Steady-state simulation of metastable stochastic chemical systems
Andreas Milias-Argeitis, and John Lygeros

Citation: The Journal of Chemical Physics 138, 184109 (2013); doi: 10.1063/1.4804191
View online: https://doi.org/10.1063/1.4804191
View Table of Contents: http://aip.scitation.org/toc/jcp/138/18
Published by the American Institute of Physics

Articles you may be interested in
Markov models of molecular kinetics: Generation and validation
The Journal of Chemical Physics 134, 174105 (2011); 10.1063/1.3565032

Steady-state simulations using weighted ensemble path sampling
The Journal of Chemical Physics 133, 014110 (2010); 10.1063/1.3456985

Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the Gillespie algorithm
The Journal of Chemical Physics 118, 4999 (2003); 10.1063/1.1545446

The finite state projection algorithm for the solution of the chemical master equation
The Journal of Chemical Physics 124, 044104 (2006); 10.1063/1.2145882

Stochastic chemical kinetics and the total quasi-steady-state assumption: Application to the stochastic simulation algorithm and chemical master equation
The Journal of Chemical Physics 129, 095105 (2008); 10.1063/1.2971036

Refining the weighted stochastic simulation algorithm
The Journal of Chemical Physics 130, 174103 (2009); 10.1063/1.3116791
I. INTRODUCTION

Well-stirred chemical systems at thermal equilibrium can be modeled by continuous-time Markov chains evolving in a countable space\(^1\) (usually \(\mathbb{Z}^{n}_{\geq 0}\)), whose probability density evolves according to the Chemical Master Equation (CME).\(^{2,3}\) In this work, we consider Markov chains that satisfy the necessary and sufficient conditions for having a unique invariant distribution, which corresponds to the stationary solution of the CME. Study of the stationary distributions of these systems can give useful insights about how they function and the robustness of their stationary behavior to parameter variations. It can also point to the existence of emergent behavior such as metastability, which is the object of this work. Unfortunately, except for special cases, direct calculation of the stationary distribution from the CME is impossible and one has to resort to approximation techniques, either analytical or numerical.

One standard idea for the numerical approximation of the stationary distribution is state-space truncation, whereby one calculates the stationary distribution of large (but finite) Markov chain that in some sense approximates the original one. Under well-characterized mathematical conditions, the truncated chain “approaches” the original as the truncation grows, and the same happens to their invariant distributions.\(^4\) In a similar fashion, the finite state projection algorithm\(^5\) can provide an approximation to the distribution of the system over any finite time horizon within a desired error tolerance. Provided the time horizon and system truncation are large enough, an estimate of the stationary distribution can thus be obtained. However, in both cases the necessary truncation size is difficult to determine. Moreover, the truncations necessary to obtain a close approximation are usually too large for the invariant distribution to be computed. Another alternative is stochastic simulation, which is always possible\(^6\) and, given the ergodicity assumption, is guaranteed to give correct results as the simulation length increases. Often, however, the computation time is too long, as systems may evolve slowly towards equilibrium. One cause for the slow convergence is the coexistence of two or more different time scales in the system transition rates, which lead some species to equilibrate much faster than others. The way out of this situation is the use of quasi-steady-state approximation techniques that treat the two time scales separately.\(^7\)

Unfortunately, not all slowly converging systems can be treated with quasi steady-state approximations. In some cases the whole Markov chain may evolve quickly towards a state of “pseudo-equilibrium” (which may even depend on the initial condition), and get trapped in a region of the state space for a large amount of time. Viewed at a longer time scale, however, the system is not globally in equilibrium because it makes rare transitions from one pseudo-equilibrium to another. This behavior is translated into the existence of two or more modes in the invariant distribution, separated by areas of very low probability. Such systems are called metastable,\(^8,9\) because they spend a large amount of time around each mode, making rare fluctuation-driven transitions between the mode regions. This is the case for bistable switch-like systems, for example, that are frequently encountered in the systems biology literature. Metastable behavior frequently underlies fundamental decision mechanisms of the cell. The different metastable regions usually correspond to distinct cellular phenotypes, which the cells assume with a certain probability, depending on environmental queues. In some cases, random switching among phenotypes might be beneficial for the survival of the cell,\(^10\) while in others the decisions have to be firmly maintained over time, despite random fluctuations that tend to perturb the system out of equilibrium.\(^11\) Study of the metastable behavior of a biochemical model is thus an important step towards understanding noise exploitation and control mechanisms in the cells.

---

\(^{a)}\)Electronic mail: milias@control.ee.ethz.ch
Given that the convergence to equilibrium of metastable systems can be extremely slow, powerful rare event simulation techniques, such as Non-Equilibrium Umbrella Sampling (NEUS)\textsuperscript{12} and Forward Flux Sampling (FFS)\textsuperscript{13} have to be used to simulate them efficiently. They both employ a discretization of an order-parameter space, and stochastic simulations are performed separately in the resulting regions. For the calculation of the full steady-state distribution, one needs to know the weight of each region, which is obtained in different ways in the two methods. Trajectories that attempt to exit the regions are reinitialized according to a distribution of flux entry points, and both weights and fluxes are computed iteratively.\textsuperscript{14} Contrary to the simplicity of Stochastic Simulation Algorithm (SSA), the efficient implementation and tuning of these algorithms is non-trivial.

In this work, we propose a simpler method for calculating the stationary distribution of metastable systems based on the theory of stochastic complement.\textsuperscript{15} We show how one can simulate the Markov chain on a given subset of the state space, by “masking out” transitions to states outside this set so that the true (conditional) invariant distribution over this subset is preserved. To exploit this property, we propose a method that divides the system state space into different parts (namely the metastable and the transition regions), approximates the invariant distribution of the chain conditioned on being inside a certain region, and combines the results from individual regions to obtain the full stationary distribution. Preliminary results in this direction were presented in Ref. 16. Our method achieves a significant reduction in computation cost compared to SSA, thus expanding the range of computationally tractable systems. On the other hand, it offers a simple and intuitive implementation, that can be easily tuned to achieve high efficiency.

II. INTRODUCTION TO STOCHASTIC COMPLEMENT THEORY

(Notational conventions: All vectors are assumed to be row vectors and $e_i$ denotes the unit vector with 1 at the $i$th position. Let also $|| \cdot ||$ denote the norm of $\ell^1$ and $\mathbf{1}$ denote a vector of ones.)

