

RESEARCH ARTICLE

A Hybrid of the Chemical Master Equation and the Gillespie Algorithm for Efficient Stochastic Simulations of Sub-Networks

Jaroslav Albert*

Applied Physics Research Group, Vrije Universiteit Brussel, Brussels, Belgium

* jalbert@ulb.ac.be



OPEN ACCESS

Citation: Albert J (2016) A Hybrid of the Chemical Master Equation and the Gillespie Algorithm for Efficient Stochastic Simulations of Sub-Networks. PLoS ONE 11(3): e0149909. doi:10.1371/journal.pone.0149909

Editor: Ramon Grima, University of Edinburgh, UNITED KINGDOM

Received: November 16, 2015

Accepted: February 5, 2016

Published: March 1, 2016

Copyright: © 2016 Jaroslav Albert. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All data are available in the paper.

Funding: This work was supported by The Interuniversity Attraction Poles program of the Belgian Science Policy Office, under grant IAP P7-35 (www.belspo.be) and The Onderzoeksraad of the Vrije Universiteit Brussel (www.vub.ac.be) through the strategic research program "Applied Physics and Systems Biology".

Competing Interests: The authors have declared that no competing interests exist.

Abstract

Modeling stochastic behavior of chemical reaction networks is an important endeavor in many aspects of chemistry and systems biology. The chemical master equation (CME) and the Gillespie algorithm (GA) are the two most fundamental approaches to such modeling; however, each of them has its own limitations: the GA may require long computing times, while the CME may demand unrealistic memory storage capacity. We propose a method that combines the CME and the GA that allows one to simulate stochastically a part of a reaction network. First, a reaction network is divided into two parts. The first part is simulated via the GA, while the solution of the CME for the second part is fed into the GA in order to update its propensities. The advantage of this method is that it avoids the need to solve the CME or stochastically simulate the entire network, which makes it highly efficient. One of its drawbacks, however, is that most of the information about the second part of the network is lost in the process. Therefore, this method is most useful when only partial information about a reaction network is needed. We tested this method against the GA on two systems of interest in biology - the gene switch and the Griffith model of a genetic oscillator—and have shown it to be highly accurate. Comparing this method to four different stochastic algorithms revealed it to be at least an order of magnitude faster than the fastest among them.

Introduction

In a network of chemical reactions, the molecular concentrations at any given time cannot be predicted with a certainty; they can only be anticipated with a certain probability. This probability can in principle be determined by solving (analytically or numerically) the chemical master equation (CME). Attempting to do so, however, can more often than not be a frustrating exercise: except for a handful of simple cases, the CME cannot be solved analytically, and for a lot of interesting cases even a numerical solution can be near impossible to attain. One way around this obstacle was an algorithm proposed by Doob [1] and later presented and popularized by Gillespie [2]. The authors showed that the information stored in the CME can be

extracted through a series of relatively simple steps coupled with the help of a pseudo-random number generator. Known today by its popular name as the Gillespie algorithm (GA) (also known as kinetic Monte Carlo or stochastic simulation algorithm (SSA)), this procedure guarantees an exact solution to the CME, provided these steps are repeated sufficiently many times so as to build a statistically significant ensemble of data points. The solution to the CME can thus be reconstructed step by step without the need for enormous memory storage capacity that is usually required to solve the CME directly. The one drawback of the GA is that the number of steps required scales with the number of reactions and the magnitude of their rates. Consequently, for large reaction networks the running time may become impractical.

Since it first appeared, researchers have devised faster versions of the GA, some of which are exact [3, 4], in the sense that they give statistically identical answers as the CME, while others rely on approximations [5–8]. In conjunction with the Langevin approximation [9], these algorithms comprise a library of methods to choose from when simulating reaction networks. The Dizzy package [10], for instance, is one such library containing four stochastic simulators of various speed and accuracy.

With the advent of stochastic algorithms, the appeal for solving the CME directly has not diminished however. Having an analytical solution to the CME, even an approximate one, is extremely useful and can provide insight into the stochastic properties of chemically interacting networks. It is not surprising then that many methods for finding approximate solutions to the CME exist and continue to appear [11–16].

There exists a class of methods which involve partitioning a reaction network in a way that facilitates either a solution to the CME [17, 18] or a faster stochastic simulation [19–21] of one part of the reaction network, while yielding only a partial information about the rest of the network. Other partitioning methods rely on large differences among the values of the reaction rates [22, 23]. Such methods take advantage of the fast rates by considering the chemical species affected by these rates to be in a quasi-steady state.

In this paper, we present another method where synergy between the CME and the GA is exploited for the purpose of simulating one part of a reaction network. While previous methods of this kind rely on *a priori* approximations of either the CME or the GA, we begin by deriving the exact equations from which the next reaction time and the next reaction probability can be computed, as well as the exact CME for the non-simulated part of the network. Once these equations take form, they may be solved by virtue of approximations, either as a matter of necessity or in order to speed up the simulation process. We show on two biologically relevant examples how this method can be applied and discuss what its limitations are.

Materials and Methods

Chemical Master Equation

The time evolution of the joint probability distribution $P(\mathbf{n}, t)$ for a chemically reacting system comprising N molecular species and J reactions is governed by the chemical master equation (CME):

$$\dot{P}(\mathbf{n}, t) = \sum_{\mu=1}^J a_{\mu}(\mathbf{n} - \mathbf{v}_{\mu})P(\mathbf{n} - \mathbf{v}_{\mu}, t) - P(\mathbf{n}, t) \sum_{\mu=1}^J a_{\mu}(\mathbf{n}), \quad (1)$$

where \mathbf{n} is short for the set $\{n_1, n_2, \dots, n_N\}$, and $a_1(\mathbf{n}) \dots a_J(\mathbf{n})$, are the reaction propensities, which, for our purposes here, will depend on time only explicitly, i. e. through the variables \mathbf{n} . The matrix elements $v_{i\mu}$ specify the change in n_i due to the μ th reaction. It will be useful later

on to represent Eq (1) in another way [9, 24–26]. Let us define a vector state

$$|\psi(t)\rangle = \sum_{n_1, n_2, \dots, n_N} P(\mathbf{n}, t) \prod_{i=1}^N |n_i\rangle, \tag{2}$$

and operators $\hat{A}_i, \hat{A}_i^\dagger$ and \hat{n}_i the action of which on the state $|n_i\rangle$ is

$$\hat{A}_i |n_i\rangle = n_i |n_i - 1\rangle, \quad \hat{A}_i^\dagger |n_i\rangle = |n_i + 1\rangle, \quad \hat{n}_i |n_i\rangle = n_i |n_i\rangle, \tag{3}$$

