Finite-time and finite-size scalings in the evaluation of large-deviation functions: Analytical study using a birth-death process

Takahiro Nemoto,[1,*] Esteban Guevara Hidalgo,[1,2,†] and Vivien Lecomte[1,‡]

[1]*Laboratoire de Probabilités et Modèles Aléatoires, Sorbonne Paris Cité, UMR 7599 CNRS, Université Paris Diderot, 75013 Paris, France*

[2]*Institut Jacques Monod, CNRS UMR 7592, Université Paris Diderot, Sorbonne Paris Cité, F-750205, Paris, France*

(Received 29 July 2016; published 3 January 2017)

The Giardinà-Kurchan-Peliti algorithm is a numerical procedure that uses population dynamics in order to calculate large deviation functions associated to the distribution of time-averaged observables. To study the numerical errors of this algorithm, we explicitly devise a stochastic birth-death process that describes the time evolution of the population probability. From this formulation, we derive that systematic errors of the algorithm decrease proportionally to the inverse of the population size. Based on this observation, we propose a simple interpolation technique for the better estimation of large deviation functions. The approach we present is detailed explicitly in a two-state model.

DOI: 10.1103/PhysRevE.95.012102

I. INTRODUCTION

Cloning algorithms are numerical procedures aimed at simulating rare events efficiently, using a population dynamics scheme. In such algorithms, copies of the system are evolved in parallel and the ones showing the rare behavior of interest are multiplied iteratively [1–18] (see Fig. 1). One of these algorithms proposed by Giardinà *et al.* [12–18] is used to evaluate numerically the cumulant generating function (a large deviation function, LDF) of additive (or time-extensive) observables in Markov processes [19,20]. It has been applied to many physical systems, including chaotic systems, glassy dynamics, and nonequilibrium lattice gas models, and it has allowed the study of novel properties, such as the behavior of breathers in the Fermi-Pasta-Ulam-Tsingou chain [14], dynamical phase transitions in kinetically constrained models [21], and an additivity principle for simple exclusion processes [22,23].

While the method has been used widely, there have been fewer studies focusing on the analytical justification of the algorithm. Even though it is heuristically believed that the LDF estimator converges to the correct result as the number of copies N_c increases, there is no proof of this convergence. Related to this lack of the proof, although we use the algorithm by assuming its validity, we do not have any clue how fast the estimator converges as $N_c \to \infty$. In order to discuss this convergence, we define two types of numerical errors. First, for a fixed finite N_c, averaging over a large number of realizations, the LDF estimator converges to an incorrect value, which is different from the desired large deviation result. We call this deviation from the correct value *systematic errors*. Compared with these errors, we also consider the fluctuations of the estimated value. More precisely, for a fixed value of N_c, the results obtained in different realizations are distributed around this incorrect value. We call the errors associated to these fluctuations the *stochastic errors*. Although both errors are important in numerical simulations, the former one can lead this algorithm to produce wrong results. For example, as seen in Ref. [24], the systematic error grows exponentially as a temperature decreases (or generically in the weak noise limit of diffusive dynamics).

In order to study these errors, we employ a birth-death process [25,26] description of the population dynamics algorithm as explained below: We focus on physical systems described by a Markov dynamics [12,13,16] with a finite number of states M, and we denote by i $(i = 0,1,\ldots,M-1)$ the states of the system. This Markov process has its own stochastic dynamics, described by the transition rates $w(i \to j)$. In population dynamics algorithms, in order to study its rare trajectories, one prepares N_c copies of the system and simulates these copies according to (i) the dynamics of $w(i \to j)$ (followed independently by all copies) and (ii) cloning step in which the ensemble of copies is directly manipulated, i.e., some copies are eliminated while some are multiplied (see Table I). Formally, the population dynamics represents, for a *single* copy of the system, a process that does not preserve probability. This fact has motivated the studies of auxiliary processes [27], effective processes [28], and driven processes [29] to construct modified dynamics (and their approximations [30]) that preserve probability. Different from these methods, in this article, we formulate explicitly the metadynamics of the copies themselves by using a stochastic birth-death process. The process preserves probability, and it allows us to study the numerical errors of the algorithm when evaluating LDF.

In this article, we consider the dynamics of the copies as a stochastic birth-death process whose state is denoted $n = (n_0,n_1,n_2,\ldots,n_{M-1})$, where $0 \leqslant n_i \leqslant N_c$ represents the number of copies which are in state i in the ensemble of copies. We explicitly introduce the transition rates describing the dynamics of n, which we denote by $\sigma(n \to \tilde{n})$. We show that the dynamics described by these transition rates lead in general to the correct LDF estimation of the original system $w(i \to j)$ in the $N_c \to \infty$ limit. We also show that the systematic errors are of the order $O(1/N_c)$, whereas the numerical errors are of the order $O[1/(\tau N_c)]$ (where τ is an averaging duration). This result is in clear contrast with standard Monte Carlo methods, where the systematic errors are always 0. Based on this convergence speed, we then propose a simple interpolation technique to make the cloning algorithm

*nemoto@lpt.ens.fr
†esteban_guevarah@hotmail.com
‡vivien.lecomte@univ-paris-diderot.fr

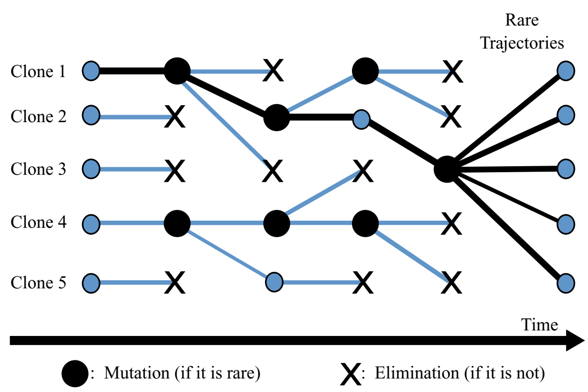

FIG. 1. Schematic picture illustrating the principle of the population dynamics algorithm. Clones (or copies) of the system are prepared and they evolve following a mutation-and-selection process, maintaining the total population constant.

more reliable. Furthermore, the formulation developed in this paper provides us the possibility to compute exactly the expressions of the convergence coefficients, as we do in Sec. IV on a simple example.

The analytical analysis presented in this paper is supplemented with a thorough numerical study in a companion paper [31]. In the companion paper, we employ an intrinsically different cloning algorithm, which is the continuous-time population dynamics algorithm, that cannot be studied by the methods presented in this paper (see Sec. II D 2). We show in the companion paper [31] that the validity of the scaling that we derive analytically here is very general. In particular, we demonstrate in practice the efficiency of the interpolation technique in the evaluation of the LDF, irrespective of the details of the population dynamics algorithm.

The construction of this paper is as follows. We first define the LDF problem in the beginning of Sec. II, and then formulate the birth-death process used to describe the algorithm in Sec. II A. By using this birth-death process, we demonstrate that the estimator of the algorithm converges to the correct large deviation function in Sec. II B. At the end of this section, in Sec. II C, we discuss the convergence speed of this estimator (the systematic errors) and derive its scaling $\sim 1/N_c$. In Sec. III, we turn to stochastic errors. For discussing this, we introduce the large deviation function of the estimator, from which we derive that the convergence speed of the stochastic errors is proportional to $1/(\tau N_c)$. In the next section, Sec. IV, we introduce a simple two-state model, to which we apply the formulations developed in the previous sections. We derive the exact expressions of the systematic errors in Sec. IV A and of the stochastic errors in Sec. IV B. At the end of this section, in Sec. IV C, based on these exact expressions, we propose another large deviation estimator defined in the population dynamics algorithm. In the final section, Sec. V, we first summarize the result obtained throughout this paper, and then in Sec. V A, we propose a simple interpolation technique based on the convergence speed of the systematic errors which allows us to devise a better practical evaluation of the LDF. Finally in Sec. V B, we discuss two open questions.

II. BIRTH-DEATH PROCESS DESCRIBING THE POPULATION DYNAMICS ALGORITHM

As explained in the introduction (also see Table I), the state of the population is $n = (n_0,n_1,\dots,n_{M-1})$, where n_i represents the number of clones in the state i. The total population is preserved: $\sum_i n_i = N_c$. Below, we introduce the transition rates of the dynamics between the occupations n, $\sigma(n \to \tilde{n})$ that describe corresponding large deviations of the original system, where the dynamics of the original system is given by the rates $w(i \to j)$ as detailed below.