Consider an ergodic.\textsuperscript{17} Markov chain $\{X_t, t \geq 0\}$ with generator matrix $Q$ over a countable set $\mathcal{S} \subseteq \mathbb{Z}_{\geq 0}$, with a (unique) invariant distribution denoted by $\pi$. It is known that the evolution of the probability distribution of the chain, $p$, is described by the Kolmogorov forward equation\textsuperscript{17} (also known as the Chemical Master Equation\textsuperscript{1} in the fields of physics and chemistry),

$$\frac{dp}{dt} = pQ.$$ 

Since the chain is ergodic, we know that it possesses a unique invariant distribution, $\pi$, which satisfies

$$\pi Q = 0.$$ 

Assume that $\mathcal{S}$ has been partitioned into 2 disjoint subsets, $\mathcal{B}$ and $\mathcal{B}^c$ (where $\mathcal{A}^c$ denotes the complement of set $\mathcal{A}$ in $\mathcal{S}$) such that $\mathcal{B} \cup \mathcal{B}^c = \mathcal{S}$, and consider the corresponding partition of $Q$,

$$Q = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix},$$

where $Q_{11}$ contains the transition rates from $\mathcal{B}$ to $\mathcal{B}$, $Q_{22}$ contains transitions from $\mathcal{B}^c$ to $\mathcal{B}^c$, while $Q_{12}$ and $Q_{21}$ contain the transition rates from one set to the other. Finally, let $P$ denote the transition matrix of the embedded Markov chain (Ch. 2 of Ref. 17) associated with $X$, partitioned in the same way as $Q$ and following the same notation for its blocks. We know that $P = I + D \cdot Q$, where $D$ is a diagonal matrix whose $i$th diagonal entry is equal to $1/|q_{ii}|$.

The censored (or watched) Markov chain $\{X_t^\nu, t \geq 0\}$ on $\mathcal{B}$ is loosely defined as the process whose sample paths are obtained from the sample paths of $\{X_t, t \geq 0\}$ by deleting the parts outside $\mathcal{B}$, i.e., by watching $X^\nu_t$ only when it is in $\mathcal{B}$. This means that a transition $b_1 \rightarrow b_2$ in $\mathcal{B}$ can correspond either to a direct path $b_1 \rightarrow b_2$ or a detour $b_1 \rightarrow b^c_2 \rightarrow b_2$ in the full chain. For a rigorous definition, see Ref. 18. The censored process on $\mathcal{B}$ is again a Markov chain,\textsuperscript{18,19} whose generator is given by

$$R_{11} = Q_{11} - Q_{12} \left( \sum_{n=0}^{\infty} P_{22}^n \right) D_{22} Q_{21},$$

where $P_{22}$ and $D_{22}$ are submatrices of $P$ and $D$ defined in the same way as $Q_{22}$ (the closely related definition for discrete-time Markov chains is given in Refs. 19 and 22). Matrix $R_{11}$ is called the stochastic complement of $Q_{11}$ in $Q$, a name given by Meyer\textsuperscript{15} due to its close connection to the Schur complement. It can be shown that the stochastic complement defines an irreducible Markov chain, provided the full chain is irreducible.\textsuperscript{19,21,22} The following facts about stochastic complements\textsuperscript{20-22} form the basis of our approach:

**Theorem 2.1.** Assume that the full chain is ergodic with stationary distribution $\pi$ and write $\pi = (\pi^{(1)}, \pi^{(2)})$, according to the state space partition. Then the vector

$$v_1 = \frac{\pi^{(1)}}{\pi^{(1)} \cdot 1^T}$$

is the unique stationary distribution of the censored chain on $\mathcal{B}$ (i.e., $v_1 R_{11} = 0$).

Moreover, the stationary distribution of $Q$ is given by

$$\pi = (\xi_1 v_1, \xi_2 v_2),$$

where $v_2 = \pi^{(2)}/(\pi^{(2)} \cdot 1^T)$ and the vector $\xi = (\xi_1, \xi_2)$ is the invariant distribution of the $2 \times 2$ generator matrix $C$ with entries given by

$$c_{ij} = v_i Q_{ij} 1^T.$$
constant $\xi$, is called the coupling factor, is given by

$$\xi = \sum_{b_j \in A} \pi(b_j),$$

and is equal to the fraction of time the chain spends in $A$ at stationarity. Matrix $C$ in Theorem 2.1 can be also used to define the so-called aggregated Markov chain, that shares several important properties with the full chain. A closer look at the definition of $C$ shows that $c_{12}$ is proportional to the probability flux entering $B^c$ from $B$ at stationarity ($c_{21}$ being the flux in the opposite direction), while $c_{11}$ ($c_{22}$) is proportional to the flux leaving $B$ ($B^c$) at stationarity. The condition $\xi \cdot C = 0$ is then just an expression of the balance of fluxes between the two sets at equilibrium.

All definitions above can be easily extended when $S$ is partitioned into any finite number of subsets, however for our purposes it suffices to consider a partition consisting of two sets.

III. SIMULATION ALGORITHM

We first make a key observation, based on (1), that if each state in $S$ leads to finitely many states and $B^c$ is finite (which implies that $B$ is infinite), then $R_{11}$ can be obtained by modifying $Q_{11}$ at finitely many points. This can be easily seen by observing that $Q_{12}$ and $Q_{21}$ have a finite number of columns and rows, respectively, while $Q_{22}$ is a finite matrix. It is thus possible to calculate any finite number of elements in $R_{11}$ and use them for simulation purposes, as will be explained below. Moreover, given that the finite substochastic matrix $P_{22}$ satisfies $\|P_{22}\| < 1$, the infinite sum in (1) converges to

$$(I - P_{22})^{-1} = (I - I - D_{22} Q_{22})^{-1} = -Q_{22}^{-1} D_{22}^{-1}$$

(Ch. 7 of Ref. 24). We then have

$$R_{11} = Q_{11} - Q_{12} Q_{22}^{-1} Q_{21}. \quad (5)$$

Submatrix $Q_{22}$ can be thought of as the generator of an absorbing Markov chain on the finite set $B^c$, in which case the $(m, n)$ element of $-Q_{22}^{-1}$ is equal to the mean time spent at the $n$-th state of $B^c$ starting from the $m$-th state before absorption (into $B$) occurs (Ch. 8.6 of Ref. 25 and Ref. 26).