where, by definition, $\hat{n}_i = \hat{A}_i^\dagger \hat{A}_i$. The index i means that the operators act only on the i th vector state, leaving the rest of them untouched. The vector state $|n_i\rangle$, and its transpose $\langle n_i|$ are simply the orthogonal unit column and row vectors respectively:

$$|n_i\rangle = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ \vdots \\ \vdots \end{pmatrix}_i \quad \langle n_i| = (0, \dots, 1, 0, \dots, \dots)_i \tag{4}$$

so that $\langle 0| = (1, 0, 0, \dots)$, $\langle 1| = (0, 1, 0, \dots)$, etc. The operators \hat{A}_i and \hat{A}_i^\dagger have the form

$$\hat{A}_i = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & \dots \\ 0 & 0 & 2 & 0 & & \\ 0 & 0 & 0 & 3 & & \\ \vdots & & & & \ddots & \\ \vdots & & & & & \ddots \\ \vdots & & & & & & \ddots \end{bmatrix}_i \quad \hat{A}_i^\dagger = \begin{bmatrix} 0 & 0 & 0 & \dots & \dots & \dots \\ 1 & 0 & 0 & & & \\ 0 & 1 & 0 & & & \\ 0 & 0 & 1 & \dots & & \\ \vdots & & & \ddots & & \\ \vdots & & & & \ddots & \\ \vdots & & & & & \ddots \end{bmatrix}_i. \tag{5}$$

The vector state product in Eq (2) can be thought of as a vector whose elements are the individual vector states, $|n_i\rangle$:

$$\prod_{i=1}^N |n_i\rangle = \begin{pmatrix} |n_1\rangle \\ |n_2\rangle \\ \cdot \\ \cdot \\ |n_N\rangle \end{pmatrix}. \tag{6}$$

A product of any operators, e. g. $\hat{A}_1\hat{A}_2$, could then be represented by an $N \times N$ matrix:

$$\hat{A}_1\hat{A}_2 = \hat{A}_1\hat{A}_2\mathbb{1}_3\dots\mathbb{1}_N = \begin{bmatrix} \hat{A}_1 & 0 & 0 & \cdot & \cdot & \cdot \\ 0 & \hat{A}_2 & 0 & 0 & & \\ 0 & 0 & \mathbb{1}_3 & 0 & & \\ \cdot & & & \cdot & & \\ \cdot & & & & \cdot & \\ \cdot & & & & & \mathbb{1}_N \end{bmatrix}, \tag{7}$$

where $\mathbb{1}_i$ is the identity operator: $\mathbb{1}_i|n_i\rangle = |n_i\rangle$.

With this notation, the master equation can be written in the form

$$\frac{d}{dt} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle, \tag{8}$$

where

$$\hat{H} = \sum_{\mu=1}^J \left[\prod_{j=1}^N ((\hat{n}_j + 1)^{-1} \hat{A}_j)^{\theta(-v_{j\mu})|v_{j\mu}|} (\hat{A}_j^\dagger)^{\theta(v_{j\mu})|v_{j\mu}|} - \prod_{j=1}^N \mathbb{1}_j \right] a_\mu(\hat{\mathbf{n}}) \tag{9}$$

is an operator acting on the vector state $\prod_{i=1}^N |n_i\rangle$ and $\theta(\cdot)$ is a step function centered around zero.

Let us check that Eqs (8) and (1) are indeed identical. To see how the operator \hat{H} in Eq (9) acts on the state $|\psi\rangle$, let us first look at how the operators within it act on the individual vector

states $|n_j\rangle$. We have

$$\begin{aligned}
 ((\hat{n}_j + 1)^{-1} \hat{A}_j)^a (\hat{A}_j^\dagger)^b |n_j\rangle &= ((\hat{n}_j + 1)^{-1} \hat{A}_j)^{a-1} [((\hat{n}_j + 1)^{-1} \hat{A}_j)] |n_j + b\rangle \\
 &= ((\hat{n}_j + 1)^{-1} \hat{A}_j)^{a-1} [(n_j + b)(\hat{n}_j + 1)^{-1}] |n_j + b - 1\rangle \\
 &= ((\hat{n}_j + 1)^{-1} \hat{A}_j)^{a-1} |n_j + b - 1\rangle \\
 &= ((\hat{n}_j + 1)^{-1} \hat{A}_j)^{a-2} |n_j + b - 2\rangle \\
 &= ((\hat{n}_j + 1)^{-1} \hat{A}_j)^{a-3} |n_j + b - 3\rangle \\
 &\vdots \\
 &\vdots \\
 &= |n_j + b - a\rangle
 \end{aligned} \tag{10}$$

and hence,

$$((\hat{n}_j + 1)^{-1} \hat{A}_j)^{\theta(-v_{j\mu})|v_{j\mu}|} (\hat{A}_j^\dagger)^{\theta(v_{j\mu})|v_{j\mu}|} |n_j\rangle = |n_j + \theta(v_{j\mu})|v_{j\mu}| - \theta(-v_{j\mu})|v_{j\mu}|\rangle. \tag{11}$$

Since the second term is merely the identity operator, it leaves the state unchanged. The operator $a_\mu(\hat{n})$ acting on the product state $\prod_j |n_j\rangle$ also leaves it unchanged and itself becomes a number, $a_\mu(\mathbf{n})$. Putting the above relations together, we can write

$$\begin{aligned}
 \hat{H}|\psi\rangle &= \sum_{n_1, n_2, \dots, n_N} P(\mathbf{n}, t) \hat{H} \left[\prod_{j=1}^N |n_j\rangle \right] \\
 &= \sum_{n_1, n_2, \dots, n_N} \sum_{\mu=1}^J P(\mathbf{n}, t) a_\mu(\mathbf{n}) \left[\prod_{j=1}^N |n_j + \theta(v_{j\mu})|v_{j\mu}| - \theta(-v_{j\mu})|v_{j\mu}|\rangle - \prod_{j=1}^N |n_j\rangle \right] \\
 &= \sum_{n_1, n_2, \dots, n_N} \sum_{\mu=1}^J P(\mathbf{n} - \mathbf{v}_\mu, t) a_\mu(\mathbf{n} - \mathbf{v}_\mu) \left[\prod_{j=1}^N |n_j\rangle \right],
 \end{aligned} \tag{12}$$

where in the last line we used the fact that $v_{j\mu} = \theta(v_{j\mu})|v_{j\mu}| - \theta(-v_{j\mu})|v_{j\mu}|$. The left hand side of Eq (8) states that

$$\frac{d}{dt} |\psi(t)\rangle = \sum_{n_1, n_2, \dots, n_N} \frac{d}{dt} P(\mathbf{n}, t) \left[\prod_{j=1}^N |n_j\rangle \right]. \tag{13}$$

The only way expressions Eqs (12) and (13) can be equal is if the coefficients of the product vector state $\prod_j |n_j\rangle$ on both sides are equal. This leads to Eq (1).

