithm's efficiency compared to BDS and WE techniques. Finally, we discuss the results (Sec. IV) and present a short conclusion (Sec. V).

## II. THE ALGORITHM

First, for the sake of clarity, we restrict ourselves here to particle motion in one dimension,  $d = 1$ , under additive noise. The deterministic part of the dynamics, Eq. (1), can always be represented as a conservative force,  $f(x) = -U'(x)$ . The noise strength is scaled by the parameter  $D$ ; we put  $g(x) = \sqrt{2D}$ . For such systems, the SPDF is known to be

$$p_{\text{st}}(x) = Z_{\text{st}}^{-1} e^{-\frac{U(x)}{D}}, \quad (2)$$

where  $Z_{\text{st}}$  is a normalization constant.

We are interested in finding numerically the system's SPDF,  $p_{\text{st}}(X_j)$ , at a set of  $M_{\text{res}}$  evenly spaced supporting points  $X_j$  in a finite part of the physical space, given by  $x \in [L^-, L^+]$ . The region of interest is divided into  $M \geq M_{\text{res}}$  subregions of size  $\Delta x = \frac{L^+ - L^-}{M}$ , the  $i$ th subregion is bounded by  $(x_i, x_{i+1})$ ,  $i = 0, \dots, M - 1$ , with  $x_i = i\Delta x + L^-$ . Supporting points are given explicitly by  $X_j = L^- + (j - \frac{1}{2})\Delta x$ ,  $j = 1, 2, \dots, M_{\text{res}}$ , with  $\Delta x_{\text{res}} = \lceil \frac{M}{M_{\text{res}}} \rceil \Delta x$  (see Fig. 1). Here  $\lceil z \rceil$  denotes the largest integer less than or equal to  $z$ .

Let us introduce the probability  $P_i(t)$  of finding a particle in the  $i$ th subregion at time  $t$ , and the corresponding set  $\mathbf{P}(t) = (P_0(t), P_1(t), \dots, P_{M-1}(t))$ . Initially, no information on the system is available, thus, each subregion is given an arbitrary amount of probability  $P_i(0)$ , simply fulfilling the normalization condition  $\sum_{i=0}^{M-1} P_i(0) = 1$ . Naturally, equilibration can be accelerated if one already has information on the SPDF of the system [Eq. (1)]. In that case, one can choose  $\mathbf{P}(0)$  close to the set  $\mathbf{P}_{\text{st}}$ , which optimally approximates the SPDF:

$$P_{\text{st},i} = \int_{x_i}^{x_{i+1}} dx p_{\text{st}}(x). \quad (3)$$

However, in general no such information is required.

### A. Time evolution

After setting the initial set  $\mathbf{P}(t = 0)$ , the time evolution of the  $\mathbf{P}(t) \rightarrow \mathbf{P}(t + h)$  is performed using three steps. We start with the *redistribution step*, in which  $N$  walkers (copies of the

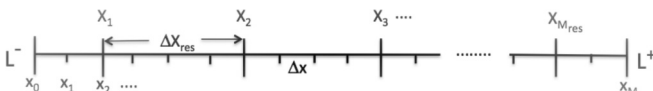


FIG. 1. Scheme of the interpolation points  $x_i$  and the supporting points  $X_j$  used in the numerical calculations. In this scheme we used  $\lceil \frac{M}{M_{\text{res}}} \rceil = 4$ .

system) are uniformly distributed in each subregion. Besides their individual positions  $x_i^k(t)$ , where  $i = 0, \dots, M - 1$  denotes the particular subregion and  $k = 1, \dots, N$  the individual walkers, each walker possesses a given amount of weight  $q_i^k(t)$ . This is nothing but the present probability in the  $i$ th subregion distributed to the  $N$  walkers, which yields

$$q_i^k(t) = \frac{P_i(t)}{N}. \quad (4)$$

Note that one does not need to introduce walkers in subregions with  $P_i(t) = 0$ .

After the redistribution step has been performed, Eq. (1) is integrated for all walkers, using a Brownian dynamic simulation step  $h$  and an arbitrary integration scheme. This *integration step* realizes the time evolution  $x_i^k(t) \rightarrow x_i^k(t + h)$ . Here walkers transport probability between the subregions. As walkers are independent of each other, it is important to note that the particular time evolution of each one of the  $N \times M$  walkers is due to different sample paths in the stochastic parts of the Langevin equation.

Finally, the *updating step* is performed, in which the new probabilities  $P_i(t) \rightarrow P_i(t + h)$  are calculated by summing up the weights of all walkers that are currently located in the particular subregion,

$$P_i(t + h) = \sum_{i', k | x_i^k(t+h) \in (x_i, x_{i+1})} q_{i'}^k(t). \quad (5)$$

In what follows, we name the sequence of redistribution, integration, and updating steps as *running step*. After an equilibration time  $T_{\text{therm}}$ , the set  $\mathbf{P}(t)$  reaches a stationary regime, where the  $P_i(t)$ 's fluctuate around their mean values  $\langle P_i \rangle$ .

### B. Calculating the stationary probability

The individual  $\langle P_i \rangle$  are estimated by averaging over a total amount of  $N_T$  sets  $\mathbf{P}(t_\ell)$ ,  $\ell = 1, 2, \dots, N_T$ , taken, after the system has reached the stationary regime, every  $n_{\text{av}}$  running steps:  $t_\ell = T_{\text{therm}} + (\ell - 1)n_{\text{av}}h$ . It turns out that the mean probabilities  $\langle P_i \rangle$  coincide with the stationary probability  $P_{\text{st},i}$  [see Eq. (3)], for compatibly chosen time step  $h$  and the size of a subregion  $\Delta x$  (see Sec. II E).

Finally, the SPDF on the supporting points  $p_{\text{st}}(X_j)$  is calculated by adding the adjacent  $\langle P_i \rangle$  and dividing by the size  $\Delta x_{\text{res}}$  (in order to have a properly normalized PDF):

$$p_{\text{st}}(X_j) = \frac{1}{\Delta x_{\text{res}}} \sum_{i=(j-1)\lceil \frac{M}{M_{\text{res}}} \rceil}^{j\lceil \frac{M}{M_{\text{res}}} \rceil - 1} \langle P_i \rangle. \quad (6)$$

### C. Calculation of the probability current

The stationary probability current  $J_{\text{st}}(x)$  at position  $x$  can be easily calculated by adding up (with the right sign) the weights of all walkers, passing  $x$  to the right and to the left per unit time. In practice,  $x$  should be the boundary of a subregion. If  $x = x_i$ , the current  $J(x_i, t)$  is given by

$$J(x_i, t) = \frac{1}{h} \left( \sum_{i', k \in R_i} q_{i'}^k(t) - \sum_{i', k \in L_i} q_{i'}^k(t) \right), \quad (7)$$

and  $R_i$  indicate these walkers which cross the boundary moving rightwards, i.e.,  $x_i^k(t) > x_i \wedge x_i^k(t-h) < x_i$ . Alternatively,  $L_i$  assign walkers transporting weight leftwards,  $x_i^k(t) < x_i \wedge x_i^k(t-h) > x_i$ .

Averaging over  $N_T$  such estimates, taken in the stationary regime, leads to the average current  $\langle J(x_i) \rangle$ , which converges towards  $J_{\text{st}}(x_i)$  for  $N_T \rightarrow \infty$ .

#### D. Implementation of boundary conditions

The implementation of boundary conditions for the probability current or the SPDF is straightforward. Right now, absorbing boundaries are already implemented at  $L^-$  and  $L^+$ , since walkers that pass these boundaries are not located in any subregion. Therefore, their weights would get lost in the next updating step. To avoid this, reflecting boundary conditions at  $L^+$  can be implemented by setting  $x_i^k(t+h) \rightarrow 2L^+ - x_i^k(t+h)$  for all walkers with  $x_i^k(t+h) > L^+$ . Hence, the probability current at  $L^+$  will be 0. Reflecting boundaries at  $L^-$  can be implemented analogously.