A. Setup

The proper definition of a metastable set is quite technical and not useful for the practical purposes of this work (more details can be found in Refs. 8 and 9). However, we intuitively understand metastable sets as the separated regions of the state space in which the Markov chain spends most of its time, while making rare transitions among them. Generalization of the proposed algorithm to more than two metastable sets is straightforward and hence omitted due to its notational complexity. We therefore consider a metastable Markov chain with two finite and disjoint metastable sets, $B_1$ and $B_2$. We further define $B_3 = (B_1 \cup B_2)^c$ and assume that no transition between $B_1$ to $B_2$ is possible in one step. By appropriately enumerating the states of $S$, the generator of this Markov chain can then be written as

$$Q = \begin{bmatrix}
    Q_{11} & Q_{12} & Q_{13} \\
    Q_{21} & Q_{22} & Q_{23} \\
    Q_{31} & Q_{32} & Q_{33}
\end{bmatrix} = \begin{bmatrix}
    Q_{11} & 0 & Q_{13} \\
    0 & Q_{22} & Q_{23} \\
    Q_{31} & Q_{32} & Q_{33}
\end{bmatrix},$$

where index 1 refers to states in $B_1$, index 2 corresponds to $B_2$ and index 3 to $B_3$. The stochastic complement $R_{33}$ is then given by

$$R_{33} = Q_{33} - \begin{bmatrix}
    Q_{31} & Q_{32}
\end{bmatrix} \begin{bmatrix}
    Q_{11}^{-1} & 0 \\
    0 & Q_{22}^{-1}
\end{bmatrix} \begin{bmatrix}
    Q_{13} \\
    Q_{23}
\end{bmatrix} = Q_{33} - Q_{31} Q_{11}^{-1} Q_{13} - Q_{32} Q_{22}^{-1} Q_{23}. \quad (6)$$

Finally, let $B_3^{3 \rightarrow 1}$ ($B_3^{3 \rightarrow 2}$) denote the subset of $B_3$ from which a transition to $B_1$ ($B_2$) is possible in one step. We also consider $B_3^{3 \rightarrow 1} \subset B_1$ and $B_3^{3 \rightarrow 2} \subset B_2$, the sets of points in $B_1$ and $B_2$ that are accessible from $B_3$ as well as $B_3^{1 \rightarrow 3} \subset B_1$ and $B_3^{2 \rightarrow 3} \subset B_2$, the sets from which $B_3$ can be accessed in one step.

B. First step

The first step of our algorithm is the simulation of a Markov chain with generator $R_{33}$. This chain evolves in $B_3$, a set of states visited by the full Markov chain only very rarely, during its excursions out of one metastable region (which may result in a transition to another region, or not). The use of the stochastic complement thus enables us to sample this region much more efficiently than by SSA alone, and forms the basis of our approach.

More concretely, the reduced Markov chain on $B_3$ evolves exactly like the full Markov chain in $B_3 \setminus (B_3^{3 \rightarrow 1} \cup B_3^{3 \rightarrow 2})$, until it reaches $B_3^{3 \rightarrow 1}$ or $B_3^{3 \rightarrow 2}$. Due to the structure of $R_{33}$, the transition rates for these states are modified according to the second or third term in (6). In case $B_3^{3 \rightarrow 1}$ and $B_3^{3 \rightarrow 2}$ overlap, both terms have to be taken into account. The modified transition rates can be calculated from the same formula, and stored prior to the simulation of the chain, to maximize the efficiency of the simulator. Note that $Q_{11}^{-1}$ and $Q_{22}^{-1}$, which are typically full matrices even if $Q$ is sparse, are only used to compute the modified transition rates and are not needed during the simulation. Moreover, the second and third terms in (6) can be calculated using linear algebraic techniques that avoid actually calculating and storing matrix inverses.

The advantage of using the stochastic complement comes from the definition of the censored Markov chain given in Sec. II: every time the full chain hits a state in $B_3^{3 \rightarrow 1}$ or $B_3^{3 \rightarrow 2}$, it can exit $B_3$ with a certain probability and return to it through some other state. These transitions are “masked out” in the censored chain, which thus seems to make jumps as it evolves in $B_3$. Since the full chain spends most of its time inside the metastable sets, by “fast-forwarding” its transitions to the moment it returns to $B_3$, we can make large leaps forward in time. Moreover, we know that $\nu_3$, differs from $\pi(3)$ only by a (yet unknown) factor.

Thus, the first step of our simulation algorithm simply requires the offline calculation and the storage of the
modified transition rates for the states in \( B_3^{3-1} \) and \( B_3^{3-2} \) and the simulation of the reduced Markov chain on \( B_3 \).

C. Second step

After the approximation of \( v_3 \) via simulation, the second step consists of the calculation of \( v_1 \) and \( v_2 \), the conditional stationary distributions on \( B_1 \) and \( B_2 \). From the Chemical Master Equation, we know that for each metastable set \( B_i \) (\( i = 1, 2 \)), it holds that

\[
\frac{dp_i}{dt} = p_i Q_{ii} + f_i(t),
\]

where \( f_i \) denotes the incoming probability flux from states outside \( B_i \) and \( p_i \) is the vector of probabilities for states in \( B_i \). Both row vectors are defined according to the same enumeration of states used to define \( Q_{ii} \). At stationarity, the outgoing flux from \( B_i \) has to be equal and opposite to the incoming flux, which implies that the two terms on the right-hand side of (7) cancel each other. Defining \( \pi_i = \lim_{t \to \infty} p_i(t) \) and \( f_{i}^{ss} = \lim_{t \to \infty} f_i(t) \), this requirement translates into

\[
\pi_i Q_{ii} = -f_{i}^{ss},
\]

which is a system of linear equations. Using the approximation of \( v_3 \) obtained from the previous step, we can compute a vector proportional to \( f_{i}^{ss} \), which we denote by \( \tilde{f}_{i}^{ss} \), by exploiting the fact that

\[
\tilde{f}_{i}^{ss} = v_3 Q_{3i}.
\]

Combining (8) and (9), we finally arrive at \( v_1 \) and \( v_2 \),

\[
v_1 = \frac{-v_3 Q_{3i} Q_{1i}^{-1}}{\| v_3 Q_{3i} Q_{1i}^{-1} \|_1},
\]

\[
v_2 = \frac{-v_3 Q_{3i} Q_{2i}^{-1}}{\| v_3 Q_{3i} Q_{2i}^{-1} \|_1}.
\]

D. Third step

The last step consists of the coupling factor calculation. The coupling factors enable us to weigh the distributions computed in steps 1 and 2 correctly against each other, and produce the unconditional stationary distribution over the whole space. Having \( v_1 \), \( v_2 \), and \( v_3 \) from the previous steps, this calculation is simply an application of Theorem 2.1.

IV. PRACTICAL IMPLEMENTATION

For the successful application of the method, the first step plays the most crucial role, as all subsequent results depend on the quality of the approximation of \( v_3 \). We therefore provide here some more remarks about the practical implementation of this step, which can help improve this approximation in a computationally efficient way.