The formal solution to Eq (8) is

$$|\psi(t)\rangle = e^{\hat{H}t} |\psi(0)\rangle, \tag{14}$$

where $|\psi(0)\rangle$ is the initial vector state specified entirely by $P(\mathbf{n}, 0)$. The operator $e^{\hat{H}t}$ is called the evolution operator. Multiplying both sides of Eq (14) by $\langle \mathbf{n} |$ and invoking the orthogonality relation $\langle \mathbf{n}' | \mathbf{n} \rangle = \delta_{n'_1 n_1} \delta_{n'_2 n_2} \dots \delta_{n'_N n_N}$ we obtain the probability distribution

$$\langle \mathbf{n} | \psi(t) \rangle = \langle \mathbf{n} | e^{\hat{H}t} |\psi(0)\rangle = P(\mathbf{n}, t), \tag{15}$$

where for brevity $|\mathbf{n}\rangle = \prod_{i=1}^N |n_i\rangle$. With this formalism it is easy to write down quantities such

as the transition probabilities. For example,

$$W(\mathbf{n}', t' | \mathbf{n}, t) = \langle \mathbf{n}' | e^{\hat{H}(t'-t)} | \mathbf{n} \rangle, \tag{16}$$

means the probability to find the system in the state \mathbf{n}' at time t' if at time t it was in the state \mathbf{n} .

Lastly, let us also write down the identity operator in the form

$$\mathbb{1} = \sum_n |n\rangle \langle n| \tag{17}$$

as it will be useful in later sections.

Gillespie Algorithm

The idea behind the GA is to simulate a chain of Markov processes by sampling the probability distribution of the time elapsed since the last reaction, τ , and the probability that a specific reaction, μ , will occur at τ , such that any reaction occurring at τ has probability 1. The steps are as follows:

1. At some initial time t (e. g. $t = 0$) select your initial state \mathbf{n} and compute the propensities $a_\mu(\mathbf{n})$.
2. Select two random numbers r_1 and r_2 .
3. Compute τ using the formula

$$\tau = (1/a_0) \ln(1/r_1), \tag{18}$$

where $a_0 = \sum_\mu a_\mu(\mathbf{n})$.

4. Find the smallest integer j that satisfies

$$\sum_{j'=1}^j a_{j'}(\mathbf{n}) > r_2 a_0(\mathbf{n}), \tag{19}$$

and set $j = \mu$.

5. Update the system according to $\mathbf{n}(t + \tau) = \mathbf{n}(t) + \nu_\mu$ and set $t = t + \tau$.
6. Return to step 1.

Repeating these steps until t reaches some final time leads to a particular path, or realization, for \mathbf{n} . In order to obtain the same information within this time interval as is contained in the CME, one must compute this realization infinitely many times. It is in this sense that the GA and the CME are exactly equivalent. Of course, in practice one only needs to compute a finite number of realizations to extract meaningful information about the system. Depending on how many realizations are considered “sufficient” and how long it takes to compute each realization, the GA may be a fast route to solving the CME, or it may be a very slow one. In the next section we will show how one can combine the CME and the GA in order to simulate stochastically a part of a system.

CME-GA hybrid

Consider a reaction network of J reactions with propensities $\{a_1, \dots, a_J\}$, comprising two sets of molecular species: $\mathbf{m} = \{m_1, \dots, m_{N_1}\}$ and $\mathbf{n} = \{n_1, \dots, n_{N_2}\}$ ($N_1 + N_2 = N$). The propensities are some functions of \mathbf{m} and \mathbf{n} , but not time. Let us arrange the reactions into two groups, $G_1 = \{a_1, \dots, a_{K-1}\}$ and $G_2 = \{a_K, \dots, a_J\}$, such that the reactions in group G_1 can affect both

\mathbf{m} and \mathbf{n} , while the reactions in group G_2 can only affect \mathbf{n} . During a time interval, τ , in which no reaction occurs in G_1 , the CME for G_2 can be written as follows:

$$\begin{aligned} \dot{P}_{\mathbf{m}}(\mathbf{n}, t) = & \sum_{\mu=K}^J a_{\mu}(\mathbf{m}, \mathbf{n} - v_{\mu}) P_{\mathbf{m}}(\mathbf{n} - v_{\mu}, t) \\ & - P_{\mathbf{m}}(\mathbf{n}, t) \sum_{\mu=K}^J a_{\mu}(\mathbf{m}, \mathbf{n}). \end{aligned} \tag{20}$$

The subscripts \mathbf{m} in $P_{\mathbf{m}}(\mathbf{n}, t)$ serve as a reminder that the solution of $P_{\mathbf{m}}(\mathbf{n}, t)$ will depend on their value. Note that by definition \mathbf{n} may change during τ only via the reaction channels in G_2 , but not in G_1 . Let us now see how one can sample τ and the next reaction in G_1 from their respective probability distributions in a manner similar to the one described in the previous section. We begin by deriving the probability distribution for τ .

Let us divide time into L discrete infinitesimal intervals, Δt , such that $\Delta t L = \tau$. Using the notation introduced earlier, the probability that no reaction in G_1 occurs in the time τ can be expressed as

$$\begin{aligned} Q(\tau) = & \sum_{\mathbf{n}_0, \mathbf{n}_1, \dots} P_{\mathbf{m}}(\mathbf{n}_0, 0) \prod_{k=0}^{L-1} e^{-\Sigma_k \Delta t} \times \\ & \langle \mathbf{n}_L | e^{\hat{H} \Delta t} | \mathbf{n}_{L-1} \rangle \langle \mathbf{n}_{L-1} | e^{\hat{H} \Delta t} | \mathbf{n}_{L-2} \rangle \dots \langle \mathbf{n}_2 | e^{\hat{H} \Delta t} | \mathbf{n}_1 \rangle \langle \mathbf{n}_1 | e^{\hat{H} \Delta t} | \mathbf{n}_0 \rangle, \end{aligned} \tag{21}$$

where $\Sigma_k = \sum_{i=K}^J a_i(\mathbf{m}, \mathbf{n}_k)$ and k refers to the k th time interval. The exponential terms, $\exp[-\Sigma_k \Delta t]$, are the probabilities that, starting at time $t_k = k \Delta t$, no reaction occurs in the time Δt . Since the Σ s are numbers, not operators, we can move them wherever we want within the total product. Especially useful is to move each $\exp[-\Sigma_k \Delta t]$ just left of the state $|\mathbf{n}\rangle$ for each k :