As the original system, we consider the continuous-time Markov process in a discrete-time representation. By denoting by dt as the time step, the transition matrix $R_{j,i}$ for time evolution of the state i is described as

$$R_{j,i} = \delta_{i,j} + dt\left[w(i \to j) - \delta_{i,j}\sum_k w(i \to k)\right], \qquad (1)$$

where we set $w(i \to i) = 0$. The probability distribution of the state i, $p_i(t)$, evolves in time as $p_i(t+dt) = \sum_j R_{i,j}p_j(t)$. In the $dt \to 0$ limit, one obtains the continuous-time master equation describing the evolution of $p_i(t)$ [25,26]. For simplicity, especially for the cloning part of the algorithm, we keep here a small finite dt. The reason why we use a discrete-time representation is solely for simplicity of the discussion. The main results can be derived even if we start with a continuous-time representation (see Sec. II D 1). For the original dynamics described by the transition matrix (1), we consider an observable b_i depending on the state i and we are interested in the distribution of its time-averaged value during a time interval τ, defined as

$$B(\tau) = \frac{1}{\tau}\sum_{t=0}^{\tau/dt} dt\, b_{i(t)}. \qquad (2)$$

TABLE I. Correspondence between the population dynamics and the birth-death process to describe it.

	Population dynamics algorithm	Birth-death process describing the population dynamics
State of the system	i	$n = (n_0,n_2,\dots,n_{M-1})$
	$(i = 0,1,\dots,M-1)$	$(0 \leqslant n_i \leqslant N_c$ with $\sum_i n_i = N_c)$
Transition rates	$w(i \to j)$	$\sigma(n \to \tilde{n})$
	Markov process on states i	Markov process on states n
Numerical procedure	Prepare N_c clones and evolve those	Described by the dynamics
for rare-event sampling	with a mutation-selection procedure	of rates $\sigma(n \to \tilde{n})$

TABLE II. Transition matrices [see Eq. (4)] describing the birth-death process.

	Transition matrices
Dynamics ("mutations")	$\mathcal{T}_{\tilde{n},n} \equiv \delta_{\tilde{n},n} + dt \sum_{i=0}^{M-1} n_i \sum_{j=0,(j\neq i)}^{M-1} w(i\to j)\big[\delta_{\tilde{n}_i,n_i-1}\delta_{\tilde{n}_j,n_j+1}\,\delta_{\tilde{n},n}^{i,j} - \delta_{\tilde{n},n}\big]$
Cloning ("selection")	$\mathcal{C}_{\tilde{n},n} = \delta_{\tilde{n},n} + s\,dt \sum_{i=0}^{M-1} n_i\lvert\alpha_i\rvert\big[\delta_{\tilde{n}_i,n_i+\alpha_i/\lvert\alpha_i\rvert}\,\delta_{\tilde{n},n}^{i} - \delta_{\tilde{n},n}\big] + \mathcal{O}(dt^2)$
Maintaining N_c	$\mathcal{K}_{\tilde{n},n} = \delta_{\sum n_i,N_c}\delta_{\tilde{n},n} + \sum_{k=-1,1}\delta_{\sum_i n_i,N_c+k}\sum_{i=0}^{M-1}\delta_{\tilde{n}_i,n_i-k}\,\delta_{\tilde{n},n}^{i}\,\frac{n_i}{N_c+k}$
Full process	$(\mathcal{KCT})_{\tilde{n},n} = \delta_{\tilde{n},n} + dt\sum_{i=0}^{M-1} n_i \sum_{j=0,(j\neq i)}^{M-1}[w(i\to j) + s\tilde{w}_n(i\to j)]\big[\delta_{\tilde{n}_i,n_i-1}\delta_{\tilde{n}_j,n_j+1}\,\delta_{\tilde{n},n}^{i,j} - \delta_{\tilde{n},n}\big]$
	with $\tilde{w}_n(i\to j) = \frac{n_j}{N_c}\big[\alpha_j\delta_{j\in\Omega^{(+)}}\frac{N_c}{N_c+1} - \alpha_i\delta_{i\in\Omega^{(-)}}\frac{N_c}{N_c-1}\big]$

Here $i(t)$ is a trajectory of the system generated by the Markov dynamics described by $R_{j,i}$. We note that $B(\tau)$ is a path- (or history- or realization-) dependent quantity. Since $\tau B(\tau)$ is an additive observable, the fluctuations of $B(\tau)$ depending on the realizations are small when τ is large, but one can describe the large deviations of $B(\tau)$. Those occur with a small probability and obey a large deviation principle. We denote by Prob(B) the distribution function of $B(\tau)$. The large deviation principle ensures that Prob(B) takes an asymptotic form Prob$(B) \sim \exp[-\tau I(B)]$ for large τ, where $I(B)$ is a large deviation function (or rate function) [19,20]. If the rate function $I(B)$ is convex, the large deviation function is expressed as a Legendre transform of a cumulant generating function (CGF) $\psi(s)$ defined as

$$\psi(s) = \lim_{\tau\to\infty}\frac{1}{\tau}\log\langle e^{-s\tau B(\tau)}\rangle, \tag{3}$$

namely, $I(B) = -\inf_s[sB + \psi(s)]$. The large deviation function $I(B)$ and this generating function $\psi(s)$ are by definition difficult to evaluate numerically in Monte Carlo simulations of the original system of transition rates $w(i\to j)$ (see, for example, [32]). To overcome this difficulty, population dynamics algorithms have been developed [12–18]. Here, we describe this population dynamics algorithm by using a birth-death process on the occupation state n, allowing us to study systematically the errors in the estimation of $\psi(s)$ within the population dynamics algorithm. We mention that, without loss of generality, we restrict our study to so-called type B observables that do not depend on the transitions of the state [33], i.e., which are time integrals of the state of the system, as in (2). Indeed, as explained, for example, in Refs. [16] and [24], one can always reformulate the determination of the CGF of mixed-type observables into that of a type B variable, by modifying the transition rates of the given system.

A. Transition matrices representing the population dynamics algorithm

We denote the probability distribution of the occupation n at time t by $P_n(t)$. The time evolution of this probability is decomposed into three parts. The first one is the original Monte Carlo dynamics based on the transition rates $w(i\to j)$. The second one is the cloning procedure of the population dynamics algorithm, which favors or disfavors configurations according to a well-defined rule. The third one is a supplementary (but important) part which maintains the total number of clones to a constant N_c. We denote the transition matrices corresponding to these steps by $\mathcal{T}$, $\mathcal{C}$, and $\mathcal{K}$, respectively. By using these matrices, then, the time evolution of the distribution function is given as

$$P_n(t+dt) = \sum_{\tilde{n}}(\mathcal{KCT})_{n,\tilde{n}}P_{\tilde{n}}(t). \tag{4}$$

We derive explicit expressions of these matrices in the following subsections. We also summarize the obtained results in Table II.

1. Derivation of the original dynamics part, $\mathcal{T}$

We first consider the transition matrix $\mathcal{T}$, which describes the evolution of the occupation state n solely due to the dynamics based on the rates $w(i\to j)$. During an infinitesimally small time step dt, the occupation $n = (n_0,n_1,\ldots,n_{M-1})$ changes to $\tilde{n} = (n_0,n_1,\ldots,n_i-1,\ldots,n_j+1,\ldots,n_{M-1})$, where $0\leqslant i<M$ and $0\leqslant j<M$ (for all $i\neq j$). Since there are n_i clones in the state i before the transition, the transition probability of this change is given as $n_i w(i\to j)dt$. Thus, we obtain

$$\mathcal{T}_{\tilde{n},n} \equiv \delta_{\tilde{n},n} + dt\sum_{i=0}^{M-1} n_i \sum_{j=0,(j\neq i)}^{M-1} w(i\to j) \times\big[\delta_{\tilde{n}_i,n_i-1}\delta_{\tilde{n}_j,n_j+1}\,\delta_{\tilde{n},n}^{i,j} - \delta_{\tilde{n},n}\big], \tag{5}$$

where $\delta_{\tilde{n},n}^{i,j}$ is a Kronecker δ for the indices except for i,j: $\delta_{\tilde{n},n}^{i,j} \equiv \prod_{k\neq i,j}\delta_{\tilde{n}_k,n_k}$. One can easily check that this matrix satisfies the conservation of the probability: $\sum_{\tilde{n}}\mathcal{T}_{\tilde{n},n} = 1$. It corresponds to the evolution of N_c independent copies of the original system with rates $w(i\to j)$.