A. Definition of \( B_1 \) and \( B_2 \)

Metastable behavior of a given Markov chain can be easily observed by simulation: starting from different initial conditions, the chain quickly evolves towards different regions of the state space, and remains there for large amounts of time. Depending on the relative stability of these sets, the chain may transit from one region to another during the simulation horizon, or spend the whole time in one of them. Before applying our proposed method, these regions have to be located in the state space. This information can be obtained directly from the sample paths of the chain: one simply needs to record the states visited within each metastable region, and either use them directly to define \( B_1 \) and \( B_2 \), or approximate the set by a simple geometric shape (e.g., a box). The latter case is computationally more convenient for subsequent simulations in \( B_1 \), as it is much simpler to test whether a point belongs to a set when an algebraic description of this set is available. However, geometric approximation may assign unnecessarily many states to each metastable set, most of which carry very small amounts of the invariant mass. When the chain evolves in a high dimensional space, this over approximation may result in matrices that are too large to be handled or even stored.

In such cases, it is thus much more efficient to follow the first alternative and define the metastable sets directly through simulation. However, the definition of these sets is not strict and the chain sample paths may frequently deviate far from the “centers” of the metastable regions, without however making a transition. Therefore, one has to ensure that the sets defined through this procedure are disjoint and, preferably, that \( B_3^{3-1} \cap B_3^{3-2} = \emptyset \), so that each metastable set can be handled separately. This condition can be satisfied by removing the conflicting states from the sets, a procedure that should remove only a small number of states, usually located on the outer edges of the metastable regions. One can also ensure that the remaining states in \( B_1 \) and \( B_2 \) communicate by forming the adjacency matrix of the resulting graph in the state space after the removal, and keeping only the largest strongly connected component of the graph. This last step is not necessary, but it facilitates the presentation of the resulting distributions.

Note that the internal structure of these sets (determined, for example, by correlations between different coordinates of the chain) is such that they typically contain relatively few states (e.g., much fewer than what a crude box over approximation would yield), even in the case of high-dimensional systems.

B. Matrix calculations and recursive calculations

The computational bottleneck of our proposed algorithm lies in the calculation of the stochastic complement \( R_{33} \), which involves calculations with matrix inverses. The typical complexity an \( n \times n \) matrix inversion is \( O(n^3) \), while the storage space for the inverse grows with \( n^2 \). When more than a few tens of thousands of states are contained in each metastable set, inversion of \( Q_{11} \) and \( Q_{22} \) becomes very time consuming, and storage of the inverses requires too large amounts
of memory. In case adequate computational resources are available, the inversion can be easily parallelized, by solving a set of $n$ linear equations of the form $xQ_{ii} = e_j$, which compute each row of $Q_{ii}^{-1}$ independently. In general, however, explicit matrix inversions can (and should) be avoided by using linear algebraic numerical techniques for solving large (and, in our case, sparse) systems of linear equations.

The problem of calculating the complements of very large finite subsets can be alternatively addressed by noting that stochastic complementation can be applied recursively in a very simple manner: starting from the full chain, one can proceed to define a reduced chain by removing a finite subset of the state space, $A_1$, and calculating the modified transition rates in the stochastic complement. The same procedure can be applied to the new chain, and the removal of another finite subset, $A_2$, will lead to a further reduced chain. It is then easy to see that the removal of $A_1 \cup A_2$ from the full chain in one step results in the same generator matrix with the stepwise reduced chain (more details are given in Appendix A). It is thus possible to remove very large sets of the original state space in an iterative fashion, which alleviates the computational problems described above.

C. Estimation of $\nu_1$ and $\nu_2$

The conditional stationary distributions on the metastable sets can also be obtained by means other than linear algebra. The most obvious alternative is (again) stochastic simulation, whereby one can use the stationary incoming flux distributions on the region boundaries to initialize trajectory segments that evolve inside each region and are killed whenever they attempt to exit. In case the metastable regions are too large for either the analytic or the simulation alternative, stochastic complements can be defined iteratively inside each region and the first step of our algorithm can be repeated for the censored Markov chains defined on the metastable regions. Exact stochastic simulation of these Markov chains is possible thanks to the knowledge of the boundary flux statistics, which enable the generation of correctly weighted trajectory segments inside each metastable region.

D. Dominant eigenvalue

One can determine a priori the stability of the selected metastable sets, if the sizes of $Q_{11}$ and $Q_{22}$ permit the calculation of their largest eigenvalues. From the theory of quasi-stationary distributions,27,28 we know that the largest eigenvalue of the generator of an irreducible absorbing Markov chain determines the decay rate of the probability of non-absorption over time. Provided the second largest eigenvalue is significantly smaller than the first (e.g., more than an order of magnitude), this so-called spectral gap guarantees that the distributions of the exit time from the metastable set is approximately exponential. The largest eigenvalue thus determines the average amount of time that the chain will spend inside a metastable set before exiting for the first time, and can be used to judge the relative stability of the different metastable sets in the chain, as well as which states have to be included in the sets to increase their stability.

V. EXAMPLES

We present the application of our method to two different bistable systems. The first is two-dimensional and is based on the genetic toggle switch model of Gardner,30 while the second is a five-dimensional model that serves as an abstraction of the more complicated λ-phage switch.31

All calculations were performed on a 2.6 GHz CPU with 24 GB of RAM (the maximal amount of memory used in all our simulations was around 2.5GB) using custom-written MATLAB scripts. Where parallel calculations were possible (e.g., where multiple simulation runs were performed), they were performed on 20 nodes of a computer cluster, each with a 2.6 GHz CPU and 3 GB of allocated memory.

A. 2D toggle switch

The chemical reactions for the 2D switch represent the evolution of the products, $x$ and $y$, of two genes $X$ and $Y$. The four reactions involved and their corresponding propensities are given below,

$$R_1 : \emptyset \rightarrow x, \lambda_1 = \frac{k_1}{1 + y^{n_1}}.$$  
$$R_2 : x \rightarrow \emptyset, \lambda_2 = x,$$  
$$R_3 : \emptyset \rightarrow y, \lambda_3 = \frac{k_2}{1 + x^{n_2}},$$  
$$R_4 : y \rightarrow \emptyset, \lambda_4 = y.$$

Based on these equations, we consider two different parameter sets,

Set 1 : $k_1 = 50, k_2 = 16, n_1 = 2.5, n_2 = 1,$

Set 2 : $k_1 = 60, k_2 = 30, n_1 = 3, n_2 = 2.$

Despite its small dimension, this model is already quite difficult to simulate to stationarity with SSA. In particular, the second parameter set was selected so that the system switches extremely rarely.