$$\begin{aligned} Q(\tau) = & \sum_{\mathbf{n}_1, \mathbf{n}_2, \dots} \langle \mathbf{n}_L | e^{\hat{H} \Delta t} e^{-\Sigma_{L-1} \Delta t} | \mathbf{n}_{L-1} \rangle \langle \mathbf{n}_{L-1} | e^{\hat{H} \Delta t} e^{-\Sigma_{L-2} \Delta t} | \mathbf{n}_{L-2} \rangle \dots \\ & \dots \langle \mathbf{n}_2 | e^{\hat{H} \Delta t} e^{-\Sigma_1 \Delta t} | \mathbf{n}_1 \rangle \langle \mathbf{n}_1 | e^{\hat{H} \Delta t} e^{-\Sigma_0 \Delta t} | \phi \rangle, \end{aligned} \tag{22}$$

where $|\phi\rangle = \sum_{\mathbf{n}_0} P_{\mathbf{m}}(\mathbf{n}_0, 0) |\mathbf{n}_0\rangle$ is the initial state. Now, thanks to the fact that

$$\Sigma_k |\mathbf{n}_k\rangle = \sum_{i=1}^K a_i(\mathbf{m}, \mathbf{n}_k) |\mathbf{n}_k\rangle = \hat{\Sigma}_k |\mathbf{n}_k\rangle = \sum_{i=1}^K a_i(\mathbf{m}, \hat{\mathbf{n}}_k) |\mathbf{n}_k\rangle, \tag{23}$$

and the relation $\lim_{\epsilon \rightarrow 0} \exp[\hat{B} \epsilon] \exp[\hat{C} \epsilon] = \exp[(\hat{B} + \hat{C}) \epsilon]$ for arbitrary operators \hat{B} and \hat{C} , we may write

$$\begin{aligned} Q(\tau) = & \sum_{\mathbf{n}_L} \langle \mathbf{n}_L | e^{\hat{H}' \Delta t} \left(\sum_{\mathbf{n}_{L-1}} |\mathbf{n}_{L-1}\rangle \langle \mathbf{n}_{L-1}| \right) e^{\hat{H}' \Delta t} \left(\sum_{\mathbf{n}_{L-2}} |\mathbf{n}_{L-2}\rangle \langle \mathbf{n}_{L-2}| \right) \dots \\ & \dots e^{\hat{H}' \Delta t} \left(\sum_{\mathbf{n}_1} |\mathbf{n}_1\rangle \langle \mathbf{n}_1| \right) e^{\hat{H}' \Delta t} | \phi \rangle = \sum_{\mathbf{n}_L} \langle \mathbf{n}_L | e^{\hat{H}' \tau} | \phi \rangle, \end{aligned} \tag{24}$$

where $\hat{H}' = \hat{H} + \hat{\Sigma}$. Recalling [Eq \(17\)](#), we can set all the terms in the parentheses to unity. The expression for $Q(\tau)$ can now be written as

$$Q(\tau) = \sum_{\mathbf{n}_L} Q_{\mathbf{m}}(\mathbf{n}_L, t_L), \tag{25}$$

where, setting t_L to τ ,

$$Q_m(\mathbf{n}, \tau) = \langle \mathbf{n} | e^{\hat{H}'\tau} | \phi \rangle. \tag{26}$$

This expression is identical in structure to Eq (15) and as such can be expressed in a differential form:

$$\begin{aligned} \dot{Q}_m(\mathbf{n}, t) = & \sum_{\mu=1}^{K-1} a_\mu(\mathbf{n} - \mathbf{v}_\mu) Q_m(\mathbf{n} - \mathbf{v}_\mu, t) \\ & - Q_m(\mathbf{n}, t) \sum_{\mu=1}^J a_\mu(\mathbf{n}), \end{aligned} \tag{27}$$

with the initial conditions that $Q_m(\mathbf{n}, 0) = P_m(\mathbf{n}, 0)$. This equation resembles Eq (20) except that now we have all propensities, from G_1 and G_2 , appearing in the second term.

Next we need to compute the probability p_μ that the μ th reaction in G_1 occurs at time τ . This is given by the probability that the μ th reaction occurs given a specific set \mathbf{n} , multiplied by the probability of having \mathbf{n} , and then summing over all \mathbf{n} . In symbols:

$$p_\mu = \sum_{n_1, \dots, n_N} P_m(\mathbf{n}, \tau) \frac{a_\mu(\mathbf{m}_0, \mathbf{n})}{a(\mathbf{m}_0, \mathbf{n})}, \quad a(\mathbf{m}_0, \mathbf{n}) = \sum_{v=1}^{K-1} a_v(\mathbf{m}_0, \mathbf{n}). \tag{28}$$

We now have everything we need to simulate the evolution of \mathbf{m} via the CME-GA. Here are the steps:

1. Select your initial set \mathbf{m} and initial probability $P_m(\mathbf{n}, 0)$ and hence $Q_m(\mathbf{n}, 0)$ (since $P_m(\mathbf{n}, 0) = Q_m(\mathbf{n}, 0)$).
2. Solve Eqs (20) and (27) and compute $Q(\tau)$ and p_μ for $\mu = 1, \dots, K - 1$ according to Eqs (25) and (28).
3. Compute τ and select the next reaction μ according to:
 - i. Generate a random real number ξ_1 in the range $[0, 1]$ and solve $\xi_1 = Q(\tau)$ for τ
 - ii. Generate another random real number ξ_2 in the range $[0, 1]$ and select the smallest integer k that satisfies the condition $\sum_{j=1}^k p_j > \xi_2$. Set $\mu = k$.
4. Update \mathbf{m} and let $P_m(\mathbf{n}, \tau)$ be the new initial condition for Eqs (20) and (27) if and only if the selected reaction does not effectuate a change in n_i . If the selected reaction changes an n_i by $\pm w_i$ ($w_i = 1, 2, \dots$), the new initial probability $P_m(\mathbf{n}, \tau)$ must be modified according to:

$$P_m(\mathbf{n}, \tau) \rightarrow P_m(\mathbf{n}, \tau)|_{n_i=n_i-w_i} \quad \text{if } n_i \rightarrow n_i + w_i$$

$$P_m(\mathbf{n}, \tau) \rightarrow \hat{L}_i P_m(\mathbf{n}, \tau) \quad \text{if } n_i \rightarrow n_i - w_i,$$

where \hat{L}_i is an operator that transforms $P_m(\mathbf{n}, \tau)$ like so:

$$\begin{aligned} \hat{L}_i P_m(\mathbf{n}, \tau) = & [P_m(\mathbf{n}, \tau)|_{n_i=0} + P_m(\mathbf{n}, \tau)|_{n_i=w_i}] \delta_{n_i,0} + \sum_{j=1}^{w_i-1} P_m(\mathbf{n}, \tau)|_{n_i=j} \delta_{n_i,j} \\ & + P_m(\mathbf{n}, \tau)|_{n_i=n_i+w_i} \prod_{j=0}^{w_i-1} (1 - \delta_{n_i,w_j}). \end{aligned}$$