2. Derivation of the cloning part, $\mathcal{C}$

In the population dynamics algorithm (for example, the one described in Appendix A of Ref. [24]), at every certain time interval Δt, one evaluates the exponential factor for all clones, which is equal to $e^{-s\int_t^{t+\Delta t}dt' b_{i(t')}}$ if the clone is in state $(i(t'))_{t'=t}^{t+\Delta t}$ during a time interval $t\leqslant t'\leqslant t+\Delta t$. We also call this exponential factor *cloning ratio*, because this factor determines whether each clone is copied or eliminated after this time interval. Although the details of how to determine this selection process can depend on the specific type of algorithms, the common idea is that each of the clones is copied or eliminated in such a way that a clone in state $i(t)$ has a number of descendant(s) proportional to the cloning factor on average after this time interval.

In order to implement this idea in our birth-death process, we assume this time step Δt to be small. For the sake of simplicity, we set this Δt to be our smallest time interval

dt: $\Delta t = dt$. This condition is not mandatory whenever the $\Delta t \to 0$ limit is taken at the end (see Sec. II D 1 for the case $\Delta t > dt$). Then, noticing that the time integral $\int_t^{t+\Delta t} dt' b_{i(t')}$ is expressed as $dt\, b_{i(t)}$ for small dt, we introduce the following quantity for each state i ($i = 0,1,2,\dots,M-1$):

$$\nu_i \equiv \frac{n_i e^{-s dt b_i}}{\sum_{j=0}^{M-1} n_j e^{-s dt b_j}} N_c. \tag{6}$$

Note that there is a factor n_i in front of the exponential function $e^{-s dt b_i}$ which enumerates the number of clones that occupy the state i. The quantity ν_i is aimed at being the number of clones in state i after the cloning process; however, since ν_i is not an integer but a real number, one needs a supplementary prescription to fix the corresponding integer number of descendants. In general, in the implementation of population dynamics, this integer is generated randomly from the factor ν_i, equal either to its lower or to its upper integer part. The probability to choose either the lower or upper integer part is fixed by imposing that the number of descendants is equal to ν_i on average. For instance, if ν_i is equal to 13.2, then 13 is chosen with probability 0.8, and 14 with probability 0.2. Generically, $\lfloor \nu_i \rfloor$ and $\lfloor \nu_i \rfloor + 1$ are chosen with probability $1 + \lfloor \nu_i \rfloor - \nu_i$ and $\nu_i - \lfloor \nu_i \rfloor$, respectively. We note that we need to consider these two possibilities for all indices i. We thus arrive at the following matrix:

$$\mathcal{C}_{\tilde{n},n} \equiv \sum_{x_0=0}^{1}\sum_{x_1=0}^{1}\sum_{x_2=0}^{1}\cdots\sum_{x_{M-1}=0}^{1}\prod_{i=0}^{M-1}\delta_{\tilde{n}_i,\lfloor \nu_i \rfloor + x_i} \times [(\nu_i - \lfloor \nu_i \rfloor)x_i + (1 + \lfloor \nu_i \rfloor - \nu_i)(1-x_i)]. \tag{7}$$

Now, we expand $\mathcal{C}$ at small dt and we keep only the terms proportional to $O(1)$ and $O(dt)$, which do not vanish in the continuous-time limit. For this purpose, we expand ν_i as

$$\nu_i = n_i\left[1 + s\, dt\left(\sum_j \frac{n_j b_j}{N_c} - b_i\right)\right] + O(dt^2), \tag{8}$$

where we have used $\sum_i n_i = N_c$. This expression indicates that $\lfloor \nu_i \rfloor$ is determined depending on the sign of $\sum_j n_j b_j / N_c - b_i$, where we assumed $s > 0$ for simplicity without loss of generality (because when $s < 0$, we can always redefine $-b$ as b to make s to be positive). By denoting this factor by α_i, i.e.,

$$\alpha_i(n) \equiv \sum_j \frac{n_j b_j}{N_c} - b_i, \tag{9}$$

we thus define the following state space $\Omega^{(\pm)}(n)$:

$$\Omega^{(\pm)}(n) = \{i | 0 \leqslant i < M \quad \text{and} \quad \pm\alpha_i(n) > 0\}. \tag{10}$$

From this definition, for sufficiently small dt, we obtain

$$\lfloor \nu_i \rfloor = n_i \tag{11}$$

for $i \in \Omega^{(+)}$, and

$$\lfloor \nu_i \rfloor = n_i - 1 \tag{12}$$

for $i \in \Omega^{(-)}$. Substituting these results into (7) and expanding in dt, we obtain [denoting here and thereafter $\alpha_i = \alpha_i(n)$]

$$\mathcal{C}_{\tilde{n},n} = \delta_{\tilde{n},n} + s\, dt \sum_{i=0}^{M-1} n_i |\alpha_i| \times \left[\delta_{\tilde{n}_i, n_i + \alpha_i/|\alpha_i|}\, \delta^i_{\tilde{n},n} - \delta_{\tilde{n},n}\right] + O(dt^2), \tag{13}$$

where $\delta^i_{\tilde{n},n}$ is a Kronecker δ for the indices except for i: $\delta^i_{\tilde{n},n} = \prod_{k\neq i} \delta_{\tilde{n}_k,n_k}$. One can easily check that this matrix preserves probability: $\sum_{\tilde{n}} \mathcal{C}_{\tilde{n},n} = 1$.

3. Derivation of the maintaining part, $\mathcal{K}$

As directly checked, the operator $\mathcal{T}$ preserves the total population $\sum_i n_i$. However, the operator representing the cloning $\mathcal{C}$ does not. In our birth-death implementation, this property originates from the rounding process $\lfloor \nu_i \rfloor$ in the definition of $\mathcal{C}$: Even though ν_i itself satisfies $\sum_i \nu_i = N_c$, because of the rounding process of ν_i, the number of clones after multiplying by $\mathcal{C}$ (that is designed to be proportional to ν_i on average) can change. There are several ways to keep the number N_c of copies constant without biasing the distribution of visited configurations. One of them is to choose randomly and *uniformly* δN_c clones from the ensemble, where δN_c is equal to the number of excess (respectively, lacking) clones with respect to N_c, and to eliminate (respectively, multiply) them.

In our birth-death description, we implement this procedure as follows. We denote by $\mathcal{K}$ the transition matrix maintaining the total number of clones to be the constant N_c. We now use a continuous-time asymptotics $dt \to 0$. In this limit, from the expression of the transition matrix elements (13), we find that at each cloning step the number of copies of the cloned configuration varies by ± 1 at most. Hence, the total number of clones after multiplying by $\mathcal{C}$, $\sum_i n_i$, satisfies the following inequality

$$N_c - 1 \leqslant \sum_i n_i \leqslant N_c + 1. \tag{14}$$

Among the configurations n that satisfy this inequality, there are three possibilities, which are $\sum_i n_i = N_c$ and $\sum_i n_i = N_c \pm 1$. If n satisfies $\sum_i n_i = N_c$, we do not need to adjust n, while if n satisfies $\sum_i n_i = N_c + 1$ (respectively, $\sum_i n_i = N_c - 1$), we eliminate (respectively, multiply) a clone chosen randomly and uniformly. Note that in our formulation, we do not distinguish the clones taking the same state. This means that we can choose one of the occupations n_i of a state i according to a probability proportional to the number of copies n_i in this state. In other words, the probability to choose the state i and to copy or to eliminate a clone from this state is proportional to $n_i / \sum_{j=0}^{M-1} n_j$. Therefore, we obtain the expression of the matrix $\mathcal{K}$ as

$$\mathcal{K}_{\tilde{n},n} = \delta_{\sum_i n_i, N_c}\delta_{\tilde{n},n} + \sum_{k=-1,1} \delta_{\sum_i n_i, N_c + k} \times \sum_{i=0}^{M-1} \delta_{\tilde{n}_i, n_i - k}\, \delta^i_{\tilde{n},n}\, \frac{n_i}{N_c + k} \tag{15}$$

for $\tilde{n}$ that satisfies $\sum_i \tilde{n}_i = N_c$, and $\mathcal{K}_{\tilde{n},n} = 0$ otherwise.