#### E. Convergence criteria

##### 1. One-dimensional systems

In order to ensure that  $\langle P_i \rangle$  and  $\langle J(x_i) \rangle$  converge towards the stationary probability distribution and the probability current of Eq. (1) for  $N_T \rightarrow \infty$ , the time step  $h$  and the size of a subregion  $\Delta x$  have to fulfill specific criteria. This is due to the redistribution step, where walkers are uniformly distributed in each subregion. This implicates statistical errors, since they can reach positions in a subregion that are inaccessible or, at least, more improbable. Therefore, walkers can more easily escape from potential minimums or reach regions of low probability. This effectively flattens the probability distribution, leading to more probability in regions of low probability, for instance, around local maxima of  $U(x)$ , and less probability in the potential minima. In order to overcome this problem, earlier works [17,18] have stored the positions and weights of all walkers. In the next redistribution step, walkers were only spaced on the stored positions according to the weights belonging to them. This requires a lot of computer memory, especially for large  $N$  and  $M$ .

However, we found that one does not need to store these information, if the subregions are small enough to ensure that walkers have a non-negligible probability of leaving them during one integration step. We first consider the case of additive noise  $g(x) = \sqrt{2D}$ . As a measure of how far a walker can step, due to the fluctuations, in one time step, we use the diffusion length  $L_{\text{dif}} = \sqrt{2Dh}$ . Thus, the size of a subregion  $\Delta x$  should be small compared to the diffusion length  $L_{\text{dif}}$ :

$$\Delta x \ll \sqrt{2Dh}. \quad (8)$$

The distance a walker can pass during an integration step is determined not only by  $L_{\text{dif}}$ , but also by the deterministic dynamics, leading to a step length  $L_{\text{det}} = f(x)h$  at first order. Usually  $f(x)$  changes very rapidly at the boundaries of the simulation area, which produces high deterministic velocities and regions of low probability. Walkers can only reach these regions, if the fluctuation are strong enough to balance the

deterministic force, i.e., if

$$|f(x)|h < 2\sqrt{Dh} \quad (9)$$

for all  $x \in [L^-, L^+]$ . This leads to a condition for the time step  $h$ :

$$h < h_{\text{max}} := \frac{4D}{\max_{x \in [L^-, L^+]} f^2(x)}. \quad (10)$$

Hence, lower time steps allow one to sample regions far from the potential extrema and, for instance, the tails of the SPDF. If a larger time step is chosen, the  $\langle P_i \rangle$  will run to 0 in subregions with a higher deterministic force. Since it is often difficult to fulfill Eq. (10) in the entire simulation area, one should choose a time step which allows one to fulfill Eq. (8).

##### 2. Multidimensional systems

In general, our method can be applied to stochastic dynamical systems in arbitrary dimension  $d \geq 1$  [Eq. (1)]. For the foundation of the algorithm, we refer to Appendix C. However, in order to ensure convergence in a finite region  $A$ , the criteria for the time step  $h$  and the size of a subregion  $\Delta x_n$  should be fulfilled along any direction. The condition for the time step [Eq. (10)] then becomes

$$h \approx \min_{1 \leq n \leq d} \left( \frac{4D_n}{\max_A (f_n(\mathbf{x}))^2} \right), \quad (11)$$

where  $D_n$  is the noise intensity along the  $n$  direction, i.e.,  $g_n(\mathbf{x}) = \sqrt{2D_n}$ . The criterion for the size of a subregion along the  $n$ th directions [Eq. (8)] reads

$$\Delta x_n \ll \sqrt{2D_n h}. \quad (12)$$

However, there might be directions without any noise ( $D_n = 0$ ). In that case the length of a subregion should ensure that walkers can leave it due to the deterministic term  $f_n(\mathbf{x})$ , leading to

$$\Delta x_n \leq f_n(\mathbf{x})h. \quad (13)$$

Otherwise, information on the deterministic dynamics gets lost during the redistribution step, since walkers that stay in a subregion do not produce any change in the  $P_i$  and are again randomly placed in their subregion during the next redistribution step.

If, for simplicity, we want to keep equally sized subregions, then the common value  $\Delta x_n$  should be equal to the minimum value of Eq. (12) or Eq. (13) with respect to all  $\mathbf{x}$  in the simulation area for each of the  $d$  directions. This simple criterion can lead to a huge number of subregions, especially in high-dimensional spaces.

To appropriately reduce the number of subregions, we can chose the  $\Delta x_n$  to fulfill the local criteria along direction  $n$ . This was implemented by a *grouping algorithm*, which groups original subregions into larger ones as long as walkers can leave these due to the deterministic term [Eq. (13)]. Depending on the system, this procedure greatly reduces the total number of larger subregions  $M_{\text{group}}$ . If the grouping algorithm is used,  $N$  walkers are randomly placed in each of these larger subregions and a probability  $P_i$  of finding a walker in the corresponding area is introduced. We find that such a grouping greatly reduces the computational costs, since fewer subregions and therefore fewer walkers are required. Since

walkers jump out of these regions until the next redistribution step starts, the algorithm still approximates the correct SPDF.

### 3. Multiplicative noise

Since the criteria were derived for additive noise, we briefly discuss the more general case of position-dependent noise strength  $g_n(\mathbf{x})$ , which is assumed to be nonzero in the simulation area. After a corresponding interpretation, i.e., Ito [29,30], Stratonovich [31,32], and others is chosen, the deterministic part,  $f_n(\mathbf{x})$ , in the inequalities, Eq. (12) or Eq. (13) for  $\Delta x$  and Eq. (11) for the time step, should be accommodated to  $f_n(\mathbf{x}) + q \sum_{m=1}^d \partial_{x_m} g_n(\mathbf{x}) g_m(\mathbf{x})$ , where  $q = 0$  in the Ito case and  $q = \frac{1}{2}$  in the Stratonovich case, respectively. So, for the Stratonovich interpretation, we must include the new deterministic part and  $D \rightarrow \frac{g(\mathbf{x})^2}{2}$  in the inequalities, providing the values of the parameters  $h$  and  $\Delta x$ . In order to minimize the computational effort, the subregions should be scaled to locally fulfill these inequalities. After the subregions have been chosen, the integration steps can be performed. Note that the choice of a integration scheme can already correspond to a certain interpretation [7].

### F. Simulation techniques

Simulations were performed on an Intel Xeon CPU E31245 @ 3.30 GHz processor with 16 Gb DDR-3 RAM. The algorithm described above was implemented in a C++ program for one- and two-dimensional systems. Runs of the algorithm are specified by the time step  $h$ , the size of a subregion  $\Delta x$  ( $\Delta y$  in two-dimensional problems), the simulation area, given by  $L^-$  and  $L^+$  ( $L_x^\pm, L_y^\pm$ ), the number of walkers per subregion  $N$ , the thermalization time  $T_{\text{therm}}$ , and the number of running steps between two sets of  $\mathbf{P}$  denoted  $n_{\text{av}}$ . The numerical integration of the Langevin equation, (1), was done using a Heun scheme [7,33,34]. A resolution of  $M_{\text{res}} = 200$  was used in any direction.

To compare the results with other methods, we also perform BDS using  $N_{\text{Brown}}$  initially uniformly placed particles in the simulation area. Integration was done using the Heun scheme with integration time step  $h_{\text{Brown}}$ . After a thermalization time  $T_{\text{therm,Brown}}$  the particles' positions were recorded after time intervals  $\Delta t_{\text{Brown}}$ . The BDS was given a running time  $T_{\text{run}}$  (real CPU time) which usually equals the time our algorithm needs to produce its results. After  $T_{\text{run}}$  the BDS was stopped and the SPDF was calculated using the recorded particle positions. We set  $T_{\text{therm,Brown}} = T_{\text{therm}}$ ,  $h_{\text{Brown}} = h$ , and  $\Delta t_{\text{Brown}} = n_{\text{av}} h$  to make the results comparable.