1. Steady-state simulation for set 1

This system fluctuates most of the time around two metastable regions of the $x$-$y$ space, which we shall call $B_1$ and $B_2$. In $B_1$, $x$ is much higher than $y$ (i.e., gene $X$ is on and $Y$ is off), while in $B_2$ the opposite holds. Random fluctuations can drive the system from one region to the other, but the switching is not frequent, as the simulated trajectory in Figure 1 suggests.

Starting from this generated trajectory, we can now locate the boundaries of the two regions. Our goal is to isolate the regions in which most of the simulation time is spent. Since the number of states accessed with non-negligible probability by this system is relatively small, we use the following box approximations for the metastable sets, determined with the
help of trajectory plots, such as Figure 1,

\[ B_1 = \{ x, y : 0 \leq x \leq 2 \text{ and } 6 \leq y \leq 30 \}, \]

\[ B_2 = \{ x, y : 4 \leq x \leq 70 \text{ and } 0 \leq y \leq 2 \}. \]

To assess the accuracy of our results, we also calculated a good approximation to the invariant distribution of the system using a large truncation of the state space (140 × 140 molecules). Since excursions of the system outside the truncation box are extremely rare, we can safely assume that the invariant distribution we calculated is, for all practical purposes, the true one. This distribution is displayed in Figure 2, and it will be the distribution with respect to which we shall compare our simulation approximations.

The toggle switch model with the first parameter set is still tractable by SSA. Figure 3 displays boxplots of the error in the approximation of \( \pi \) (the true invariant distribution) by \( \hat{\pi} \) (the simulation-based approximation) obtained by running 50 repetitions of SSA and our proposed algorithm for various trajectory time lengths \( T \). Note that in the case of SSA the full chain is simulated, while in our algorithm simulation is only performed for the censored chain (for the same length \( T \), and \( \hat{\pi} \) is computed via the coupling factors, as presented in Subsection III D.

Given the small sizes of the generator matrices involved in this example, all algebraic calculations are performed in less than a second for this system. Moreover, the check whether the simulated sample path lies inside \( B_1^{c}\cup B_2^{c} \) or \( B_1^{c}\cup B_2^{c} \) at every simulation step is very fast, thanks to the algebraic description of the metastable sets. Note that a direct comparison of simulation costs for the censored and full chain is not straightforward, as they evolve on different spaces. In the case of the first parameter set, for example, simulation of the full chain for \( T = 2 \times 10^4 \) time units (t.u.) requires between \( 8 \times 10^5 \) and \( 9 \times 10^5 \) steps, which on average take 14 s of CPU time. For the censored chain and the same simulation length, the number of steps is about a half, while the average CPU time is 11 s. Thus, in terms of step numbers, the runs of the full chain are about twice as fast compared to the censored chain, yet the two chains evolve on different spaces and this criterion is not as meaningful as the direct comparison of CPU times for a given simulation length \( T \).

Based on these observations, Figure 3 indicates that for the present example our method displays more than an order of magnitude faster convergence in terms of simulation length, which implies the same for computation time. This speedup is to be expected, as the estimated coupling factors are \( \xi = [0.6984 \ 0.2812 \ 0.0204] \) (corresponding to sets \( B_1, B_2, \) and \( B_3, \) with \( B_1 \) and \( B_2 \) as described above and \( B_3 = (B_1 \cup B_2)^c \)), which indicates that the full chain spends 1/50th of its time inside \( B_3 \) at stationarity.

2. Steady-state simulation for set 2

With the second parameter set, the system switches extremely rarely between the two metastable regions. This feature is also suggested by its (truncation-based) invariant distribution, shown on Figure 4, where it can be observed that
the metastable regions are connected by a very low-mass “corridor.”

Based on a few short simulation runs started from different points in the state space, we next locate the metastable regions of the chain and define their box approximations

\[ B_1 = \{ x, y : 0 \leq x \leq 2 \text{ and } 5 \leq y \leq 60 \}, \]

\[ B_2 = \{ x, y : 14 \leq x \leq 105 \text{ and } 0 \leq y \leq 2 \}. \]

The two largest eigenvalues of the constructed \( Q_{11} \) and \( Q_{22} \) matrices are shown on Table I. As mentioned in Sec. IV, the largest eigenvalue of each matrix determines the mean exit time from the corresponding region and, as can be seen from the table, these times are very long for the system considered. Consequently, brute-force stochastic simulation becomes extremely inefficient in this case.

Application of our method results in a dramatic reduction in simulation time, since relatively short trajectories of the censored chain yield a good approximation of the stationary distribution, as shown on Figure 5. As an example, trajectories of length \( T = 10^4 \) t.u. \( \approx 5.24 \times 10^4 \) t.u. yield an approximation error of about 1\%, while the full chain takes on average longer to just exist once from \( B_1 \).

Regarding run times of stochastic simulation, the same observations are true for parameter set 2 as well: for a trajectory of \( T = 2 \times 10^4 \) t.u., simulation of the full chain takes approximately 20 s of CPU time, while making \( 2.3 \times 10^6 \) steps; simulation of the censored chain for the same length \( T \) requires 7 s of CPU time on average, and \( 5 \times 10^5 \) steps.

**B. An exclusive genetic switch**

The second system we consider is a simplified model of the \( \lambda \)-phage genetic switch, which has been presented and analyzed in Refs. 31 and 33. The steady-state simulation problem of this system has been addressed in Refs. 12 and 34, while its switching properties have been studied in Ref. 35. The interested reader is referred to Ref. 31 for a discussion of the modeling assumptions and biological relevance of the system and to Ref. 36 for a very detailed, large-scale model of this system.

In its simplified form, the system consists of two genes, \( A \) and \( B \), located on opposite DNA strands, and whose regulatory domains overlap. Proteins \( A \) and \( B \) are both able to bind as homodimers to the regulatory domains of their promoter and mutually exclude each other when bound, since the operators of genes \( A \) and \( B \) partially overlap. The resulting deterministic system displays bistability for certain parameter values, and the stationary distribution of the corresponding stochastic system is bimodal. As analyzed in depth in Ref. 35, this exclusive switch is much more stable (i.e., the system spends much more time in each metastable region before switching out of it) compared to a general switch, in which both proteins can be found on DNA simultaneously.31, 33

The chemical reactions that describe this system are given in Table II. Stochastic simulation of this network is very time-consuming due to the increased number of reactions. Combined with the extreme stability that the system

---

**TABLE I.** The two largest eigenvalues of \( Q_{11} \) and \( Q_{22} \), along with the average time spent in each metastable region prior to the first exit.