4. Total transition, $\mathcal{KCT}$

We write down the matrix describing the total transition of the population dynamics [see Eq. (4)]. From the obtained expressions of $\mathcal{K}$, $\mathcal{C}$, $\mathcal{T}$, we calculate $\mathcal{KCT}$

$$\begin{aligned}(\mathcal{KCT})_{\tilde{n},n} &= \delta_{\tilde{n},n} + dt \sum_{i=0}^{M-1} n_i \sum_{j=0,(j\neq i)}^{M-1} [w(i\to j) + s\,\tilde{w}_n(i\to j)] \\ &\quad \times \left[\delta_{\tilde{n}_i,n_i-1}\delta_{\tilde{n}_j,n_j+1}\,\delta^{i,j}_{\tilde{n},n} - \delta_{\tilde{n},n}\right], \end{aligned} \tag{16}$$

where the population-dependent transition rate $\tilde{w}_n(i\to j)$ is given as

$$\tilde{w}_n(i\to j) = \frac{n_j}{N_c}\left[\alpha_j\delta_{j\in\Omega^{(+)}}\frac{N_c}{N_c+1} - \alpha_i\delta_{i\in\Omega^{(-)}}\frac{N_c}{N_c-1}\right]. \tag{17}$$

The comparison of the expression (16) with the original part $\mathcal{T}$ provides an insight into the obtained result. The jump ratio $w(i\to j)$ in the original dynamics is replaced by $w(i\to j) + s\,\tilde{w}_n(i\to j)$ in the population dynamics algorithm. We note that this transition rate depends on the population n, meaning that we cannot get a closed equation for this modified dynamics at the level of the states i in general. We finally remark that the transition matrix $\sigma(n\to\tilde{n})$ for the continuous-time limit is directly derived from (16) as

$$\begin{aligned}\sigma(n\to\tilde{n}) &= \sum_{i=0}^{M-1} n_i \sum_{j=0,(j\neq i)}^{M-1} [w(i\to j) + s\tilde{w}_n(i\to j)] \\ &\quad \times \left[\delta_{\tilde{n}_i,n_i-1}\delta_{\tilde{n}_j,n_j+1}\,\delta^{i,j}_{\tilde{n},n}\right]. \end{aligned} \tag{18}$$

B. Derivation of the large deviation results in the $N_c\to\infty$ asymptotics

In this subsection, we study the $N_c\to\infty$ limit for the transition matrix of rates $\sigma(n\to\tilde{n})$ and derive the validity of the population dynamics algorithm.

1. The estimator of the large deviation function

One of the ideal implementations of the population dynamics algorithm is as follows: We make copies of each realization (clone) at the end of simulation, where the number of copies for each realization is equal to the exponential weight $e^{-s\tau B(\tau)}$ in Eq. (3) (so that we can discuss an ensemble with this exponential weight without multiplying the probability by it). In this implementation, the number of clones grows (or decays) exponentially proportionally as $\langle e^{-s\tau B(\tau)}\rangle$ by definition. In real implementations of the algorithm, however, since taking care of an exponentially large or small number of clones can cause numerical problems, one rather keeps the total number of clones to a constant N_c at every time step, as seen in (6). Within this implementation, we reconstruct the exponential change of the total number of clones as follows: We compute the average of cloning ratio (see the beginning of Sec. II A 2 for its definition) at each cloning step, and we store the product of these ratios along the cloning steps. At final time, this product gives the empirical estimation of total (unnormalized) population during the whole duration of the simulation [18], i.e., an estimator of $\langle e^{-s\tau B(\tau)}\rangle$. One thus estimates the CGF $\psi(s)$ given in Eq. (3) [12–18] as the logarithm of this reconstructed population, divided by the total time.

In our formulation, the average cloning ratio is given as $\sum_i n_i e^{-sdtb_i}/N_c$, and thus the multiplication over whole time interval reads $\prod_{t=0}^{\tau/dt}\{n_i(t)e^{-sdtb_i}/N_c\}$. Because we empirically assume that the CGF estimator converges to $\psi(s)$ in the $N_c,\tau\to\infty$ limit, the following equality is expected to hold in probability 1:

$$\psi(s) \overset{?}{=} \lim_{N_c\to\infty}\lim_{\tau\to\infty}\frac{1}{\tau}\sum_{t=0}^{\tau/dt}\log\sum_i\frac{n_i(t)e^{-sdtb_i}}{N_c} + O(dt). \tag{19}$$

Since the dynamics of the population n is described by a Markov process, ergodicity is satisfied, i.e., time averages can be replaced by the expected value with respect to the stationary distribution function. Applying this result to the right-hand side of (19), we obtain

$$\begin{aligned}&\lim_{\tau\to\infty}\frac{1}{\tau}\sum_{t=0}^{\tau/dt}\log\sum_i\frac{n_i(t)e^{-sdtb_i}}{N_c} \\ &\quad = \frac{1}{dt}\sum_n P_n^{\mathrm{st}}\log\sum_i\frac{n_ie^{-sdtb_i}}{N_c} + O(dt), \end{aligned} \tag{20}$$

where P_n^{st} is the stationary distribution function of the population n in the $dt\to 0$ limit, (namely, P_n^{st} is the stationary distribution of the dynamics of transition rates $\sigma(n\to\tilde{n})$). By expanding this right-hand side with respect to dt, we rewrite the expected equality (19) as

$$\psi(s) \overset{?}{=} -s\lim_{N_c\to\infty}\sum_n P_n^{\mathrm{st}}\sum_i\frac{n_ib_i}{N_c} + O(dt). \tag{21}$$

where we used that $\sum_i n_i = N_c$ is a conserved quantity. Below we demonstrate that this latter equality (21) is satisfied by analyzing the stationary distribution function P_n^{st}.

2. The connection between the distribution functions of the population and of the original system

From the definition of the stationary distribution function P_n^{st}, we have

$$\sum_{\tilde{n}} P_{\tilde{n}}^{\mathrm{st}}\sigma(\tilde{n}\to n) - \sum_{\tilde{n}} P_n^{\mathrm{st}}\sigma(n\to\tilde{n}) = 0 \tag{22}$$

(which is a stationary master equation.) In this equation, we use the explicit expression of σ shown in (18). By denoting by $n^{j\to i}$ the configuration where one clone in the state j moves to the state i: $n^{j\to i} \equiv (n_0,n_1,\dots,n_i+1,\dots,n_j-1,\dots,n_{M-1})$, and the stationary master equation (22) is rewritten as

$$\sum_{i,j(i\neq j)}[f_{i\to j}(n^{j\to i}) - f_{i\to j}(n)] = 0, \tag{23}$$

where we defined $f_{i\to j}(n)$ as

$$f_{i\to j}(n) = P_n^{\mathrm{st}}n_i[w(i\to j) + s\tilde{w}_n(i\to j)]. \tag{24}$$

Now we multiply expression (23) by n_k (k is arbitrary from $k = 0,1,2,\ldots,M-1$), and sum it over all configurations n:

$$\sum_n \sum_{i,j(i\neq j)} n_k[f_{i\to j}(n^{j\to i}) - f_{i\to j}(n)] = 0. \tag{25}$$

We can change the dummy summation variable n in the first term to $n^{i\to j}$, which leads to $\sum_n \sum_{i,j(i\neq j)}(n^{i\to j})_k f_{i\to j}(n)$. Since the second term has almost the same expression as the first one except for the factor n_k, the sum in (25) over the indices (i,j), where none of i nor j is equal to k, becomes 0. The remaining term in (25) is thus

$$\begin{aligned}0 = &\sum_n \sum_{j(j\neq k)} [(n^{k\to j})_k - n_k] f_{k\to j}(n)\\ &+ \sum_n \sum_{i(i\neq k)} [(n^{i\to k})_k - n_k] f_{i\to k}(n).\end{aligned} \tag{26}$$

Using the definition of $n^{i\to j}$ in this equation, we arrive at

$$0 = \sum_n \left[\sum_{i(i\neq k)} f_{i\to k}(n) - f_{k\to i}(n) \right]. \tag{27}$$

This equation (27) connects the stationary property of the population dynamics (described by the occupation states n) and the one in the original system (described by the states i).

The easiest case where we can see this connection is when $s = 0$. By defining the empirical occupation probability of the original system as $p_i \equiv \sum_n P_n^{\rm st} n_i/N_c$, Eq. (27) leads to the following (stationary) master equation for $w(i\to j)$:

$$0 = \sum_j p_j w(j\to i) - \sum_j p_i w(i\to j) \quad (\text{for s} = 0). \tag{28}$$

This is valid for any N_c, meaning that, for original Monte Carlo simulations in $s = 0$, the empirical probability p_i is exactly equal to the steady-state probability, being the unique solution of (28). It means that there are no systematic errors in the evaluation of p_i (see the introduction of this paper for the definition of the term *systematic errors*). However, in the generic case $s \neq 0$, this property is not satisfied. One thus needs to understand the $N_c \to \infty$ limit to connect the population dynamics result with the large deviation property of the original system.