Additionally, simulations of the WE method [17,18] were performed using 200 subregions per direction (same as  $M_{\text{res}}$  for our algorithm). To allow comparability, the integration time step and integration scheme were chosen similar to those of the particular runs of our algorithm. The time evolution was performed similarly to our method, except that particle positions and weights were stored and used in the redistribution step. Here  $N$  particles per subregion were distributed to stored positions according to the positions' weights.

## III. MODEL SYSTEMS AND RESULTS

### A. One-dimensional system

In order to demonstrate the implementation of the algorithm, we study overdamped Brownian motion in a bistable potential  $U(x) = -\frac{x^2}{2} + \frac{x^4}{4}$ . Correspondingly, we put  $f(x) = x - x^3$  and  $g(x) = \sqrt{2D}$  in Eq. (1), which results in a bistable system which is often used to study bistable systems or stochastic resonance therein [25,35]. The two stable states come up to the potentials' minima, located at  $x = -1$  and  $x = 1$ , respectively. The corresponding SPDF is given by Eq. (2). For a low noise strength, the SPDF attains sharp peaks at the potential minima and decreases down to low values at the borders and the local maximum, for instance, for  $D = 0.01$   $p_{\text{st}}(0) \approx 3.887 \times 10^{-11}$ , while  $p_{\text{st}}(\pm 1) \approx 2.79895$ .

#### 1. Equilibration

At first, we study the equilibration process, performing simulations with different numbers of walkers per subregion  $N$ . Results are shown in Fig. 2. Analyzing the time dependence of the probability  $P_i(t)$ , we find that the longest thermalization time occurs at the local maximum of  $U(x)$ . Note that runs with larger  $N$  thermalize at lower  $t$ , but one needs more integration steps.

After thermalization has been achieved, we evaluate the coefficient of variation of the probability, given by

$$c(x_i) = \frac{\sqrt{\langle (P_i - \langle P_i \rangle)^2 \rangle}}{\langle P_i \rangle}, \quad (14)$$

where averages  $\langle \dots \rangle$  are performed for a fixed number  $N_T$  of sets  $\mathbf{P}$  and different  $N$  and  $x_i$ . We find it to scale according to  $\frac{1}{\sqrt{N}}$  (data not shown).

#### 2. Stationary probability density function

We start by calculating the SPDF  $p_{\text{st}}$  in the region  $[L^-, L^+]$  for a low noise strength ( $D = 0.01$ ). The values for the time step  $h$  and the box size  $\Delta x$  are summarized in Table I and fulfill the inequalities (10) and (8) sufficiently. Using the results shown in Fig. 2, we set the thermalization time

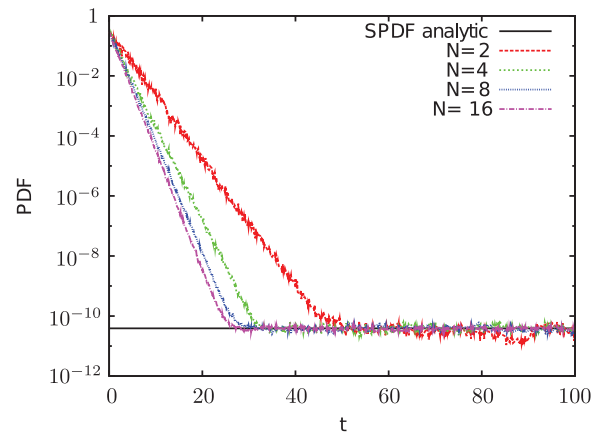


FIG. 2. (Color online) PDF at  $x = 0$  obtained from  $P_i$  by  $\text{PDF}(t) = \frac{P_i(t)}{\Delta x}$  for a subregion containing  $x = 0$  for  $D = 0.01$  and different numbers of walkers per subregion  $N$ . Parameters are chosen as in run 1 (see Table I).

TABLE I. Time step  $h$  and box size  $\Delta x$  according to convergence criteria Eqs. (10) and (8), where we choose  $h = \frac{h_{\max}}{2}$  and  $\Delta x = \frac{1}{20} L_{\text{dif}}$ .

Run	$L^-$	$L^+$	$h$	$\Delta x$	$M$
1	-1.4	1.4	0.011	0.001 048 69	2 670
2	-1.75	1.75	0.001 5	0.000 387 297	9 037
3	-2.5	2.5	0.000 1	0.000 107 75	46 404

$T_{\text{therm}} = 50$  for a run with  $N = 2$ . Time averages after thermalization were calculated over an ensemble of  $N_T = 10^4$  sets  $\mathbf{P}$ . Results for the SPDF are shown in Fig. 3. The algorithm calculates the tails of the distribution down to  $10^{-300}$  correctly, after a running time  $T_{\text{run}} \approx 27$  h. We also calculate the SPDF using BDS using  $N_{\text{Brown}} = 10^4$ , which stops estimating at a level of  $10^{-6}$  after the same running time. Further runs were performed (see Table I, runs 2 and 3), approximating the tails down to  $10^{-10}$  ( $M_{\text{group}} = 1136$ ) and  $10^{-48}$  ( $M_{\text{group}} = 6789$ ) after a running time of  $\approx 30$  s and  $\approx 10$  min, respectively (data not shown).

Simulations for different values of  $\frac{\Delta x}{L_{\text{dif}}}$  indicate that insignificant deviations from the analytic SPDF occur for  $\Delta x > \frac{1}{20}$ .

### 3. Probability current

Next, we present that our algorithm can be used to calculate the escape rate to pass the energy barrier at  $x_{\max} = 0$ . Such problems are typical for chemical reactions [36] and in the field of neuroscience [37].

Initially, only  $N$  particles are assigned at the subregion including  $x_{\min} = 1$ , so approximating an initial  $\delta$ -like probability distribution for  $t = 0$ . Furthermore, an absorbing boundary right behind the local maximum ( $x_{\text{abs}} = -0.01$ ) is included. To fulfill normalization of the SPDF, walkers that reach  $x_{\text{abs}}$  are reinjected immediately at  $x_{\min}$ . The escape rate to pass the barrier is given by the probability current  $J(x_{\max})$ . For low noise intensities, the probability current on top of the potential

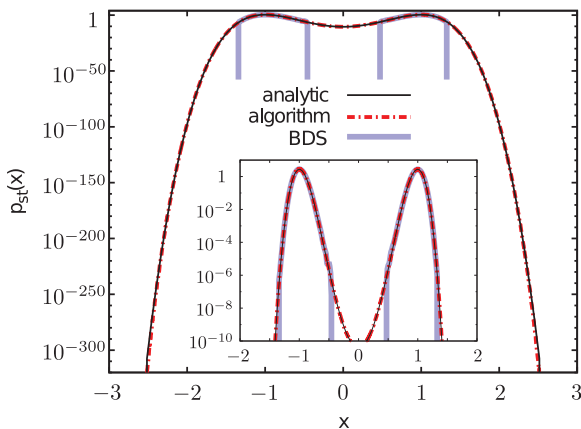


FIG. 3. (Color online) Estimates for the stationary probability density obtained from the algorithm for run 3 (see Table I) and by using a Brownian dynamics simulation for  $D = 0.01$  with  $N = 2$  walkers per subregion. Inset: The region of probabilities higher than  $10^{-10}$ . Analytic results were obtained from Eq. (2).