<table>
<thead>
<tr>
<th>( \lambda_1 )</th>
<th>( \lambda_2 )</th>
<th>( \lambda_2/\lambda_1 )</th>
<th>( T_{\text{exit}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Q_{11} )</td>
<td>(-1.23 \times 10^{-3})</td>
<td>(-0.93)</td>
<td>(7.54 \times 10^4)</td>
</tr>
<tr>
<td></td>
<td>(8.12 \times 10^4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( Q_{22} )</td>
<td>(-2.37 \times 10^{-3})</td>
<td>(-0.91)</td>
<td>(3.81 \times 10^6)</td>
</tr>
<tr>
<td></td>
<td>(4.20 \times 10^6)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE II.** Chemical reactions for the toggle switch model. \( O \) denotes the common regulatory region of both genes, which can either be bound to \( A_2 \) (\( OA_2 \)) or \( B_2 \) (\( OB_2 \)). The reaction rates are taken from Ref. 12 and are equal for the corresponding reactions of the two proteins, leading to a symmetric bimodal stationary distribution.

<table>
<thead>
<tr>
<th>Reactions for ( A )</th>
<th>Reactions for ( B )</th>
<th>( k_{\text{forward}} )</th>
<th>( k_{\text{backward}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A + A \rightleftharpoons A_2 )</td>
<td>( B + B \rightleftharpoons B_2 )</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>( O + A_2 \rightleftharpoons OA_2 )</td>
<td>( O + B_2 \rightleftharpoons OB_2 )</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>( O \rightarrow O + A )</td>
<td>( O \rightarrow O + B )</td>
<td>1</td>
<td>N/A</td>
</tr>
<tr>
<td>( OA_2 \rightarrow OA_2 + A )</td>
<td>( OB_2 \rightarrow O + B_2 )</td>
<td>1</td>
<td>N/A</td>
</tr>
<tr>
<td>( A \rightarrow \emptyset )</td>
<td>( B \rightarrow \emptyset )</td>
<td>0.25</td>
<td>N/A</td>
</tr>
</tbody>
</table>
displays, this fact precludes the use of SSA for accurately approximating its stationary distribution within any reasonable amount of time, as will be quantified below. On the other hand, the size of a state-space truncation required to compute $\pi$ with good accuracy is too large. Thus, to judge the quality of the results obtained with our method, we exploit the symmetry of the chemical reactions and select reaction rates, so that the resulting bimodal stationary distribution is symmetric. Then, by defining $B_1$ and $B_2$ in a symmetric fashion, we can evaluate the quality of our approximation by calculating the deviation of $\hat{\pi}(B_1)$ and $\hat{\pi}(B_2)$ from equality. A similar approach, based on comparing the modes of the stationary distribution of $\Delta = N_A^{\text{tot}} - N_B^{\text{tot}}$ (where $N_A^{\text{tot}} = A + A_2 + O A_2$, $N_B^{\text{tot}} = B + B_2 + O B_2$) was followed in Ref. 12.

This system evolves in a five-dimensional state space, as it comprises the following species: the monomers $A$ and $B$, the dimers $A_2$ and $B_2$, and the operator $O$ (which can be found in three states: empty, bound to $A_2$ ($O A_2$) and bound to $B_2$ ($O B_2$)). This implies that $B_1$ and $B_2$ have to be carefully defined to permit the calculation of the necessary matrices for our method. Box over approximations are not efficient in this sense, as they include unnecessarily many states: the actual stationary distribution over each metastable region shows correlations between all states. We thus used the second approach outlined in Sec. IV (a few short simulations started from different initial conditions) to define $B_1$ and $B_2$. Because of the symmetry in the reaction rates, we expect the two metastable regions to also be symmetrically located. If we define the state vector of the system as $[A \ B \ A_2 \ B_2 \ O \ O A_2 \ O B_2]$ (where only one of the last three coordinates can be equal to one at any given time, and the rest are equal to zero), we observe that for each state $x = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7] \in B_1$ there should correspond a “symmetric” state $x' = [x_2 \ x_1 \ x_4 \ x_3 \ x_5 \ x_7 \ x_6] \in B_2$. Taking into account all states $x \in B_1$ and $x' \in B_2$ visited in the simulations, we finally ended up with two sets of 7635 states each, such that for every state $x$ contained in one of them, its corresponding symmetric state $x'$ is contained in the other. With these sets, the calculation of all the necessary matrices for our method took about 7 min.

Figure 6 shows how the relative difference between the estimated stationary probabilities of the metastable regions evolves as the simulation length $T$ increases, and Figure 7 displays the approximated marginal stationary distribution of $N_A^{\text{tot}}$ and $N_B^{\text{tot}}$, obtained from a run of length $T = 5 \times 10^4$ t.u. for the censored chain. The outer contours contain several orders of magnitude smaller mass than the inner ones, hence they are not as well approximated. Moreover, since the censored chain cannot visit states with probability less than about $5 \times 10^{-12}$ in the given simulation horizon, the upper right-hand side of the plot is blank, as it contains only such states.

Each step of the censored chain requires about three to four times more CPU time than a SSA step in this case. For example, full chain simulation for $T = 10^4$ t.u. takes about 30 s of CPU time, whereas for the same $T$, the censored chain requires 100 s on average. However, the method remains very efficient: simulation of 25 full chain trajectories of length $T = 1.5 \times 10^6$ t.u. each, requiring 75 min of CPU time per trajectory on average, can only achieve a median log-absolute difference of $-0.126$ with 25% and 75% quantiles equal to $-0.25$ and $-1.5 \times 10^{-4}$, respectively. Increasing $T$ to $3.5 \times 10^6$ t.u., requires 175 min of CPU time per trajectory, while achieving a median of $-0.46$, with 25% and 75% quantiles equal to $-0.79$ and $-0.25$, respectively. This distribution looks very similar to the one shown in the first boxplot of Figure 6, which corresponds to trajectories of length $T = 10^3$ t.u. for the censored chain, requiring just 10 s of CPU time per trajectory. Therefore, the efficiency of our method remains very high. This is confirmed by the fact that the estimated coupling factor $\xi$, corresponding to the fraction of time spent in by the full chain in $B_1$ at stationarity, was $3.3 \times 10^{-4}$ with a standard deviation of $2.3 \times 10^{-6}$ (based on 50 simulation runs of length $T = 2 \times 10^5$ t.u.).

FIG. 6. Deviation from equality for the stationary probabilities of the two metastable regions, plotted against the simulation length $T$ for the censored chain. The results are based on 50 simulation runs. Empty circles and horizontal bands inside boxes denote means and medians, respectively, upper and lower ends of boxes correspond to 75th and 25th percentiles, and box whiskers extend to approximately 95% percentiles. Outliers are denoted by dots.