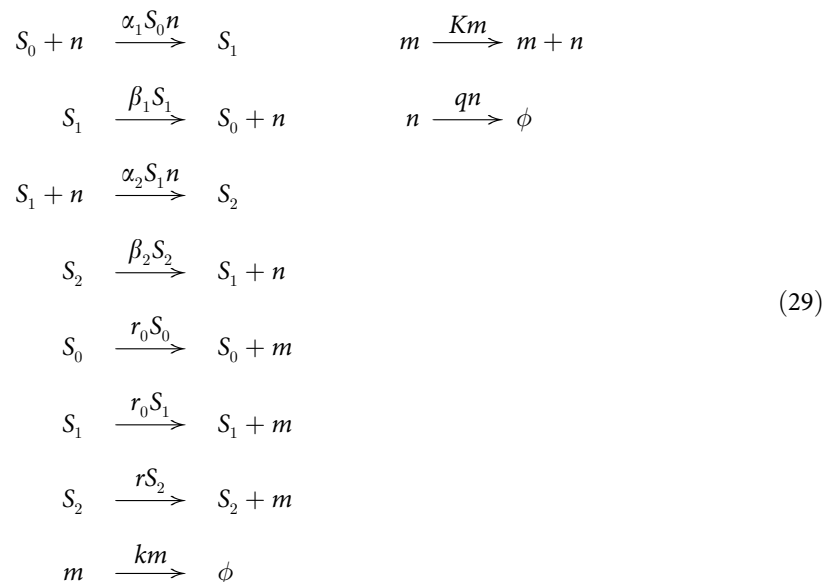
If more than one species of n is affected by a reaction in G_1 , the above transformation must be applied to all of them, e. g. $\hat{L}_1 \hat{L}_2 \dots P_{\mathbf{m}}(\mathbf{n}, \tau)$.

This procedure allows one to simulate stochastically a part of a reaction network, i. e. \mathbf{m} , at the expense of losing information about the rest of the network, i. e. \mathbf{n} . However, the tractability of this algorithm will depend on the system of interest and on the way in which it is partitioned. If, for instance, a particular choice of partition leads to Eqs (20) and/or (27) being too complicated to solve efficiently, the speed of this algorithm may end up being inferior to other stochastic algorithms. Another obstacle to efficiency is having to solve $\xi_1 = Q(\tau)$ for τ , which, depending on the particulars of $P(\tau)$, might be a difficult task. One way to solve for τ is to use a minimization algorithm that optimizes the measure $(\xi_1 - Q(\tau))^2$ with respect to τ . This, however, will likely require a number of steps, calling into question the efficiency of this algorithm. Same can be said of Eq (28), in which the summation(s) may or may not have a closed form. These potential difficulties require not only that the system be partitioned wisely, but also that some of the steps above be simplified/approximated. In the next section we will test the CME-GA on two biological systems and see how it may be applied effectively and accurately.

Results

The genetic switch

Let us consider a single-gene motif with positive autoregulation and a promoter cooperativity of 2. This system can exhibit very large noise [27] due to its positive feedback, and is therefore of interest in systems biology. The simplest yet realistic version of this gene motif is the one described by the following reactions (see Fig 1A):



where m stands for the copy number of mRNA molecules, n for the copy number of proteins, and S_0, S_1 and S_2 label the promoter states: unoccupied, occupied by one protein, and occupied by two proteins, respectively. The reaction propensities appear above each arrow. The parameters $\alpha_i, \beta_i, r, r_0, K, k, q$ are the reaction frequencies per molecule. One can get a sense for the dynamics of this system by looking at the evolution of its averaged variables, $\bar{S}_0, \bar{S}_1, \bar{S}_2, \bar{m}$ and