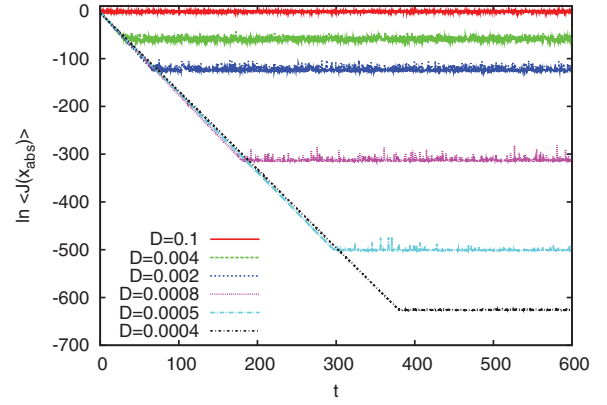


FIG. 4. (Color online) Time dependence of the probability current  $J(x_{\text{abs}})$  obtained from our algorithm for  $L^- = -0.02$ ,  $L^+ = 1.39$ , and decreasing noise intensities (from top to bottom). The time step is set at  $h = \frac{h_{\max}}{2}$  and  $\Delta x = \frac{1}{20} L_{\text{dif}}$ . Note that  $h_{\max}$  and  $L_{\text{dif}}$  vary according to Eqs. (10) and (8), respectively, resulting in longer running times for smaller  $D$ .

barrier can be described using the Arrhenius law, namely,

$$J(x_{\max}) \propto e^{-\frac{\Delta U}{D}}, \quad (15)$$

where  $\Delta U = U(x_{\max}) - U(x_{\min}) = 0.25$ . Since strong fluctuations are rare, but possible, we use  $J(x_{\text{abs}})$  to approximate  $J(x_{\max})$ . Probability currents  $J(x_{\text{abs}})$  were recorded for each time step and averaged over a sequence of  $n_{\text{av}} = \lceil \frac{0.1}{h} \rceil$  running steps, resulting in  $\langle J(x_{\text{abs}}) \rangle$ . Figure 4 shows the time dependence of  $\ln \langle J(x_{\text{abs}}) \rangle$ . After a relaxation regime, where the current decays exponentially, the current reaches its stationary value. The values of  $\ln \langle J(x_{\text{abs}}) \rangle$ , averaged over the stationary regime, are shown in Fig. 5 for different noise intensities. Fulfilling the criteria described above, the algorithm reproduces the Arrhenius law well down to  $\ln \langle J(x_{\text{abs}}) \rangle \approx -650$ , corresponding to a current  $J(x_{\max}) \approx 10^{-286}$ .

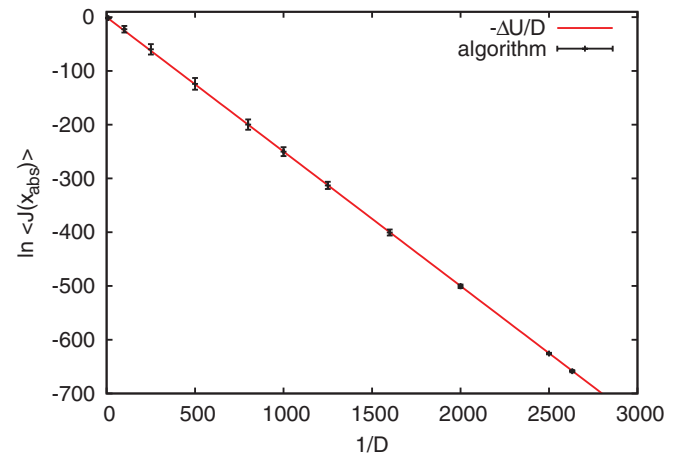


FIG. 5. (Color online) Stationary probability current as a function of the inverse noise strength obtained by time averaging the data partly shown in Fig. 4 in the stationary regime. Error bars show three standard deviations of the stationary data.

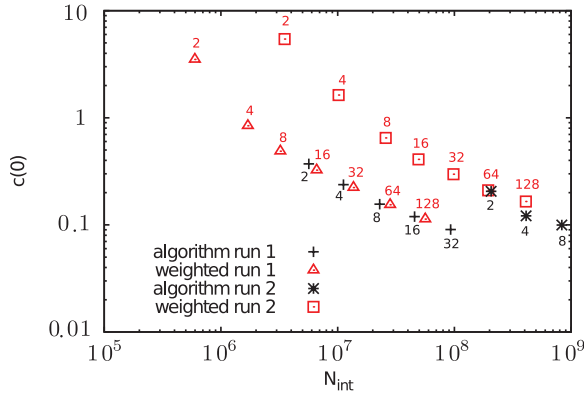


FIG. 6. (Color online) Relative fluctuations plotted over the number of integration steps, needed for equilibration, for the algorithm (black) and standard (WE) simulation (red). The number of walkers per subregion is shown for each point. Simulations were done for run 1 and 2 (compare Table I).

#### 4. Efficiency compared to that of weighted-ensemble Brownian dynamics simulation

In order to compare the efficiency of two algorithms, estimating the SPDF of a system given by Eq. (1), important quantities are the transient time required to first reach the steady state. Once the algorithm reaches the steady state, we quantify the size of the fluctuations by the coefficient of variation, Eq. (14), at the potential's local maximum  $x = 0$ . The computation time mainly depends on the number of integrations  $N_{\text{int}}$  needed to reach the stationary regime. In order to compare the efficiency, the size of fluctuations [Eq. (14)] in the local minimum relative to  $p_{\text{st}}(0) \approx 3.88717 \times 10^{-11}$  during the stationary regime is plotted over  $N_{\text{int}}$  in Fig. 6. The most effective algorithm would be located close to the origin. Comparing the efficiencies of the WE method [17,18] and our algorithm, we find that WE simulations with low  $N$  equilibrate more rapidly. However, the precision greatly depends on the fluctuations during the stationary regime. To produce results of the same precision [same  $c(0)$ ] both algorithms need approximately the same  $N_{\text{int}}$ . Usually WE simulations were performed using thousands of walkers per subregion, resulting in a low value of  $c(0)$ . Here runs of our algorithm producing the same  $c(0)$  need a much lower  $N$  and have memory requirements independent of  $N$ .

## B. Two-dimensional systems

### 1. Poincaré oscillator

As an example of a two-dimensional system with a known SPDF, we consider the Poincaré oscillator [27,38,39], represented by the dynamical system

$$\dot{x} = y, \quad \dot{y} = (\alpha - x^2 - y^2)y - x + \sqrt{2D}\xi(t), \quad (16)$$

where  $\xi(t)$  represent  $\delta$  correlated white Gaussian noise with zero mean. Using the energy function  $H(x, y) = \frac{1}{2}(x^2 + y^2)$ , which depends only on the distance to the origin, one can calculate the associated SPDF:

$$p_{\text{st}}(x, y) = Z_{\text{st}}^{-1} \exp\left(\frac{\alpha H(x, y) - H^2(x, y)}{D}\right), \quad (17)$$

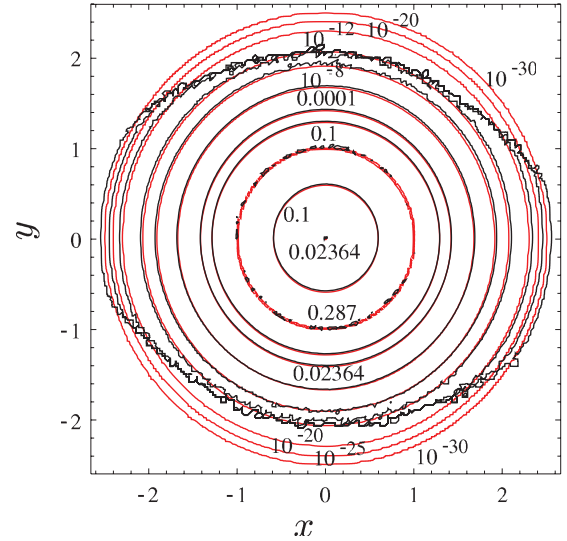


FIG. 7. (Color online) Contour plots of the SPDF ( $D = 0.1$  and  $\alpha = 1$ ) obtained from the algorithm (black), using the parameters  $L_x^- = L_y^- = -3$ ,  $L_x^+ = L_y^+ = 3$ ,  $T_{\text{therm}} = 8$ ,  $N_T = 10^3$ ,  $h = 0.01$ ,  $M_x = 189737$ ,  $M_y = 1897$ ,  $M_{\text{group}} = 1688940$ , and  $N = 2$  ( $\Delta y = \frac{1}{20}L_{\text{dif}}$ ), and the analytic solution (red), Eq. (17). Contour lines are labeled according to represented values of the SPDF and show the rotational symmetry. The SPDF possesses its global maximum at  $H(x, y) = \frac{\alpha}{2}$ , corresponding to the unit circle for our choice of  $\alpha$ , and a local minimum in the origin. Running time  $\approx 5$  h.