FIG. 7. Logarithm of the steady-state marginal probability distribution of the total numbers of $A$ and $B$ molecules in the system, approximated from a simulation run of $5 \times 10^4$ t.u. for the censored chain. To produce the simulated trajectory that yielded this approximation, a total computation time of about 15 min (including matrix calculations) was needed.
VI. COMPARISON WITH NON-EQUILIBRIUM UMBRELLA SAMPLING

NEUS is a powerful steady-state simulation algorithm introduced by Warmflash et al.\textsuperscript{12} The main method was further developed in order to handle systems with many order parameters\textsuperscript{37} and calculate transition rates.\textsuperscript{38} The basic idea of the original algorithm is to consider a partition of the space of order parameters (loosely speaking, a set of coordinates that characterize system behavior) into many regions and enforcing equal sampling in all regions. This is achieved by running restricted stochastic simulation of the system inside each region and accumulating boundary crossing flux statistics over time, which are in turn used to properly initialize the trajectory segments in each region, as well as weight the regions correctly against each other. This scheme is iterated until the accumulated process statistics inside each region converge to their stationary values.

Compared to other advanced steady-state simulation techniques, our method shares the greatest similarities with NEUS. Indeed, the state space partitions presented in the examples above can be considered as a special case of a NEUS partition. While formally similar, however, we believe the two approaches have some crucial differences: in order to enforce event sampling in a high-dimensional order-parameter space, an exponentially increasing number of regions has to be considered in NEUS,\textsuperscript{12} while the string-based version of the algorithm\textsuperscript{37} that overcomes this problem requires a much more elaborate region-handling scheme that is applicable only in the presence of two metastable regions. Moreover, the choice of order parameters in the original NEUS scheme is crucial for the convergence of the algorithm.\textsuperscript{12,14}

More generally, convergence of NEUS is determined by an interplay of several factors, such as the selected order parameters, the tiling of the state space, and the update schemes for the boundary fluxes and region weights. Proper tuning of the method is thus a highly nontrivial task, and a formal analysis of the approximation errors is very difficult. Still, it should be acknowledged that NEUS has a much greater range of applicability, as it can be used with any type of stochastic dynamics and not only Markov chains. The string version of the method is also applicable to very high-dimensional systems, where other simulation methods typically fail.

Stochastic complements on the other hand, are well-defined objects only for systems described by Markov chain dynamics, as they are intimately related to the properties of the process generator. For this class of systems (and a not too high number of dimensions), our method is able to exploit the analytical properties of the Markov chain to address some of the problems associated with NEUS. State space partitioning can be done more intuitively, based on the consideration of system dynamics and with the clear objective of removing high-probability regions from the process state space. Piecewise trajectory simulation and re-initialization in the transition region is not needed, since the stochastic complements are able to capture accurately the behavior of the process on the region boundaries. Finally, convergence is ensured by the ergodic properties of the simulated Markov chains, while steady-state simulation theory guarantees that approximation errors decay predictably with the simulation length (more details can be found in Appendix B).

NEUS has been shown to be very efficient compared to SSA in the case of the genetic toggle switch model.\textsuperscript{12} Our simulation results demonstrate that complement-based simulation also converges much faster than SSA. However, a fair direct comparison of the performance of the two methods is not straightforward, as the computer implementation of NEUS has to be carefully optimized to achieve fast convergence. To avoid comparing our method to a suboptimal implementation of NEUS, we elected to do this indirectly, using the already published data on the application of NEUS to the exclusive switch model and its convergence properties. To this end, we consider Figure 3(b) of the original NEUS paper,\textsuperscript{12} which displays the average estimation error of the method versus SSA as a function of simulation steps. In that work, estimation error is quantified by $e_1 = |P(\Delta = 40)/P(\Delta = -40) - 1|$, where $\Delta = N^B_A - N^A_B$ and 40 is a value close to the mode of the stationary distribution of $\Delta$, denoted by $P(\Delta)$. In our case, we have chosen to calculate $e_2 = (|\pi(B_1)/\pi(B_2)| - 1)$. The two error measures do not differ much conceptually, and our calculations show that in practice $e_1$ and $e_2$ agree with very good accuracy and can be used interchangeably.

In the NEUS paper, the comparison with SSA is carried out in terms of simulation steps, under the estimation of a 10% overhead per step for NEUS. Since the average number of steps grows linearly with the simulated trajectory length (in time units), a conversion between the two is very simple. As stated above, a step for our reduced chain takes about 3.5 times longer to simulate than an SSA step. Our simulation results show that the censored chain requires $1.5 \times 10^6$ steps on average for a trajectory of length $T = 10^4$ t.u. Thus, in relation to Figure 3(b) of Warmflash et al.,\textsuperscript{12} the rightmost box in our Figure 6 ($T = 10^5$ t.u.) corresponds to about $75 \times 10^6$ steps for the reduced chain, or $(3.5 \times 75) \times 10^6 = 262.5 \times 10^6$ “SSA-equivalent” steps, while the leftmost box ($T = 10^3$ t.u.) corresponds to $1.5 \times 10^5$ steps on average. Comparing the results of the two plots under the correct scaling of the horizontal axes, we can approximately infer that complement-based simulation provides the same accuracy as NEUS for a given number of steps. We have also verified the SSA error plot on Figure 3(b) of Warmflash et al.\textsuperscript{12} by running SSA trajectories up to $1.7 \times 10^{10}$ steps long, which provides an independent consistency check for our calculations.

In short, complement-based steady-state simulation performs as well as NEUS on a benchmark system that has been considered in several rare-event simulation works,\textsuperscript{12,13,34,35} while offering an intuitively simple and flexible implementation. On the other hand, we should also remark that NEUS, FFS, as well as other rare-event simulation techniques, such as the Barrier Method,\textsuperscript{39} can be used to calculate the transition rates between metastable regions. This is not possible with our method in its present form.

VII. CONCLUDING REMARKS

We have presented an algorithm for estimating the stationary distribution of a metastable Markov chain, which is centered around the theory of stochastic complements. Its
main characteristic is the use of a reduced Markov chain on the set of transition states and the separate treatment of the metastable regions. With this divide-and-conquer approach, a hard simulation task is broken up into a few easier steps, which take much less time to complete. The maximum number of species to which our methodology is applicable cannot be determined a priori, since every particular metastable system has its own characteristics. However, it is certain that the speedup achieved in comparison to SSA can increase the range of tractable simulation problems considerably.

In conclusion, we believe that the introduction of stochastic complementation in the field of rare-event simulation opens many interesting possibilities. The method proposed in this paper is a simple combination of stochastic simulation with stochastic complements, however more powerful methods can be constructed using elements from existing powerful rare-event simulation techniques.