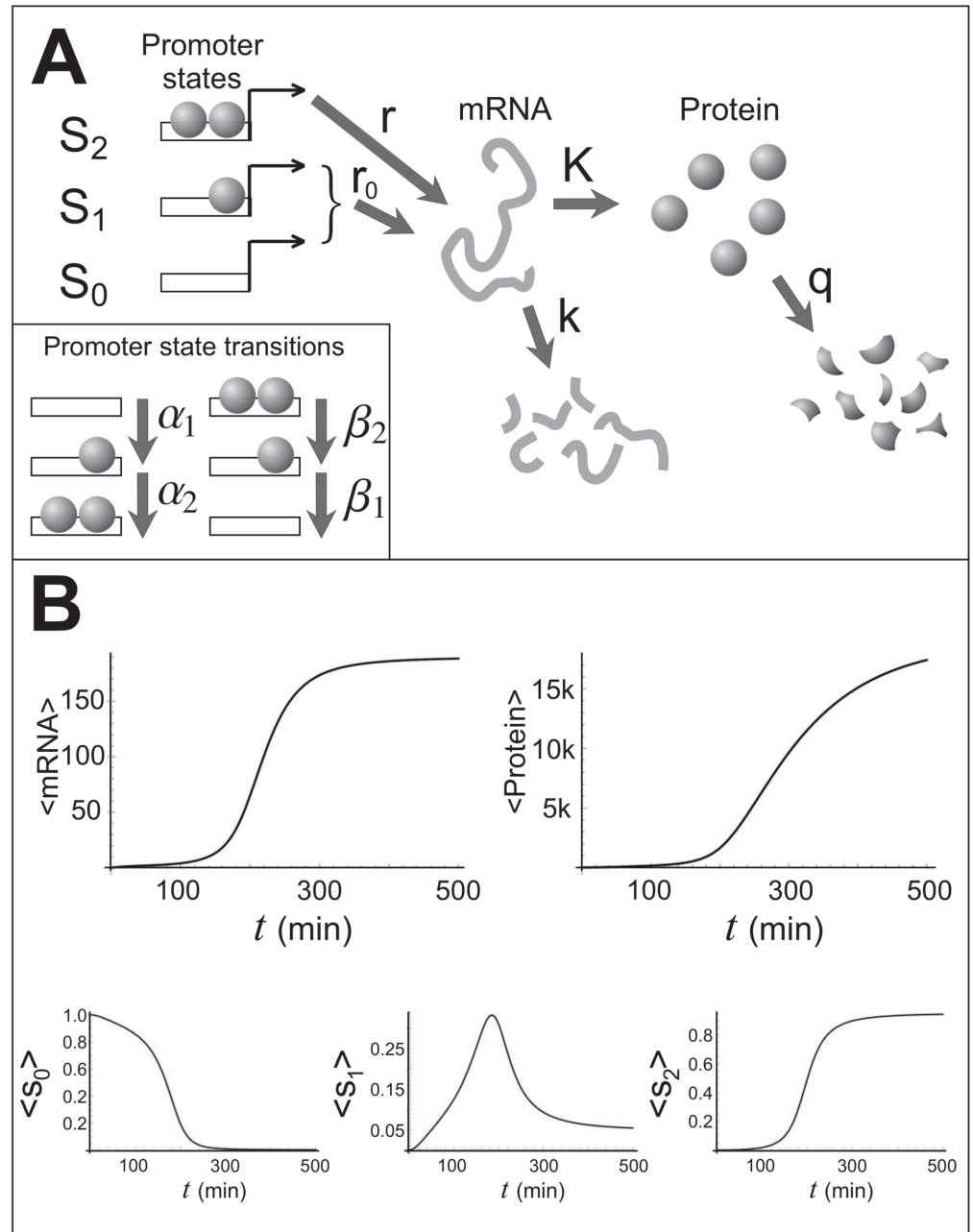


Fig 1. The genetic switch: reactions and dynamics. A) A schematic of the genetic switch. B) Dynamics of average mRNA, m , protein, n , and the three states of promoter, S_0 , S_1 and S_2 . The chosen reaction frequencies in inverse minutes were: $\alpha_1 = \alpha_2 = 0.001$, $\beta_1 = \beta_2 = 1$, $r_0 = 0.1$, $r = 10$, $K = 1$, $k = 0.05$, $q = 0.01$.

doi:10.1371/journal.pone.0149909.g001

\bar{n} , given by the set of ordinary differential equations (ODE)

$$\begin{aligned} \dot{\bar{S}}_1 &= \alpha_1 \bar{n}(1 - \bar{S}_1 - \bar{S}_2) + \beta_2 \bar{S}_2 - (\alpha_2 \bar{S}_1 \bar{n} + \beta_1) \bar{S}_1, \\ \dot{\bar{S}}_2 &= \alpha_2 \bar{S}_1 \bar{n} - \beta_2 \bar{S}_2, \\ \dot{\bar{m}} &= r \bar{S}_2 + r_0 - k \bar{m}, \\ \dot{\bar{n}} &= K \bar{m} - q \bar{n}, \end{aligned} \tag{30}$$

where $\bar{S}_0 = 1 - \bar{S}_1 - \bar{S}_2$. Fig 1B shows the dynamics of $\bar{S}_0, \bar{S}_1, \bar{S}_2, \bar{m}$ and \bar{n} .

Let us now employ the CME-GA to study the stochastic properties of a part of this system. Notice that the reactions were organized into two columns. The reactions in the left column effectuate a change in either m or in both S_i and n together, but not exclusively in n ; the reactions that change n only appear in the right column. Hence, referring to the notation in the previous section, the left column represents the set $\mathbf{m} = \{S_0, S_1, S_2, m\}$, while the right column represents the set $\mathbf{n} = \{n\}$. Hence, the two equations, Eqs (20) and (27), reduce to:

$$\dot{P}_m(n, t) = KmP_m(n - 1, t) + q(n + 1)P_m(n + 1, t) - P_m(n, t)(Km + qn), \tag{31}$$

and

$$\begin{aligned} \dot{Q}_m(n, t) &= KmQ_m(n - 1, t) + q(n + 1)Q_m(n + 1, t) \\ &\quad - Q_m(n, t)[(\alpha_1 S_0 + \alpha_2 S_1)n + \beta_1 S_1 + \beta_2 S_2 + r_0 + rS_2 + (K + k)m + qn]. \end{aligned} \tag{32}$$

It is easy to check that, provided the initial probability $P_m(n, 0)$ is a Poisson distribution, the solution to both, Eqs (31) and (32), is also a Poisson distribution. Thus

$$P(n, t) = e^{-\lambda(t)} \frac{\lambda(t)^n}{n!}, \tag{33}$$

and

$$Q(n, t) = e^{-g(t)} \frac{h(t)^n}{n!}, \tag{34}$$

where for notational simplicity the indexes \mathbf{m} were omitted. Inserting $P(n, t)$ and $Q(n, t)$ into Eqs (31) and (32) respectively yields

$$\dot{\lambda} = Km - q\lambda, \tag{35}$$

$$\dot{h} = Km - \tilde{q}h, \tag{36}$$

$$\dot{g} = \tilde{K} - qh, \tag{37}$$

where $\tilde{q} = \alpha_1 S_0 + \alpha_2 S_1 + q$ and $\tilde{K} = \beta_1 S_1 + \beta_2 S_2 + r_0 + rS_2 + (K + k)m$. With the initial

conditions $Q(n, 0) = P(n, 0)$, and hence $g(0) = h(0) = \lambda(0)$, we obtain

$$\lambda(t) = \left[\lambda(0) - \frac{Km}{q} \right] e^{-qt} + \frac{Km}{q}, \tag{38}$$

$$h(t) = \left[\lambda(0) - \frac{Km}{\tilde{q}} \right] e^{-\tilde{q}t} + \frac{Km}{\tilde{q}}, \tag{39}$$

$$g(t) = \left[\lambda(0) - \frac{Km}{\tilde{q}} \right] \left(\frac{q}{\tilde{q}} \right) (e^{-\tilde{q}t} - 1) + \left[\tilde{K} - \frac{Kmq}{\tilde{q}} \right] t + \lambda(0). \tag{40}$$

The probability distribution for τ acquires a closed form:

$$Q(\tau) = \sum_{n=0}^{\infty} Q(n, \tau) = e^{-g(\tau)+\lambda(\tau)}. \tag{41}$$

Referring to [Eq \(28\)](#), we can readily compute the probabilities for each reaction in the left column to occur. In the order in which they appear in [Eq \(29\)](#), they read:

$$\begin{aligned} p_1 &= S_0 \left[1 - \left\langle \frac{a_1}{n + a_1} \right\rangle \right] \\ p_2 &= S_1 \frac{\beta_1}{\alpha_2} \left\langle \frac{1}{n + a_1} \right\rangle \\ p_3 &= S_1 \left[1 - \left\langle \frac{a_2}{n + a_2} \right\rangle \right] \\ p_4 &= S_2 \frac{\beta_2}{\beta_2 + r + km} \\ p_5 &= S_0 \frac{r_0}{\alpha_1} \left\langle \frac{1}{n + a_1} \right\rangle + S_1 \frac{r_0}{\alpha_2} \left\langle \frac{1}{n + a_2} \right\rangle + S_2 \frac{r}{\beta_2 + r + km} \\ p_6 &= S_0 \frac{km}{\alpha_1} \left\langle \frac{1}{n + a_1} \right\rangle + S_1 \frac{km}{\alpha_2} \left\langle \frac{1}{n + a_2} \right\rangle + S_2 \frac{km}{\beta_2 + r + km}, \end{aligned} \tag{42}$$

where

$$a_1 = \frac{r_0 + km}{\alpha_1}, \quad a_2 = \frac{\beta_1 + r_0 + km}{\alpha_2}, \tag{43}$$

and $\langle \dots \rangle$ stands for $\sum_{n=0}^{\infty} P(n, \tau) (\dots)$. To work out the exact expressions for all the p_s we can employ the formula

$$\sum_{n=0}^{\infty} \frac{b^n}{(a+n)n!} = B(a, b), \quad B(a, b) = (-b)^{-a} [\Gamma(a) - \Gamma(a, -b)], \tag{44}$$

with $\Gamma(\cdot)$ being the Gamma function. Thus, we have

$$\left\langle \frac{1}{n + a_i} \right\rangle = e^{-\lambda(\tau)} B(a_i, \lambda(\tau)). \tag{45}$$