(see Appendix A). Since noise applies only to the  $y$  direction, the lengths of a subregion  $\Delta x$  and  $\Delta y$  in the  $x$  and  $y$  directions are calculated by Eqs. (13) and (8), respectively. The minimum of  $|f_x(x, y)| = |y|$  is equal to 0, therefore, we choose  $\Delta x = \Delta y h$ , which corresponds to a first-order approximation of  $|f_x(x, y)|$  in the next subregion. Results for the SPDF are shown in Fig. 7. Interestingly, the algorithm approximates better the SPDF along the direction where no noise was applied. Here the SPDF is sampled down to  $10^{-30}$ . We found that the algorithm slightly oversamples the analytic SPDF in the tails. This is due to the statistical errors made during the redistribution step. By reducing the size of a subregion, this error can be reduced further. Along the  $y$  direction, noise is applied. Here the behavior is similar to the one-dimensional example (see above). For runs with larger  $\Delta y$  (results not shown) the algorithm slightly oversamples the SPDF in the origin.

### 2. Bistable system with colored noise

As a further example, we calculate the SPDF of the two-dimensional system

$$\dot{x} = x - x^3 + y, \quad \dot{y} = -\frac{1}{\tau}y + \frac{1}{\tau}\sqrt{2D}\xi(t), \quad (18)$$

where  $\tau$  denotes the time scale separation between  $x$  and the colored noise  $y$ . The white Gaussian noise  $\xi(t)$  has already been described above. This system has been studied in [40] and [41]. As in the bistable system we have studied above, the SPDF has maxima at  $(x, y) = (1, 0)$  and  $(x, y) = (-1, 0)$ . However, for some combinations of  $\tau$  and  $D$ , the SPDF possesses a local minimum at  $(x, y) = (0, 0)$ . Plots of the SPDF

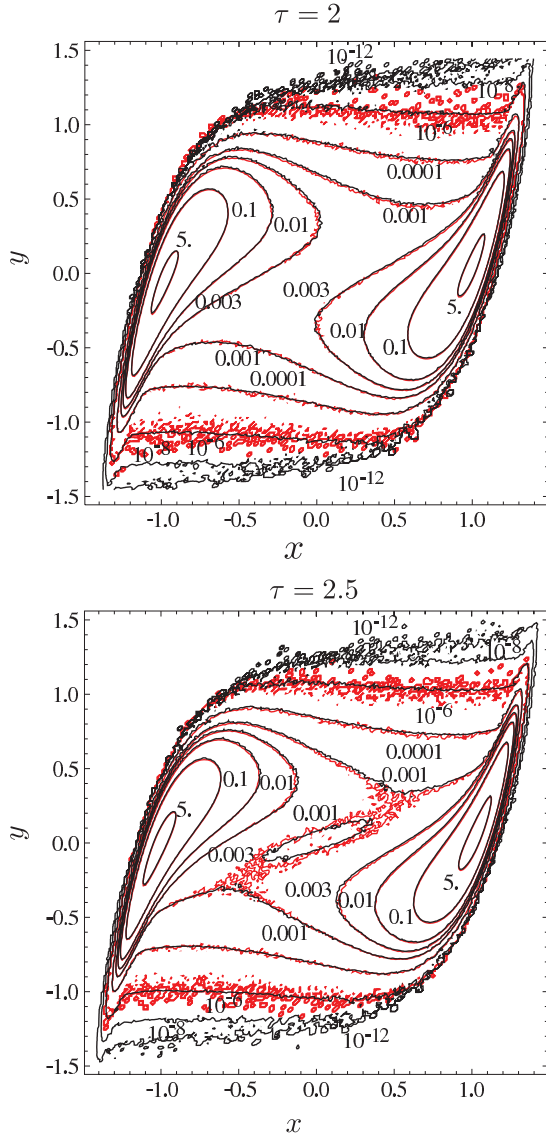


FIG. 8. (Color online) Contour plots of the SPDF of Eq. (18) ( $D = 0.1$ ,  $\tau = 2$ , and  $\tau = 2.5$ ) obtained from the algorithm (black) and BDS (red) of the same running time. Parameters: (top)  $L_x^- = L_y^- = -1.5$ ,  $L_x^+ = L_y^+ = 1.5$ ,  $T_{\text{therm}} = 10$ ,  $N_T = 10^3$ ,  $h = 0.0889$ ,  $M_x = 14\,319$ ,  $M_y = 1273$ ,  $M_{\text{group}} = 165\,003$ ,  $N = 2$  ( $\Delta x = \Delta y h$ ,  $\Delta y = \frac{1}{20} L_{\text{dif}}$ ),  $N_{\text{Brown}} = 10^3$ ; (bottom)  $L_x^- = L_y^- = -1.5$ ,  $L_x^+ = L_y^+ = 1.5$ ,  $T_{\text{therm}} = 12$ ,  $N_T = 10^3$ ,  $h = 0.0889$ ,  $M_x = 17\,899$ ,  $M_y = 1591$ ,  $M_{\text{group}} = 206\,957$ ,  $N = 2$  ( $\Delta x = \Delta y h$ ,  $\Delta y = \frac{1}{20} L_{\text{dif}}$ ),  $N_{\text{Brown}} = 10^3$ . Running times are 19 min (top) and 21 min (bottom).

are depicted in Fig. 8. Our algorithm samples the SPDF down to  $10^{-12}$ , whereas BDS breaks down at a level of  $10^{-6}$ . The minimum, occurring for  $\tau = 2.5$ , was clearly found by the algorithm.

### 3. FitzHugh-Nagumo system

As the last example, we consider the widely used FitzHugh-Nagumo system [42], which is often used in the field of neuroscience [43] or to study synchronization [44,45] and

coherence phenomena [46], represented by

$$\begin{aligned}\dot{x} &= \frac{1}{\epsilon}(x - x^3 - y) + \sqrt{2D_x}\xi_x(t), \\ \dot{y} &= \gamma x - y + b + \sqrt{2D_y}\xi_y(t).\end{aligned}\quad (19)$$

Here  $\epsilon$  denotes the time-scale separation between the activator variable  $x$  and the inhibitor variable  $y$ .  $\xi_x(t)$  and  $\xi_y(t)$ , represent independent zero-mean  $\delta$ -correlated Gaussian white noises. We want to study the stationary probability density in the case of  $D_x = D_y = D$  for a time-scale separation  $\epsilon = 0.1$ . We set the parameters according to Ref. [47] to  $b = 1.4$ ,  $\epsilon = 0.1$ , and  $\gamma = 2$ . Thus, the system is in the excitable regime. Since the deterministic part of the equation for the activator variable increases very rapidly if  $x$  is increased, we have to choose a time step  $h = 0.01$ , which is small enough that the walkers' steps are small compared to 1 but allows us to fulfill the criteria for the size of the subregions. Contour plots of the SPDF are shown in Fig. 9. Regions of low probability, especially, are much better sampled using the algorithm. In the case of low diffusion [ $D = 0.01$  (Fig. 9, top)] the algorithm runs down to  $10^{-16}$ , whereas the BDS stops at a level of  $10^{-6}$ . The minimum, especially, is much better sampled by our algorithm. Note that the local maximum located in the surroundings of  $(x, y) = (0.7, 0.5)$  was not found by BDS. For higher diffusion values [ $D = 0.1$  (Fig. 9, top)], significant differences can be found only in the tails.