ACKNOWLEDGMENTS

This work was supported by HYCON2 Network of Excellence, FP7-ICT- 257462, and the YeastX project of SystemsX.ch.

APPENDIX A: ITERATIVE STOCHASTIC COMPLEMENTATION

Consider two finite disjoint sets \( A_1, A_2 \subset S \). Contrary to the treatment of metastable sets above, we allow transitions between \( A_1 \) and \( A_2 \). By properly enumerating the states of \( A_1 \) and \( A_2 \), we write the generator matrix of the full chain as

\[
Q = \begin{bmatrix}
Q_{11} & Q_{12} & Q_{13} \\
Q_{21} & Q_{22} & Q_{23} \\
Q_{31} & Q_{32} & Q_{33}
\end{bmatrix}.
\]

We first set

\[
\tilde{Q} = \begin{bmatrix}
Q_{22} & Q_{23} \\
Q_{32} & Q_{33}
\end{bmatrix}
\]

and consider the stochastic complement of \( \tilde{Q} \) in \( Q \).

\[
\tilde{R} = \begin{bmatrix}
\tilde{R}_{11} & \tilde{R}_{12} \\
\tilde{R}_{21} & \tilde{R}_{22}
\end{bmatrix} = \tilde{Q} - \begin{bmatrix}
Q_{21} & Q_{31} \\
Q_{23} & Q_{33}
\end{bmatrix} \begin{bmatrix}
Q_{11}^{-1} & Q_{12} \\
Q_{13} & Q_{13}
\end{bmatrix}
\]

\[
= \begin{bmatrix}
Q_{22} - Q_{21}Q_{11}^{-1}Q_{12} & Q_{23} - Q_{21}Q_{11}^{-1}Q_{13} \\
Q_{32} - Q_{31}Q_{11}^{-1}Q_{12} & Q_{33} - Q_{31}Q_{11}^{-1}Q_{13}
\end{bmatrix}.
\]

This generator corresponds to a reduced Markov chain on \( S \backslash A_1 \). We next consider a further reduction of this chain by taking the stochastic complement of \( \tilde{R}_{22} \) in \( \tilde{R} \). To simplify the notation, we set \( \tilde{Q} = Q_{22} - Q_{21}Q_{11}^{-1}Q_{12} \). \( \tilde{R} \) is then given by

\[
\tilde{R} = Q_{33} - Q_{31}Q_{11}^{-1}Q_{13} - (Q_{32} - Q_{31}Q_{11}^{-1}Q_{12})
\]

\[
\times (\tilde{Q})^{-1}(Q_{23} - Q_{21}Q_{11}^{-1}Q_{13})
\]

and corresponds to a reduced Markov chain on \( S \backslash (A_1 \cup A_2) \).

The same result can be obtained by removing \( A_1 \cup A_2 \) from \( S \) in one step, thanks to the intrinsic consistency of the stochastic complement definition. To verify this, consider \( \tilde{R}' \) given by

\[
\tilde{R}' = Q_{33} - [Q_{31} Q_{32}] \begin{bmatrix}
Q_{11} & Q_{12} \\
Q_{21} & Q_{22}
\end{bmatrix}^{-1} \begin{bmatrix}
Q_{13} \\
Q_{23}
\end{bmatrix}.
\]

The block inverse matrix in this formula can be rewritten using the matrix inversion lemma and the definition of \( \tilde{Q} \) as follows:

\[
\begin{bmatrix}
Q_{11} & Q_{12} \\
Q_{21} & Q_{22}
\end{bmatrix}^{-1} = \begin{bmatrix}
Q_{11}^{-1} + Q_{11}^{-1}Q_{12}\tilde{Q}_{21}Q_{11}^{-1} - Q_{11}^{-1}Q_{12}\tilde{Q} \\
-\tilde{Q}_{21}Q_{11}^{-1} & \tilde{Q}
\end{bmatrix}.
\]

By expanding and factorizing the matrix products in (A2), one can verify that \( \tilde{R}' = \tilde{R} \).

APPENDIX B: ESTIMATOR EFFICIENCY FOR THE REDUCED CHAIN

Consider a state \( x \in B_3 \) and assume that the stationary probability of this state is \( \pi(x) = p_x \), where \( \pi \) is the stationary distribution of the full chain. Running the SSA on the full chain to approximate \( p_x \) amounts to constructing an estimator

\[
\hat{p}_x(t) = \frac{1}{t} \int_0^t I_x(x(s))ds,
\]

where \( I_x \) is the indicator function of \( x \). This estimator will converge to \( p_x \) almost surely \( t \to \infty \), thanks to the ergodic theorem for Markov chains,\(^\text{17}\) and \( \sqrt{t}(\hat{p}_x(t) - p_x) \Rightarrow \mathcal{N}(0, \sigma_x^2) \)

(\( \Rightarrow \) denotes weak convergence and \( \mathcal{N}(0, 1) \) is the standard normal distribution), thanks to the functional central limit Theorem for Markov chains.\(^\text{40,41} \) We thus see that the variance of \( \hat{p}_x \) goes down with \( t^{-1} \), provided we ignore the estimator bias due to the initial condition, which anyway decreases an order of magnitude faster than the variance.\(^\text{40} \)

The constant \( \sigma_x^2 \) is called the time-average variance constant (TAVC) of the estimator.\(^\text{40} \) Notice that the estimators for different states \( x \) have different TAVCs.

The stationary probability of \( x \) will grow by a factor \( 1/\pi(B_3) \) in the reduced chain on \( B_3 \). In other words, if \( Tp_x \) is the average amount of time spent by the full chain at \( x \) over the course of a long simulation of length \( T \), the amount of time spent at the same point by the reduced chain over the same simulation length \( T \) is \( Tp_x/\pi(B_3) \). Equivalently, one can say that the TAVC of \( \hat{p}_x(t) \) for the reduced chain will be reduced by a factor \( 1/\pi(B_3) \), a conclusion that holds for all \( x \in B_3 \).

We thus see that the reduction of the Markov chain state space through stochastic complementation implies an increase in the efficiency of our steady-state estimators, whose magnitude depends on the amount of stationary mass contained in the removed part of the state space.


29 E. van Doorn and P. Pollett, “Quasi-stationary distributions,” Memorandum No. 1945, Department of Applied Mathematics, University of Twente, 2011, see http://eprints.eemcs.utwente.nl/20245/.
32 More analytically, we removed all transitions leading out of the set [0, 140] × [0, 140] in the original chain, and formed the generator matrix of a Markov chain evolving on this finite set, with the remaining transition rates equal to those of the original chain. We then computed and verified the stationary distribution of this Markov chain using standard linear algebra tools in MATLAB.