We now have all ingredients to run the CME-GA. However, before we do, let us consider the computational expenses involved in performing all the steps. In particular, solving [Eq \(41\)](#) for τ may slow down the algorithm considerably, as it requires an optimization algorithm of

some kind. One way to avoid this is to solve Eq (41) approximately by assuming that the solution, i. e. τ , is small and expand $\ln Q(\tau)$ up to τ^2 . The equation for τ then becomes

$$\ln \xi_1 = \ln Q(\tau) \approx -b_1 \tau + b_2 \tau^2 / 2. \tag{46}$$

where

$$b_1 = \tilde{K} - Km + (\tilde{q} - q)\lambda(0), \quad b_2 = [Km - \tilde{q}\lambda(0)](\tilde{q} - q). \tag{47}$$

Writing $\tau = \tau_0 + \tau_1$, where τ_0 satisfies $\ln 1/\xi = b_1 \tau_0$ and τ_1 is a correction, we obtain the ratio

$$\frac{\tau_1}{\tau_0} = \frac{1}{2} \frac{(b_2/b_1) \ln(1/\xi_1)}{b_1 - (b_2/b_1) \ln(1/\xi_1)}. \tag{48}$$

Thus, as long as $\tau_1/\tau_0 < \epsilon$, where ϵ is some small number, e. g. 0.001, we may write

$$\tau = \frac{1}{b_1} \ln \frac{1}{\xi_1}. \tag{49}$$

This equation is identical in structure to Eq (18), especially when we see that $b_1 = (\alpha_1 S_0 + \alpha_2 S_1)\lambda(0) + \beta_1 S_1 + \beta_2 S_2 + r_0 + rS_2 + km$, which is just the sum of all reactions in the left column but with n replaced by its average, $\lambda(0)$. The condition $\tau_1/\tau_0 < \epsilon$ must be incorporated into the CME-GA and checked for each cycle; if it fails, Eq (41) must be solved by some other means, e. g. an optimization algorithm.

Similarly for the reaction probabilities, we may simplify them by expanding $1/(a_i + n)$ around $n - \langle n \rangle$ and then averaging each term:

$$\left\langle \frac{1}{a_i + n} \right\rangle \approx \frac{1}{a_i + \lambda(0)} + \frac{\lambda(0)}{(a_i + \lambda(0))^3} - \frac{\lambda(0)}{(a_i + \lambda(0))^4} \dots, \tag{50}$$

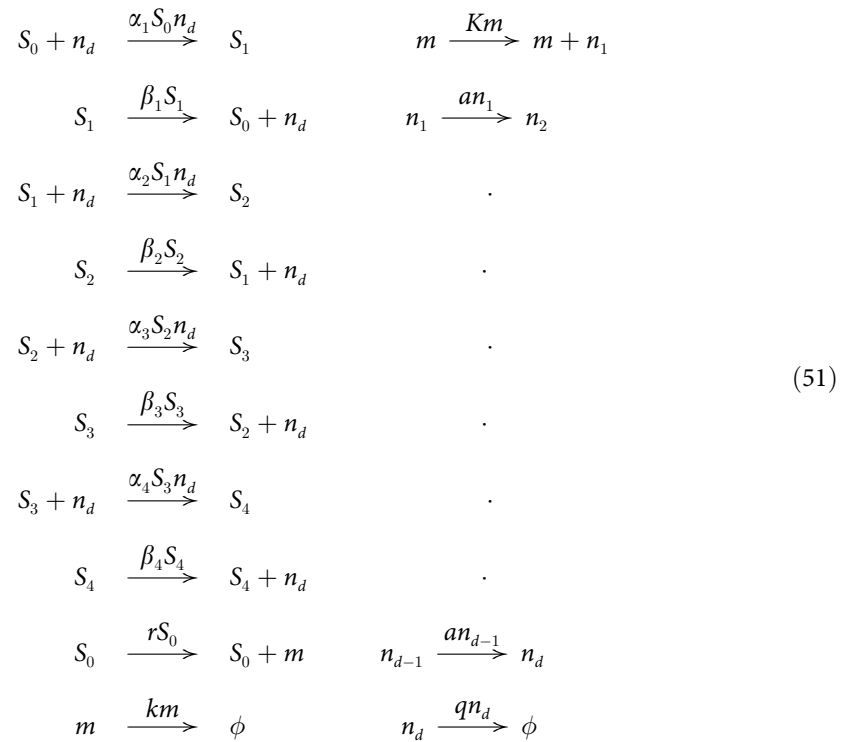
where we used the fact that for a Poisson distribution $\langle n \rangle = \langle (n - \langle n \rangle)^2 \rangle = \langle (n - \langle n \rangle)^3 \rangle$, which in the present case is equivalent to $\lambda(0)$. Here again we need to keep in mind that this approximation may become inaccurate (depending on the number of terms), as for instance when $\lambda(0), a_1 < 1$.

Finally, before running the CME-GA, we need to address its forth step. Remember that expressions Eqs (33) and (34) are the solutions to Eqs (31) and (32) only if their initial distributions are Poissonian. However, when reactions 1–4 in the left column occur, we must add to or subtract from the system one copy of n , and then modify the new initial probability $P(n, \tau)$ according to step 4 of the CME-GA. This however will render Eqs (33) and (34) incorrect. There may be ways to overcome this problem; however, in the present case, with $\alpha_1, \alpha_2 < 1$, we are justified in ignoring it. Running the CME-SSA with the parameters of Fig 1 leads to the results of Fig 2.