## IV. DISCUSSION

### A. One-dimensional system

In the one-dimensional system (see Sec. III A) we succeeded in approximating the SPDF down to  $10^{-300}$ . If equally sized subregions are used, a huge number of subregions has to be implemented, in order to fulfill the convergence criteria (see Sec. II E). However, by evaluating the SPDF according to Eq. (6) the additional computational costs also reduce the fluctuations of the estimated SPDF. If the subregions are too large, the theoretical SPDF is overestimated by the algorithm in potential minima and underestimated in the potentials' maxima.

We also managed to calculate escape rates of size  $10^{-286}$ . Although the finite size of a subregion leads to small errors, in the calculated SPDF, the Arrhenius law is well reproduced for such low probability currents.

Under the stationary regime the  $P_i$  fluctuate around their mean value, estimating the SPDF. By increasing either the number of averages  $N_T$  or the number of walkers per subregion  $N$ , these fluctuations can be reduced, leading to a higher precision. The estimation error scales with  $N_T^{-1/2}$  and  $N^{-1/2}$ , respectively. However, increasing the number of walkers per subregion also affects the computational costs during the thermalization. Simulations for different  $N$  show that runs with a higher  $N$  become stationary more rapidly, but this does not compensate for the additional computational costs. We also find that increasing  $N$  slightly improves the sampling of the SPDF's tails. Once the system is in the stationary regime the increase in the computational costs scale linearly with  $N$  and  $N_T$ . An advantage of simulations with a small  $N$  is that one does not need to perform running steps for subregions with

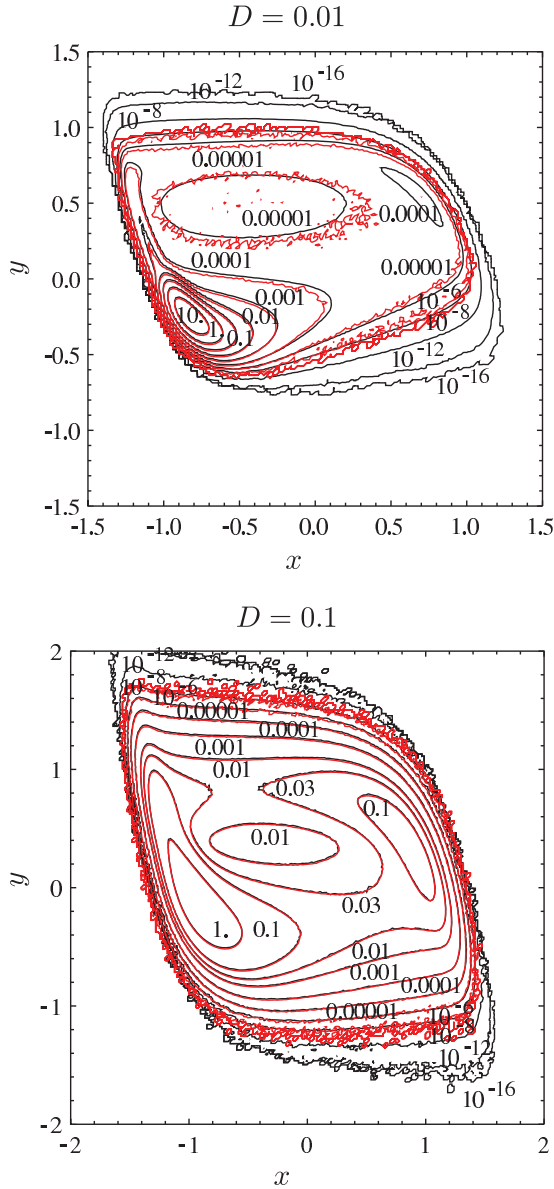


FIG. 9. (Color online) Contour plots of the SPDF of Eq. (19) [ $D = 0.01$  (top) and  $D = 0.1$  (bottom)] obtained from the algorithm (black) and BDS (red) with the same running time. Parameters: (top)  $L_x^- = L_y^- = -2$ ,  $L_x^+ = L_y^+ = 2$ ,  $T_{\text{therm}} = 5$ ,  $N_T = 10^3$ ,  $h = 0.01$ ,  $M_x = 4000$ ,  $M_y = 4000$ ,  $N = 2$  ( $\frac{\Delta x}{\sqrt{Dh}} = \frac{1}{10}$ ),  $N_{\text{Brown}} = 10^3$ , and  $M_{\text{group}} = 314\,027$ ; (bottom)  $L_x^- = L_y^- = -2$ ,  $L_x^+ = L_y^+ = 2$ ,  $T_{\text{therm}} = 3$ ,  $N_T = 10^3$ ,  $h = 0.01$ ,  $M_x = 1333$ ,  $M_y = 1333$ ,  $N = 2$  ( $\Delta x = \Delta y = \frac{1}{20} L_{\text{dir}}$ ),  $N_{\text{Brown}} = 10^3$ ,  $M_{\text{group}} = 146\,568$ . Running times are 27 min (top) and 19 min (bottom). Simulation times for the BDS were chosen three times larger.

$P_i = 0$ , and the number of such regions naturally increases for small  $N$ . We usually use the smallest possible  $N = 2$  and scale the estimation error by increasing  $N_T$ .

### B. Comparison with weighted-ensemble Brownian dynamics

Since the general idea of our method was adapted from prior simulation techniques known as WE sampling [17,18,20], we want to discuss advantages and disadvantages of our algorithm in comparison with these techniques. The main difference

in WE techniques is the redistribution step. Here, using WE techniques, the positions and weights of all walkers are stored, and new walkers are introduced in the stored positions considering their particular weights.

In our algorithm, there is no need to store any position or weight, since walkers are randomly placed in each subregion. The resulting statistical errors can be neglected if the size of the subregions fulfills conditions which, unfortunately, lead to much larger numbers of subregions. By averaging the probabilities, these extra computational costs contribute to a reduction of fluctuations in the stationary regime.

Comparison of the computational costs until equilibration and the achieved precision for a one-dimensional system shows that both algorithms possess the same efficiency for high-precision runs. Comparing the efficiency for the two-dimensional system, Eq. (18), produces quite similar results (see Appendix B).

### C. Multidimensional systems

In the case of two-dimensional systems, we found that the algorithm outperforms BDSs. However, the number of subregions needed according to the criteria can be really high, especially if there are some directions without any noise. Here it is necessary to size the subregions to fulfill the criteria locally. The first step in that direction has already been done by the implementation of the grouping algorithm. Such grouping algorithms become indispensable when high-dimensional systems are studied. Since one needs to simulate the SPDF for each combination of dynamical variables, the number of subregions increases exponentially with the system's dimension. We are quite confident that it is possible to reduce the computational costs by optimizing the mesh. The analyzed examples show that the running times depend greatly on the investigated system. For the bistable system with colored noise and the FitzHugh-Nagumo system, we obtained good results within  $\approx 20$  min, whereas the algorithm required about 5 h for the Van der Pol oscillator.

### D. Integration scheme

All runs were performed using a stochastic version of the Runge-Kutta algorithm, known as the Heun algorithm. Besides an enhanced efficiency in the deterministic part of the trajectory, for the stochastic part it has a mean-square error of order  $h^3$  [34]. Depending on the interpretation (Ito, Stratonovich, etc.), higher order schemes can be used (see, for instance, [7,48]). These schemes allow much larger integration time steps and, therefore, can speed up the simulation. Considering the inequalities (13) and (12), one easily finds that the number of subregions scales according to  $\frac{1}{\sqrt{h}}$  for directions with a finite noise strength, but according to  $\frac{1}{h}$  for directions with purely deterministic flows. But we did not compare the algorithm with the WE method using such higher order schemes.