3. *Justification of the convergence of the large deviation estimator as population size becomes large*

In order to take the $N_c \to \infty$ limit, we define a scaled variable x_i as n_i/N_c. With keeping this occupation fractions x_i to be $O(1)$, we take the $N_c \to \infty$ limit in (27), which leads to

$$\begin{aligned}0 = \sum_n P_n^{\rm st} \Bigg[&\sum_j x_j w(j\to i) - \sum_j x_i w(i\to j)\\ &- s x_i \Bigg(b_i - \sum_k x_k b_k \Bigg) \Bigg] + O(1/N_c).\end{aligned} \tag{29}$$

Inspired by this expression, we define a matrix $L^s_{i,j}$ as

$$L^s_{i,j} = w(j\to i) - \delta_{i,j}\left(\sum_k w(i\to k) + s b_i \right), \tag{30}$$

and a correlation function between x_i and x_j as

$$c_{i,j} = \sum_n x_i x_j P_n^{\rm st} - p_i p_j \tag{31}$$

(where we recall $p_i \equiv \sum_n x_i P_n^{\rm st}$). From these definitions, (29) is rewritten as

$$\sum_j p_j L^s_{i,j} = -s p_i \sum_k p_k b_k - s \sum_k c_{i,k} b_k + O\left(\frac{1}{N_c}\right). \tag{32}$$

Since x_i is an averaged quantity (an arithmetic mean) with respect to the total number of clones ($x_i \equiv n_i/N_c$), we can safely assume that the correlation $c_{i,j}$ becomes 0 in $N_c \to \infty$ limit:

$$\lim_{N_c\to\infty} c_{i,k} = 0. \tag{33}$$

[For more detailed discussion of why this is valid, see the description after Eq. (36).] Thus, by defining $p_i^\infty \equiv \lim_{N_c\to\infty} p_i$, we obtain

$$\sum_j p_j^\infty L^s_{i,j} = -s p_i^\infty \sum_k p_k^\infty b_k. \tag{34}$$

From the Perron-Frobenius theory, the positive eigenvector of the matrix $L^s_{i,j}$ is unique and corresponds to its eigenvector of largest eigenvalue (in real part). This means that $-s\sum_k p_k^\infty b_k$ is the largest eigenvalue of the matrix $L^s_{i,j}$. Finally, by recalling that the largest eigenvalue of this matrix $L^s_{i,j}$ is equal to the generating function $\psi(s)$ (see Ref. [33], for example), we have finally justified that the CGF estimator (21) is valid in the large-N_c limit.

C. Systematic errors due to finite N_c; convergence speed of the large deviation estimator as $N_c \to \infty$

In the introduction of this paper, we defined the systematic errors as the deviations of the large deviation estimator from the correct value due to a finite number of clones N_c. From (21), we quantitatively define this systematic error $\epsilon_{\rm sys}$ as

$$\epsilon_{\rm sys} \equiv \left| \psi(s) + s \sum_i p_i b_i \right|. \tag{35}$$

From a simple argument based on a system size expansion, we below show that this $\epsilon_{\rm sys}$ is of order $O(1/N_c)$.

We first show that one can perform a system size expansion (as, e.g., in van Kampen [25]) for the population dynamics. In (23), by recalling the definition of the vector x as $x = n/N_c$, and by denoting $\tilde{P}^{\rm st}(x) = P^{\rm st}_{xN_c}$, we obtain

$$\begin{aligned}0 = &\sum_{i,j(i\neq j)} \sum_{r=1}^\infty \frac{1}{r!}\frac{1}{N_c^r}\left(\frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_j} \right)^r x_i \tilde{P}^{\rm st}(x)\\ &\times [w(i\to j) + s\tilde{w}_n(i\to j)|_{n=xN_c}].\end{aligned} \tag{36}$$

This indicates that the stochastic process governing the evolution of x becomes deterministic in the $N_c \to \infty$ limit.

The deterministic trajectory for x is governed by a differential equation derived from the sole term $r = 1$ in the expansion (36) (see, e.g., Sec. 3.5.3 in Ref. [26] for the detail of how to derive this property). Thus if x converges to a fixed point as N_c increases, *which is normally observed in implementations of cloning algorithms*, the assumption (33) is satisfied.

From the expression of $\epsilon_{\rm sys}$, we see that the dependence in N_c comes solely from p_i, which can be calculated from the first-order correction of $P_n^{\rm st}$ (at large N_c). The equation to determine $P_n^{\rm st}$ is the stationary master equation (22) or equivalently, the system-size expansion formula (36). We expand the jump ratio $w(i \to j) + s\tilde{w}_n(i \to j)$ in (36) with respect to $1/N_c$ as

$$w(i \to j) + s\tilde{w}_n(i \to j) = w(i \to j) + s\tilde{w}_x^{\infty}(i \to j) + \frac{s}{N_c}\delta w_x(i \to j) + O\big(1/N_c^2\big), \tag{37}$$

where $\tilde{w}_x^{\infty}(i \to j)$ and $\delta w_x(i \to j)$ are defined as

$$\tilde{w}_x^{\infty}(i \to j) = x_j[\alpha_j \delta_{j\in\Omega^{(+)}} - \alpha_i \delta_{i\in\Omega^{(-)}}] \tag{38}$$

and

$$\delta w_x(i \to j) = -x_j[\alpha_j \delta_{j\in\Omega^{(+)}} + \alpha_i \delta_{i\in\Omega^{(-)}}]. \tag{39}$$

By substituting (37) into the system-size expansion formula (36) and performing a perturbation expansion, we find that a first-order correction of p is naturally of order $O(1/N_c)$, i.e., $\epsilon_{\rm sys} = O(1/N_c)$. For a practical scheme of how to implement this perturbation on a specific example, see Sec. IV A. In our companion paper [31], the scaling analysis of the $1/N_c$ correction is shown to hold numerically with the continuous-time cloning algorithm (see Sec. II D 2). We also show that the $1/N_c$ correction behavior remains in fact valid at *finite time* [31], an open question that remains to be investigated analytically.

D. Remarks

Here, we discuss some remarks on the formulation presented in this section.

1. Relaxing the condition $dt = \Delta t$

In Sec. II A 2, we set the discretization time of the process dt to be equal to the time interval for cloning Δt, and we took the $dt = \Delta t \to 0$ limit at the end. We note that the condition $\Delta t = dt$ is not necessary if both limits $\Delta t \to 0$ and $dt \to 0$ (with $dt < \Delta t$) are taken at the end. This is practically important, because we can use the continuous-time *process* to perform the algorithm presented here by setting $dt = 0$ first, and $\Delta t \to 0$ limit afterwards. More precisely, replacing dt by Δt in the matrix $\mathcal{C}$ and $\mathcal{K}$, we build a new matrix $\mathcal{KC}(\mathcal{T}^{\Delta t/dt})$. Taking the $dt \to 0$ limit in this matrix while keeping Δt noninfinitesimal (but small), this matrix represents the population dynamics algorithm of a continuous-time process with a finite cloning time interval Δt. The arguments presented in this section can then be applied in the same way, replacing dt by Δt. We note that the deviation due to a noninfinitesimal Δt should thus appear as $O(\Delta t)$ [see Eq. (19), for example].

2. A continuous-time algorithm used in the companion paper

The $\Delta t \to 0$ limit is the key point in the formulation developed in this section. Thanks to this limit, upon each cloning step, the total number of clones $\sum_{j=0}^{M-1} n_j$ always varies only by ± 1, which makes the expression of the matrices $\mathcal{C}$ and $\mathcal{K}$ simple enough to develop the arguments presented in Secs. II B and II C. Furthermore, during the time interval Δt separating two cloning steps, the configuration is changing *at most once*. The process between cloning steps is thus simple, which allows us to represent the corresponding time-evolution matrix as $\mathcal{T}$ (by replacing dt by Δt as explained in Sec. II D 1 above). Generalizing our analytical study to a cloning dynamics in which the limit $\Delta t \to 0$ is not taken is therefore a very challenging task, which is out of the scope of this paper.

However, interestingly, in the companion to this paper [31] we observe numerically that our predictions for the finite-time and finite-population scalings are still valid in a different version of algorithm for which $\sum_{j=0}^{M-1} n_j$ can vary by an arbitrary amount—supporting the hypothesis that the analytical arguments that we present here could be extended to more general algorithms. More precisely, we use a continuous-time version of the algorithm [13] to study numerically an observable of type A [33]. This version of the algorithm differs from that considered in this paper, in the sense that the cloning steps are separated by *nonfixed* noninfinitesimal time intervals. These time intervals are distributed exponentially, in contrast to the fixed ones taken in here (where Δt is a constant). This results in an important difference: The effective interaction between copies due to the cloning and pruning procedure is unbounded (it can *a priori* affect any proportion of the population), while in the algorithm of the present paper, this effective interaction is restricted to a maximum of one cloning and pruning event in the $\Delta t \to 0$ limit. We stress that the $dt \to 0$ limit of the cloning algorithm studied here with a fixed Δt *does not yield the continuous-time cloning algorithm*, stressing that these two versions of the population dynamics present essential differences.