### V. CONCLUSION

We have provided and tested an algorithm that allows the calculation of low probabilities and low rates. The algorithm



is based on WE methods but uses a uniform distribution of walkers within each subregion. In our findings, the resulting statistical errors can be neglected if one uses subregions that are small compared to the diffusion length. In contrast to WE methods, the required memory does not depend on the number of walkers, which leads to fewer memory requirements for runs with large numbers of walkers.

Special attention has been paid to nonequilibrium dynamical systems. Applying the method to one- and two-dimensional model systems, we analyze its efficiency compared to that of standard BDS. Our method outperforms BDS by several orders of magnitude, its efficiency is comparable to that of WE methods in all studied systems, and it leads to impressive results in regions of low probability and low rates.

### ACKNOWLEDGMENTS

This paper was developed within the scope of IRTG 1740/TRP 2011/50151-0, funded by the DFG/FAPESP. L.S.G. acknowledges the IFISC of the University of Balearic Island for cordial hospitality and support. He also acknowledges Dr. Volkhard Bucholtz, Logos-Verlag Berlin, for earlier participation in and work on the project. R.T. acknowledges financial support from MINECO (Spain), Comunitat Autònoma de les Illes Balears, FEDER, and the European Commission under Projects No. FIS2007-60327 and No. FIS2012-30634, as well as the warm hospitality of Humboldt University.

### APPENDIX A: STATIONARY PROBABILITY DENSITY OF THE POINCARÉ OSCILLATOR

Canonic dissipative systems are noise-driven nonlinear dissipative systems whose stationary probability distribution  $P(x, v)$  depends on the Hamilton function  $H(x, v)$ ; i.e.,  $P(x, v) = P(H)$ . In the case of nonlinear dissipative oscillators with additive noise the dynamics obeys ( $m = 1$ )

$$\begin{aligned}\dot{x} &= \frac{\partial}{\partial v} H(x, v), \\ \dot{v} &= -\frac{\partial}{\partial x} H(x, v) + g(H) \frac{\partial}{\partial v} H(x, v) + \sqrt{2D} \xi(t).\end{aligned}\quad (\text{A1})$$

Using the energy function  $H(x, y) = \frac{1}{2}(x^2 + y^2)$ , we get from Eq. (16) to a representation as a canonic dissipative system:

$$\dot{x} = \partial_y H, \quad \dot{y} = \partial_y(\alpha H - H^2) - \partial_x H + \sqrt{2D} \xi(t). \quad (\text{A2})$$

The corresponding Fokker Planck equation in the  $x, y$  phase space for the SPDF  $p_{\text{st}}(x, y)$  reads [27]

$$\begin{aligned}\partial_t p_{\text{st}} = 0 &= -\partial_y H \partial_x p_{\text{st}} + \partial_x H \partial_y p_{\text{st}} \\ &\quad - \partial_y(\partial_y(\alpha H - H^2) p_{\text{st}}) + D \partial_y^2 p_{\text{st}}.\end{aligned}\quad (\text{A3})$$

Using the ansatz  $p_{\text{st}}(x, y) = p_{\text{st}}(H(x, y))$ , the first two items on the right-hand side cancel. The remaining second line can be integrated once. Assuming an exponentially decaying SPDF at infinitely high energies yields the disappearance of the irreversible probability flux in the  $y$  direction [22,24]. One finds, afterwards,

$$p_{\text{st}} \frac{d}{dH}(\alpha H - H^2) = D \frac{d}{dH} p_{\text{st}}. \quad (\text{A4})$$

This leads to Eq. (17).

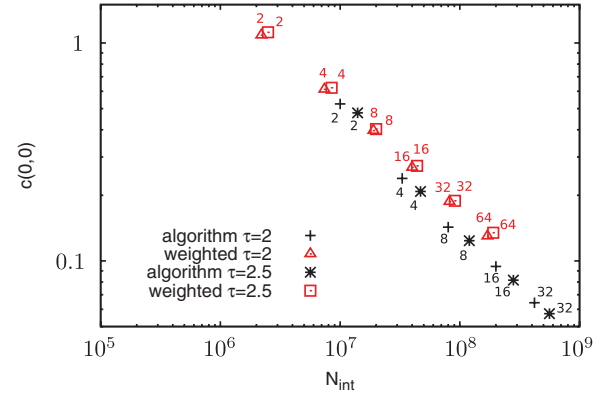


FIG. 10. (Color online) Relative fluctuations plotted over the number of integration steps needed for equilibration, for the algorithm (black) and standard (WE) simulation (red). The number of walkers per subregion is shown for each point. Simulations were done for the system Eq. (18). Parameters for runs of the algorithm were chosen according to Fig. 8 except that  $N$  was varied. Runs of the WE method were performed with the same time step  $h$  for 200 boxes per direction.

### APPENDIX B: EFFICIENCY COMPARED TO THAT OF WEIGHTED-ENSEMBLE BROWNIAN DYNAMICS SIMULATION IN A TWO-DIMENSIONAL SYSTEM

For the two-dimensional system Eq. (18) a comparison with the WE method has been performed. The coefficient of variation, Eq. (14), for the subregion including point (0,0) is plotted over  $N_{\text{int}}$  in Fig. 10. The results are quite similar to the one-dimensional case discussed in Sec III A4. Note that our algorithm leads to smaller fluctuations  $c(0,0)$  for a comparable number of integration steps  $N_{\text{int}}$  for this two-dimensional system.

### APPENDIX C: FOUNDATION OF THE ALGORITHM

In this Appendix we show that the presented algorithm is described by a corresponding master equation for the probability distribution density  $P_i(t)$  for the case of a Markovian hopping process between boxes. We use a single index  $i$  to label the boxes, but the argument applies to any spatial dimension  $d$ . We identify the dynamics of the stochastic system which shall be simulated with the discrete stochastic dynamics of the walkers. The latter is defined via the matrices of probabilities per unit time  $w(i \rightarrow i')$  which describe the hopping in the given discretized space. We assume that it shall converge for sufficiently small time scales and box lengths to the outgoing dynamics.

Let us assume that we have the probability distribution density given at time  $t$  in every box with index  $i$ . It holds that

$$\sum_i P_i(t) = 1. \quad (\text{C1})$$

We redistribute the probability in every box to  $N$  walkers. In the result any walker  $k = 1, \dots, N$  of the same box gets an identical weight:

$$q_i^k(t) = \frac{P_i(t)}{N}. \quad (\text{C2})$$

until the next new redistribution.

At a later time  $t + h$  the walkers may still be inside the box or may have jumped to other boxes. Let  $U_i$  denote all possible box indices which can be reached during a single step from the  $i$  box. Then, in accordance with the assumption above, the probability is  $w(i \rightarrow i')h$  that during  $h$  a single walker leaves the  $i$  box and jumps to the box with index  $i' \in U_i$ . With this hopping the walker  $k$  carries the weight  $q_i^k(t)$  to the new box; this step is, obviously, connected with a loss in the outgoing box. Therefore, the loss per particle for this specific hopping from  $i \rightarrow i'$  can be expressed as

$$w(i \rightarrow i')h q_i^k(t). \quad (\text{C3})$$

The whole loss of weight will be realized in all possible hopping channels. It is identical for all  $N$  particles located in the present box. Hence, the full loss becomes

$$N q_i^k(t) \sum_{i' \in U_i} w(i \rightarrow i')h = P_i(t) \sum_{i' \in U_i} w(i \rightarrow i')h. \quad (\text{C4})$$