III. STOCHASTIC ERRORS: LARGE DEVIATIONS OF THE POPULATION DYNAMICS

In the previous section, we formulated the population dynamics algorithm as a birth-death process and evaluated the systematic errors (which are the deviation of the large deviation estimator from the correct value) due to a finite number of clones (Table III). In this section, we focus on stochastic errors corresponding to the run-to-run fluctuations of the large deviation estimator within the algorithm, at fixed N_c (see the introduction of this paper for the definition of the terms *stochastic errors* and *systematic errors*).

TABLE III. Magnitudes of the numerical errors.

	Magnitude of errors
Systematic errors	$O(1/N_c)$
Numerical errors	$O(1/(\tau N_c))$

In order to study stochastic errors, we formulate the large deviation principle of the large deviation estimator. In the population dynamics algorithm, the CGF estimator to measure is the time-average of the average cloning ratio of the population (see Sec. II B 1):

$$\psi_{N_c,\tau}(s) \equiv -s\frac{1}{\tau}\int_0^\tau dt\sum_{i=0}^{M-1}\frac{n_i(t)b_i}{N_c}. \tag{40}$$

As τ increases, this quantity converges to the expected value (which depends on N_c) with probability 1. However whenever we consider a finite τ, dynamical fluctuations are present, and there is a probability that this estimator deviates from its expected value. Since the population dynamics in the occupation states n is described by a Markov process, the probability of these deviations are themselves described by a large deviation principle [19,20]: By denoting by $\text{Prob}(\psi)$ the probability of $\psi_{N_c,\tau}(s)$, one has

$$\text{Prob}(\psi) \sim \exp\left[-\tau I_{N_c,s}(\psi)\right], \tag{41}$$

where $I_{N_c,s}(\psi)$ is a large deviation "rate function" (of the large deviation estimator). To study these large deviations, we can apply a standard technique using a biased evolution operator for our population dynamics: For a given Markov system, to calculate large deviations of additive quantities such as (40), one biases the time-evolution matrix with an exponential factor [19]. Specifically, by defining the following matrix

$$L^h_{\tilde{n},n} = \sigma(n\to\tilde{n}) - \delta_{\tilde{n},n}\sum_{n'}\sigma(n\to n') - hs\sum_{i=0}^{M-1}\frac{n_ib_i}{N_c} \tag{42}$$

and by denoting the largest eigenvalue of this matrix $G(h,s)$ [corresponding, as a function of h, to a scaled cumulant generating function for the observable (40)], the large deviation function $I_{N_c,s}(\psi)$ is obtained as the Legendre transform $\sup_h[h\psi - G(h,s)]$. In the companion paper [31], we show that a quadratic approximation of the rate function $I_{N_c,s}(\psi)$ (i.e., a Gaussian approximation) can be estimated directly from the cloning algorithm.

We consider the scaling properties of $I_{N_c,s}$ in the large-N_c limit. For this, we define a scaled variable $\tilde{h} \equiv h/N_c$ and a scaled function $\tilde{G}(\tilde{h},s) \equiv G(\tilde{h}N_c,s)/N_c$. If this scaled function $\tilde{G}(\tilde{h},s) \equiv G(\tilde{h}N_c,s)/N_c$ is well defined in the $N_c \to \infty$ limit (which is natural, as checked in the next paragraph), then we can derive that $I_{N_c,s}$ has the following scaling:

$$I_{N_c,s}(\psi) = N_c I_s(\psi) + o(N_c) \tag{43}$$

or equivalently,

$$\text{Prob}(\psi) \sim e^{-\tau N_c I_s(\psi)}, \tag{44}$$

where $I_s(\psi) = \max_{\tilde{h}}[\tilde{h}\psi - \tilde{G}(\tilde{h},s)]$. The scaling form (43) is validated numerically in Ref. [31]. From this large deviation principle, we can see that the stochastic errors of the large deviation estimator is of $O[1/(N_c\tau)]$ as shown in Table III.

In the largest eigenvalue problem for the transition matrix (42), by performing a system size expansion (see Sec. II C), we obtain

$$\begin{aligned}\tilde{G}(\tilde{h},s) = &\sum_{i,j(i\neq j)}\left(\frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_j}\right)x_iq(x)\\ &\times\left[w(i\to j) + s\tilde{w}^\infty_x(i\to j)\right]\\ &-\frac{\tilde{h}}{s}s\sum_i x_ib_iq(x) + O(1/N_c),\end{aligned} \tag{45}$$

where $q(x)$ is the right-eigenvector associated to the largest eigenvalue of $L^h_{\tilde{n},n}$ (represented as a function of $x \equiv n/N_c$). The first order of the right-hand side is of order $O(N_c^0)$, so that $\tilde{G}(\tilde{h},s)$ is also of order $O(N_c^0)$ in $N_c \to \infty$. (For an analytical example of the function $\tilde{G}(\tilde{h},s)$, see Sec. IV B.)

IV. EXAMPLE: A SIMPLE TWO-STATE MODEL

In this section, to illustrate the formulation that we developed in the previous sections, we consider a simple two-state model. In this system, the dimension of the state i is two ($M = 2$) and the transition rates $w(i \to j)$ are

$$w(0\to 1) = c, \tag{46}$$

$$w(1\to 0) = d, \tag{47}$$

with positive parameters c,d and $w(i \to i) = 0$. In this model, the quantity α_i defined in (9) becomes

$$\alpha_i = \delta_{i,0}\frac{n_1}{N_c}(b_1 - b_0) + \delta_{i,1}\frac{n_0}{N_c}(b_0 - b_1). \tag{48}$$

Hereafter, we assume that $b_1 > b_0$ without loss of generality. From this, the space $\Omega^{(\pm)}$ is determined as $\Omega^{(+)} = \{0\}$ and $\Omega^{(-)} = \{1\}$, which leads to the jump ratio $\tilde{w}_n(i \to j)$ as

$$\tilde{w}_n(i\to j) = \delta_{i,1}\delta_{j,0}\frac{n_0}{N_c}(b_1 - b_0)\left[\frac{n_1}{N_c+1} + \frac{n_0}{N_c-1}\right]. \tag{49}$$

Finally, from the conservation of the total population, $n_0 + n_1 = N_c$, we find that the state of the population n can be uniquely determined by specifying only the variable n_0. Thus the transition rate for the population dynamics is a function of n_0 (and $\tilde{n}_0$), $\sigma(n_0 \to \tilde{n}_0)$, which is derived as

$$\begin{aligned}\sigma(n_0\to\tilde{n}_0) = \delta_{\tilde{n}_0,n_0+1}\Bigg[&(N_c - n_0)d + k(n_0,N_c-n_0)\\ &\times\left(\frac{n_0}{N_c-1} + \frac{N_c-n_0}{N_c+1}\right)\Bigg] + \delta_{\tilde{n}_0,n_0-1}\, n_0\, c,\end{aligned} \tag{50}$$

where we have defined

$$k(n_0,n_1) = \frac{n_0n_1}{N_c}s[b_1 - b_0]. \tag{51}$$

A. Systematic errors

We first evaluate the systematic errors (see Sec. II C). For this, we consider the distribution function P^{st}_n. Since the system is described by a one-dimensional variable n_0 restricted to $0 \leqslant n_0 \leqslant N_c$, the transition rates $\sigma(n_0 \to \tilde{n}_0)$ satisfy the detailed balance condition

$$P^{\text{st}}_{n_0}\sigma(n_0\to n_0+1) = P^{\text{st}}_{n_0+1}\sigma(n_0+1\to n_0). \tag{52}$$

We can solve this equation exactly, but to illustrate the large-N_c limit, it is in fact sufficient to study the solution in an expansion $1/N_c \ll 1$. The result is

$$P^{\mathrm{st}}_{xN_c} = C\exp\left[-N_c I_{\mathrm{conf}}(x) + \delta I(x) + O(1/N_c)\right] \tag{53}$$

(with here $x \equiv n_0/N_c$), where, explicitly

$$I_{\mathrm{conf}}(x) = x + \log(1-x) - \frac{d\log\left[d+(b_1-b_0)sx\right]}{(b_1-b_0)s} - x\log\left[\frac{1}{cx}(1-x)(d+(b_1-b_0)sx)\right] \tag{54}$$

and

$$\delta I(x) = -x - \frac{2dx}{(b_1-b_0)s} + x^2 - \log x + \frac{2d^2\log[d+(b_1-b_0)sx]}{(b_1-b_0)^2 s^2} + \frac{d\log[d+(b_1-b_0)sx]}{(b_1-b_0)s}. \tag{55}$$