Alternatively, one can also introduce  $U'_i$  as the boxes from which walkers can reach the  $i$  box. Then the gain of weight transferred by every walker arriving at the  $i$  box is expressed, if  $i' \in U'_i$ , by

$$w(i' \rightarrow i)h q_{i'}^k. \quad (\text{C5})$$

Again, summing over the different hopping steps, and considering that all the  $N$  walkers inside the box with  $i' \in U'_i$  reach the  $i$  box, yields the gain

$$\sum_{i' \in U'_i} w(i' \rightarrow i)h N q_{i'}^k(t) = \sum_{i' \in U'_i} w(i' \rightarrow i)h P_{i'}(t). \quad (\text{C6})$$

Therefore, the balance of transported weight results in the following shift of the full weight in the  $i$  box at time  $t + h$  compared to the former one,

$$\begin{aligned} P_i(t+h) - P_i(t) &= -P_i(t) \sum_{i' \in U_i} w(i \rightarrow i')h + \sum_{i' \in U'_i} w(i' \rightarrow i)h P_{i'}(t), \end{aligned} \quad (\text{C7})$$

plus corrections of order  $O(h^2)$  corresponding to cases in which two or more walkers jump from one box to another during the time interval  $h$ . After dividing by the time step  $h$  and taking the limit  $h \rightarrow 0$  we obtain the wanted master equation for the discretized dynamics in the boxed phase space:

$$\partial_t P_i(t) = -P_i(t) \sum_{i' \in U_i} w(i \rightarrow i') + \sum_{i' \in U'_i} w(i' \rightarrow i) P_{i'}(t). \quad (\text{C8})$$

- 
- [1] F. Moss and P. V. McClintock, *Noise in Nonlinear Dynamical Systems, Vols. I-III* (Cambridge University Press, New York, 1989).
- [2] N. G. van Kampen, *Stochastic Processes in Physics and Chemistry* (North-Holland, Amsterdam, 1992).
- [3] G. Nicolis and I. Prigogine, *Self-Organization in Nonequilibrium Systems: From Dissipative Structures to Order Through Fluctuations* (Wiley-Interscience, New York, 1977).
- [4] R. Mannella and V. Palleschi, *Phys. Rev. A* **40**, 3381 (1989).
- [5] R. Mannella, in *Stochastic Processes in Physics, Chemistry, and Biology, Lecture Notes in Physics*, Vol. 557, edited by J. A. Freund and T. Pöschel (Springer, Berlin, 2000), p. 353.
- [6] P. Burrage, Ph.D. thesis, University of Queensland, 1999.
- [7] P. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*, Vol. 23 (Springer, Berlin, 2011).
- [8] G. Bhanot, R. Salvador, S. Black, P. Carter, and R. Toral, *Phys. Rev. Lett.* **59**, 803 (1987).
- [9] C. Giardinà, J. Kurchan, and L. Peliti, *Phys. Rev. Lett.* **96**, 120603 (2006).
- [10] A. Dickson and A. Dinner, *Annu. Rev. Phys. Chem.* **61**, 441 (2010).
- [11] F. Wang and D. P. Landau, *Phys. Rev. Lett.* **86**, 2050 (2001).
- [12] G. Torrie and J. Valleau, *J. Comp. Phys.* **23**, 187 (1977).
- [13] A. Warmflash, P. Bhimalapuram, and A. Dinner, *J. Chem. Phys.* **127**, 154112 (2007).
- [14] R. Allen, C. Valeriani, and P. ten Wolde, *J. Phys.: Condens. Matter* **21**, 463102 (2009).
- [15] A. Dickson, A. Warmflash, and A. Dinner, *J. Chem. Phys.* **131**, 154104 (2009).
- [16] C. Valeriani, R. Allen, M. Morelli, D. Frenkel, and P. Wolde, *J. Chem. Phys.* **127**, 114109 (2007).
- [17] G. Huber and S. Kim, *Biophys. J.* **70**, 97 (1996).
- [18] D. Bhatt, B. Zhang, and D. Zuckerman, *J. Chem. Phys.* **133**, 014110 (2010).
- [19] B. Zhang, D. Jasnow, and D. Zuckerman, *J. Chem. Phys.* **132**, 054107 (2010).
- [20] D. Bhatt and I. Bahar, *J. Chem. Phys.* **137**, 104101 (2012).
- [21] D. Bhatt and D. Zuckerman, *J. Chem. Theory Comput.* **7**, 2520 (2011).
- [22] R. Graham and H. Haken, *Z. Phys.* **243**, 289 (1971).
- [23] R. Graham and H. Haken, *Z. Phys.* **245**, 141 (1971).
- [24] H. Haken, *Rev. Mod. Phys.* **47**, 67 (1975).
- [25] H. Risken, *The Fokker Planck Equation*, Vol. 23 (Springer, Berlin, 1984).
- [26] L. Schimansky-Geier, A. V. Tolstopjatenko, and W. Ebeling, *Phys. Lett. A* **108**, 329 (1985).
- [27] W. Ebeling and H. Engel-Herbert, *Physica A* **104**, 378 (1980).
- [28] Y. Klimontovich, *Kinetic Theory of Electromagnetic Processes* (Springer, Berlin, 1983) [in Russian: (Nauka, Moscow, (1980))].
- [29] K. Ito, *Proc. Imp. Acad. Tokyo* **20**, 19 (1944).
- [30] K. Ito, *Mem. Am. Math. Soc.* **4**, 51 (1951).
- [31] R. Stratonovich, *Vestnik Mosk. Univ. Ser. I: Mat. Mekh.* **1**, 3 (1964).
- [32] R. Stratonovich, *SIAM J. Control* **4**, 362 (1966).
- [33] H. Gilsing and T. Shardlow, *J. Comput. Appl. Math.* **205**, 1002 (2007).
- [34] M. San Miguel and R. Toral, in *Instabilities and Nonequilibrium Structures VI*, edited by E. Tirapegui, J. Martinez, and R. Tiemann (Kluwer Academic, Dordrecht, 2000), pp. 35–130.
- [35] L. Gammaitoni, P. Hänggi, P. Jung, and F. Marchesoni, *Rev. Mod. Phys.* **70**, 223 (1998).
- [36] P. Hänggi, P. Talkner, and N. Borkovec, *Rev. Mod. Phys.* **62**, 251 (1990).

- [37] H. Tuckwell, *Introduction to Theoretical Neurobiology*, Vol. 2 (Cambridge University Press, New York, 1988).
- [38] W. Ebeling, H. Herzog, W. Richert, and L. Schimansky-Geier, *Z Math. Mech.* **66**, 141 (1986).
- [39] K. Lekkas, L. Schimansky-Geier, and H. Engel-Herbert, *Z. Phys. B Condens. Matter* **70**, 517 (1988).
- [40] G. Debnath, F. Moss, T. Leiber, H. Risken, and F. Marchesoni, *Phys. Rev. A* **42**, 703 (1990).
- [41] P. Hänggi and P. Jung, *Adv. Chem. Phys.* **89**, 239 (1995).
- [42] R. Fitzhugh, *Biophys. J.* **1**, 445 (1961).
- [43] B. Lindner, J. García-Ojalvo, A. Neiman, and L. Schimansky-Geier, *Phys. Rep.* **392**, 321 (2004).
- [44] C. J. Tessone, E. Ullner, A. A. Zaikin, J. Kurths, and R. Toral, *Phys. Rev. E* **74**, 046220 (2006).
- [45] R. Toral, C. Masoller, C. Mirasso, M. Ciszak, and O. Calvo, *Physica A* **325**, 192 (2003).
- [46] R. Toral, C. Mirasso, and J. Gunton, *Europhys. Lett.* **61**, 162 (2007).
- [47] M. Kostur, X. Sailer, and L. Schimansky-Geier, *Fluct. Noise Lett.* **3**, 155 (2003).
- [48] G. Milstein, *Numerical Integration of Stochastic Differential Equations*, Vol. 313 (Kluwer Academic, Dordrecht, 1995).