We now determine the value of x that minimizes $-N_c I_s(x) + \delta I(x)$, which leads to a finite-size correction (i.e., the systematic errors) of the population dynamics estimator. Indeed, denoting this optimal value of x by $x^*_{N_c}$, the large deviation estimator is obtained as

$$\psi_{N_c}(s) = -s\left[x^*_{N_c} b_0 + \left(1-x^*_{N_c}\right)b_1\right] \tag{56}$$

(see Sec. II B 1). From a straightforward calculation based on the expressions $I_{\mathrm{conf}}(x)$ and $\delta I(x)$, we obtain the expression of $x^*_{N_c}$ as

$$x^*_{N_c} = x^* + \frac{1}{N_c}\delta x^* + O[(1/N_c)^2], \tag{57}$$

with

$$x^* = \frac{-c-d+(b_1-b_0)s}{2(b_1-b_0)s} + \frac{\sqrt{4d(b_1-b_0)s + [-c-d+(b_1-b_0)s]^2}}{2(b_1-b_0)s} \tag{58}$$

and

$$\delta x^* = [2d + 2(b_1-b_0)sx^*]^{-1} \times \frac{2c\{-d-(b_1-b_0)sx^*[1+x^*-2(x^*)^2]\}}{\sqrt{4d(b_1-b_0)s + [c+d-(b_1-b_0)s]^2}}. \tag{59}$$

We thus arrive at

$$\psi(s) = \frac{-c-d-(b_1+b_0)s}{2} + \frac{\sqrt{4d(b_1-b_0)s + [-c-d+(b_1-b_0)s]^2}}{2} \tag{60}$$

and

$$\epsilon_{\mathrm{sys}} = \frac{1}{N_c}\frac{1}{|d+(b_1-b_0)sx^*|} \times \left|\frac{sc(b_0-b_1)[d+(b_0-b_1)s(x^*-1)x^*(1+2x^*)]}{\sqrt{4(b_1-b_0)ds + [c+d+(b_0-b_1)s]^2}}\right| \tag{61}$$

(see Eq. (35) for the definition of the systematic error ϵ_{sys}.) We check easily that the expression of $\psi(s)$ is the same as the one obtained from a standard method by solving the largest eigenvalue problem of a biased time-evolution operator (see, for example, Ref. [18]).

B. Stochastic errors

We now turn our attention to the stochastic errors. The scaled cumulant generating function $N_c\tilde{G}(\tilde{h},s)$ is the largest eigenvalue of a matrix $L^h_{\tilde{n},n}$ (see Eq. (42) and the explanations around it). We then recall a formula to calculate this largest eigenvalue problem from the following variational principle:

$$\tilde{G}(\tilde{h},s) = \sup_{\phi>0}\sum_n p_{\mathrm{st}}(n_0)\phi(n_0)^2 \times\left[\frac{\sigma(n\to n+1)}{N_c}\left(\frac{\phi(n_0+1)}{\phi(n_0)}-1\right) + \frac{\sigma(n\to n-1)}{N_c}\left(\frac{\phi(n_0-1)}{\phi(n_0)}-1\right) - s\tilde{h}\frac{\sum_i n_i b_i}{N_c^2}\right]. \tag{62}$$

(See, e.g., Appendix G of Ref. [34] or Ref. [33] for the derivation of this variational principle.) By following the usual route to solve such equations (see, e.g., Sec. 2.5 of Ref. [35]), we obtain

$$\tilde{G}(\tilde{h},s) = \sup_x\left[-\left(\sqrt{(1-x)[d+(b_1-b_0)sx]} - \sqrt{cx}\right)^2 - s\tilde{h}[xb_0 + (1-x)b_1]\right]. \tag{63}$$

Thus, $\tilde{G}(\tilde{h},s)$ is well defined, demonstrating that the large deviation principle (44) is satisfied. Furthermore, by expanding this variational principle with respect to $\tilde{h}$, we obtain

$$\tilde{G}(\tilde{h},s) = \psi(s)\tilde{h} + \frac{\kappa_s}{2}\tilde{h}^2 + O(\tilde{h}^3), \tag{64}$$

where $\psi(s)$ is given in (60), and the variance κ_s is given as

$$\kappa_s = c + \frac{cs(b_1-b_0)}{\sqrt{4(b_1-b_0)sd + [c+d+(b_0-b_1)s]^2}} - \frac{c(c+d)^2 + c(b_0-b_1)(c-3d)s}{c^2 + 2c[d+(b_0-b_1)s] + [d+(b_1-b_0)s]^2}. \tag{65}$$

We note that the expansion (64) is equivalent to the following expansion of the large deviation function $I_s(\psi)$ [see (44)] around the expected value $\psi(s)$:

$$I_s(\psi) = \frac{[\psi-\psi(s)]^2}{2\kappa_s} + O[(\psi-\psi_s)^3]. \tag{66}$$

The variance of the obtained large deviation estimator is thus $\kappa_s/(N_c\tau)$.

C. A different large deviation estimator

As an application of these exact expressions, we expand the systematic error ϵ_{sys} and the stochastic error (variance) κ_s with respect to s. A straightforward calculation leads to

$$\epsilon_{\mathrm{sys}} N_c = \left|\frac{2c(b_0-b_1)}{c+d}s\right| + O(s^2) \tag{67}$$

and

$$\kappa_s = \frac{2(b_0 - b_1)^2 cd}{(c+d)^3} s^2 + O(s^3). \tag{68}$$

We thus find that the first-order of the error $\epsilon_{\rm sys}$ scales as $O(s)$ at small s, but that the variance κ_s is of order $O(s^2)$. From this scaling, as we explain below, one can argue that the following large deviation estimator can be better than the standard one for small s:

$$\tilde{\Psi}(s) \equiv \frac{1}{\tau} \log \overline{\prod_{t=0}^{\tau/dt} \sum_i \frac{n_i(t) e^{-s dt b_i}}{N_c}}, \tag{69}$$

where the overline represents the averaging with respect to the realizations of the algorithm. [Normally, this realization average is taken *after* calculating the logarithm, which corresponds to the estimator (19).] Mathematically, this average [(69), before taking the logarithm] corresponds to a bias of the time-evolution matrix σ as seen in (42) for $h = 1$. This means that, in the limit $\tau \to \infty$ with a sufficiently large number of realizations, this averaged value behaves as $\tilde{\Psi}(s) \sim e^{\tau G(1,s)}$. By combining this result with the expansion (64), we thus obtain

$$\lim_{\tau\to\infty} \lim_{\substack{\text{many}\\ \text{realizations}}} \tilde{\Psi}(s) = \psi(s) + \frac{\kappa_s}{2} N_c^{-1} + O\left(N_c^{-2}\right) \tag{70}$$

(recalling $\tilde{G} = G/N_c$ and $\tilde{h} = h/N_c$). When we consider small s, by recalling $\epsilon_{\rm sys} N_c = O(s)$ and $\kappa_s = O(s^2)$, we thus find that the deviations from the correct value are smaller in the estimator $\tilde{\Psi}(s)$ than in the normal estimator given in (19), which comes as a surprise because in (69) the average and the logarithm are inverted with respect to a natural definition of the CGF estimator.

To use this estimator, we need to discuss the two following points. First, since the scaled cumulant generating function $G(1,s)$ has small fluctuations, one needs a very large number of realizations in order to attain the equality (70). The difficulty of this measurement is the same level as the one of direct observations of a large deviation function; see, for example, Ref. [32]. However, we stress that this point may not be fatal in this estimator, because we do not need to attain completely this equality; i.e., our aim is the zeroth-order coefficient, $\psi(s)$, in (70). Second, we have not proved yet the scaling properties with respect to s, which are $\epsilon_{\rm sys} N_c = O(s)$ and $\kappa_s = O(s^2)$, in a general setup aside from this simple two-state model. We show in practice in Ref. [31] that for small values of s, the estimator (69) is affected by smaller systematic errors, in the numerical study of the creation-annihilation process studied in this section. We will focus on the generality of our observations on these points in a future study.

V. DISCUSSION

In this paper, we formulated a birth-death process that describes population dynamics algorithms and aims at evaluating numerically large deviation functions. We derived that this birth-death process leads generically to the correct large deviation results in the large limit of the number of clones $N_c \to \infty$. From this formulation, we also derived that the deviation of large deviation estimator from the desired value (which we called systematic errors) is small and proportional with $O(N_c^{-1})$. Below, based on this observation, we propose a simple interpolation technique to improve the numerical estimation of large deviation functions in practical uses of the algorithm.

A. An interpolation technique using the $O(1/N_c)$ scaling of the systematic error

Imagine that we now apply the population dynamics algorithm to a given system. We need to carefully consider the asymptotic limit of the two large parameters τ and N_c in the convergence of the large deviation estimator (40). Indeed, what one needs to do in this simulation is (i) take the large-τ limit for a fixed N_c and estimate the $\tau \to \infty$ value of the estimator for this fixed N_c, then (ii) estimate this large-τ value for several (and increasing) N_c, and finally estimate large-τ-N_c limit value. This is different from standard Monte Carlo simulations, where one needs to consider only the large-τ limit, thanks to ergodicity.

Any method that can make the LDF estimation easier thus will be appreciated. Based on our observations, we know that the second part [(ii) above] converges with an error proportional to $1/N_c$. Also, from the large deviation estimator (40), one can easily see that the convergence speed with respect to τ for a fixed N_c is proportional to $1/\tau$ (i.e., the first part [(i) above] converges proportionally to $1/\tau$). By using these $1/\tau$ and $1/N_c$ scalings for (i) and (ii), one can interpolate the large-τ and large-N_c asymptotic value of the LDF estimator from the measured values for finite τ and N_c. We introduce this numerical method in practice in the companion paper [31]. We demonstrate numerically that the interpolation technique is very efficient in practice, by a direct comparison of the resulting estimation of the CGF to its analytical value, which is also available in the studied system. We also stress that it is developed for a different cloning algorithm by using a continuous-time population dynamics [13] (see Sec. II D 2 for the description of the conceptual difference). From these results, we conjecture that the validity of the large-τ and large-N_c scalings is very general and independent of the details of the algorithm.

B. Open questions

We mention two open questions. The first question is about the precise estimate of the error due to a noninfinitesimal time interval Δt between cloning steps: As explained in Secs. II D 1 and II D 2, taking the $\Delta t \to 0$ limit is important in our analysis, in order to make the estimator converge to the correct LDF. The error due to noninfinitesimal Δt is at most of order Δt as seen from Eq. (19) (see also Sec. II D 1). From a practical point of view, taking this limit can, however, be problematic, since it requires infinitely many cloning procedures per unit time (as $\Delta t \to 0$). Interestingly, most of existing algorithms do not take such a limit (see, for instance, the original version of the algorithm [12]). Empirically, one thus expects that the error goes to zero as $N_c \to \infty$ while keeping Δt finite. Within the method developed in this paper, the analytical estimation of this error is challenging (see Sec. II D 2) and remains an open problem, but for example, one can approach to this issue numerically at least.

The second question is about possible extensions of the formulation developed in this paper. In our algorithm, we perform a cloning procedure for a fixed time interval, which means that our formulation cannot cover the case of algorithms where Δt itself is statistically distributed, as in continuous-time cloning algorithms [13]. Moreover, our formulation is limited to Markov systems, although population dynamics algorithms are applied to chaotic deterministic dynamics [14,17] or to non-Markovian evolutions [36]. Once one removes the Markov condition in the dynamics, developing analytical approaches becomes more challenging. However, as the physics of those systems are important scientifically and industrially [37], the understanding of such dynamics cannot be avoided for the further development of population algorithms.

ACKNOWLEDGMENTS

T.N. gratefully acknowledges the support of Fondation Sciences Mathématiques de Paris – EOTP NEMOT15RPO, PEPS LABS, and LAABS Inphyniti CNRS project. E.G. thanks Khashayar Pakdaman for his support and discussions. Special thanks go to the Ecuadorian Government and the Secretaría Nacional de Educación Superior, Ciencia, Tecnología e Innovación, SENESCYT, for support. V.L. acknowledges support by the National Science Foundation under Grant No. NSF PHY11-25915 during a stay at KITP, UCSB, and support by the ANR-15-CE40-0020-03 Grant LSD. V.L. acknowledges partial support by the ERC Starting Grant No. 680275 MALIG. T.N. and V.L. are grateful to B. Derrida and S. Shiri for discussions. We are grateful to S. Shiri for discussions about the first paragraph of Sec. V B (open questions).

[1] J. B. Anderson, J. Chem. Phys. **63**, 1499 (1975).
[2] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic, *Proceedings of the 28th Conference on Winter Simulation, WSC '96* (IEEE Computer Society, Washington, DC, 1996), pp. 302–308.
[3] Y. Iba, Trans. Japan. Soc. Artificial Intelligence **16**, 279 (2001).
[4] P. Grassberger, Comput. Phys. Commun. **147**, 64 (2002).
[5] P. L'Ecuyer, V. Demers, and B. Tuffin, in *Proceedings of the 2006 Winter Simulation Conference* (IEEE, 2006), pp. 137–148.
[6] O. Cappé, A. Guillin, J. M. Marin, and C. P. Robert, J. Comput. Graph. Stat. **13**, 907 (2004).
[7] T. S. van Erp, D. Moroni, and P. G. Bolhuis, J. Chem. Phys. **118**, 7762 (2003).
[8] R. J. Allen, P. B. Warren, and P. R. ten Wolde, Phys. Rev. Lett. **94**, 018104 (2005).
[9] P. Del Moral and J. Garnier, Ann. Appl. Probab. **15**, 2496 (2005).
[10] T. Dean and P. Dupuis, Stoch. Process. Appl. **119**, 562 (2009).
[11] C. Frédéric and G. Arnaud, Stoch. Anal. Appl. **25**, 417 (2007).
[12] C. Giardinà, J. Kurchan, and L. Peliti, Phys. Rev. Lett. **96**, 120603 (2006).
[13] V. Lecomte and J. Tailleur, J. Stat. Mech. (2007) P03004.
[14] J. Tailleur and J. Kurchan, Nat. Phys. **3**, 203 (2007).
[15] J. Tailleur and V. Lecomte, AIP Conf. Proc. **1091**, 212 (2009).
[16] C. Giardinà, J. Kurchan, V. Lecomte, and J. Tailleur, J. Stat. Phys. **145**, 787 (2011).
[17] T. Laffargue, K.-D. N. T. Lam, J. Kurchan, and J. Tailleur, J. Phys. A: Math. Theor. **46**, 254002 (2013).
[18] E. Guevara Hidalgo and V. Lecomte, J. Phys. A: Math. Theor. **49**, 205002 (2016).
[19] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*, Applications of Mathematics (Springer, New York, 1998).
[20] H. Touchette, Phys. Rep. **478**, 1 (2009).
[21] J. P. Garrahan, R. L. Jack, V. Lecomte, E. Pitard, K. van Duijvendijk, and F. van Wijland, Phys. Rev. Lett. **98**, 195702 (2007).
[22] T. Bodineau and B. Derrida, Phys. Rev. Lett. **92**, 180601 (2004).
[23] P. I. Hurtado and P. L. Garrido, Phys. Rev. E **81**, 041102 (2010).
[24] T. Nemoto, F. Bouchet, R. L. Jack, and V. Lecomte, Phys. Rev. E **93**, 062123 (2016).
[25] N. V. Kampen (ed.), *Stochastic Processes in Physics and Chemistry*, 3rd ed. (Elsevier, Amsterdam, 2007), p. iii.
[26] C. W. Gardiner, *Handbook of Stochastic Methods: For Physics, Chemistry, and the Natural Sciences*, Springer Series in Synergetics (Springer, Berlin, 1983), includes indexes.
[27] R. L. Jack and P. Sollich, Prog. Theor. Phys. Supplement **184**, 304 (2010).
[28] V. Popkov, G. M. Schütz, and D. Simon, J. Stat. Mech. (2010) P10007.
[29] R. Chetrite and H. Touchette, Phys. Rev. Lett. **111**, 120601 (2013).
[30] T. Nemoto and S.-i. Sasa, Phys. Rev. Lett. **112**, 090602 (2014).
[31] E. Guevara Hidalgo, T. Nemoto, and V. Lecomte, arXiv:1607.08804 [cond-mat].
[32] C. M. Rohwer, F. Angeletti, and H. Touchette, Phys. Rev. E **92**, 052104 (2015).
[33] J. P. Garrahan, R. L. Jack, V. Lecomte, E. Pitard, K. van Duijvendijk, and F. van Wijland, J. Phys. A **42**, 075007 (2009).
[34] T. Nemoto and S.-i. Sasa, Phys. Rev. E **84**, 061113 (2011).
[35] T. Nemoto, V. Lecomte, S. Sasa, and F. van Wijland, J. Stat. Mech. (2014) P10001.
[36] M. Cavallaro and R. J. Harris, J. Phys. A: Math. Theor. **49**, 47LT02 (2016).
[37] J. S. Wettlaufer, Phys. Rev. Lett. **116**, 150002 (2016).