The Griffith model of a genetic oscillator

Consider now a larger network consisting of a promoter with three states, S_0, S_1 , and S_2 , an mRNA, and a protein that can be in several conformations, e. g. when undergoing a multi-step phosphorylation [28], such that in its final conformation the protein can bind to its promoter

and repress it (see Fig 3A). The reactions for this system are as follows:



Here again m refers to the mRNA copy number, n_i to the protein copy number, where the indexes $i = 1, \dots, d$ label different protein conformations. The differential equations for $\bar{S}_0, \bar{S}_1, \bar{S}_2, \bar{S}_3, \bar{S}_4, \bar{m}$ and \bar{n}_i read:

$$\begin{aligned}
 \dot{\bar{S}}_1 &= \alpha_1 \bar{n}_d \bar{S}_0 + \beta_2 \bar{S}_2 - (\alpha_2 \bar{n}_d + \beta_1) \bar{S}_1, \\
 \dot{\bar{S}}_2 &= \alpha_2 \bar{n}_d \bar{S}_1 + \beta_3 \bar{S}_3 - (\alpha_3 \bar{n}_d + \beta_2) \bar{S}_2, \\
 \dot{\bar{S}}_3 &= \alpha_3 \bar{n}_d \bar{S}_2 + \beta_4 \bar{S}_4 - (\alpha_4 \bar{n}_d + \beta_3) \bar{S}_3, \\
 \dot{\bar{S}}_4 &= \alpha_4 \bar{n}_d \bar{S}_3 - \beta_4 \bar{S}_4, \\
 \dot{\bar{m}} &= r\bar{S}_0 - k\bar{m}, \\
 \dot{\bar{n}}_1 &= K\bar{m} - a\bar{n}_1, \\
 \dot{\bar{n}}_2 &= a(\bar{n}_1 - \bar{n}_2), \\
 &\cdot \\
 &\cdot \\
 &\cdot \\
 \dot{\bar{n}}_d &= a\bar{n}_{d-1} - q\bar{n}_d, \\
 \bar{S}_0 &= 1 - \bar{S}_1 - \bar{S}_2 - \bar{S}_3 - \bar{S}_4
 \end{aligned} \tag{52}$$

For some values of its reaction rates, this system can exhibit sustained oscillations, as shown in Fig 3.

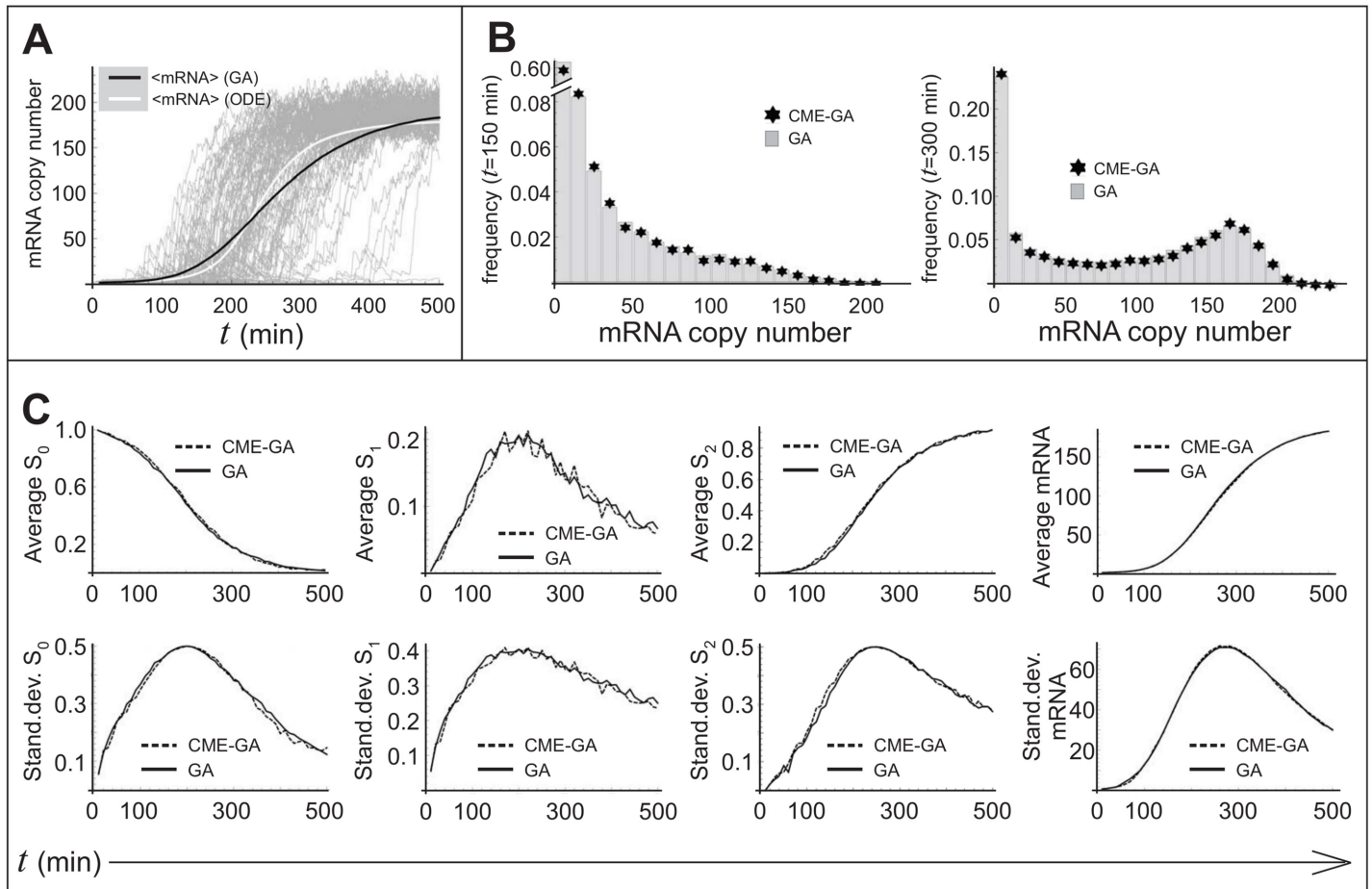


Fig 2. Comparison of the CME-GA with the GA—Genetic switch. **A**) A superposition of 100 realizations generated by the GA. The black solid curve represents the average of 500 realizations, while the white curve is the solution of Eq (30) for m . **B**) Probability distributions for m at $t = 150\text{min}$ and $t = 300\text{min}$, showing the match between the CME-GA (asterisk) and the GA (bar) constructed from an ensemble size of 10000. **C**) Comparison between the CME-GA and the GA of the averages and standard deviations of m , S_0 , S_1 and S_2 . The ensemble size was 1000.

doi:10.1371/journal.pone.0149909.g002

The solutions to Eqs (20) and (27) for the right column are products of Poisson distributions,

$$P(\mathbf{n}, t) = \prod_{i=1}^d \frac{\lambda_i(t)^{n_i}}{n_i!} e^{-\lambda_i(t)} \tag{53}$$

$$Q(\mathbf{n}, t) = e^{-g(t)} \prod_{i=1}^d \frac{h_i(t)^{n_i}}{n_i!}, \tag{54}$$

provided again that their initial distributions are also Poisson. Inserting these into Eqs (20) and

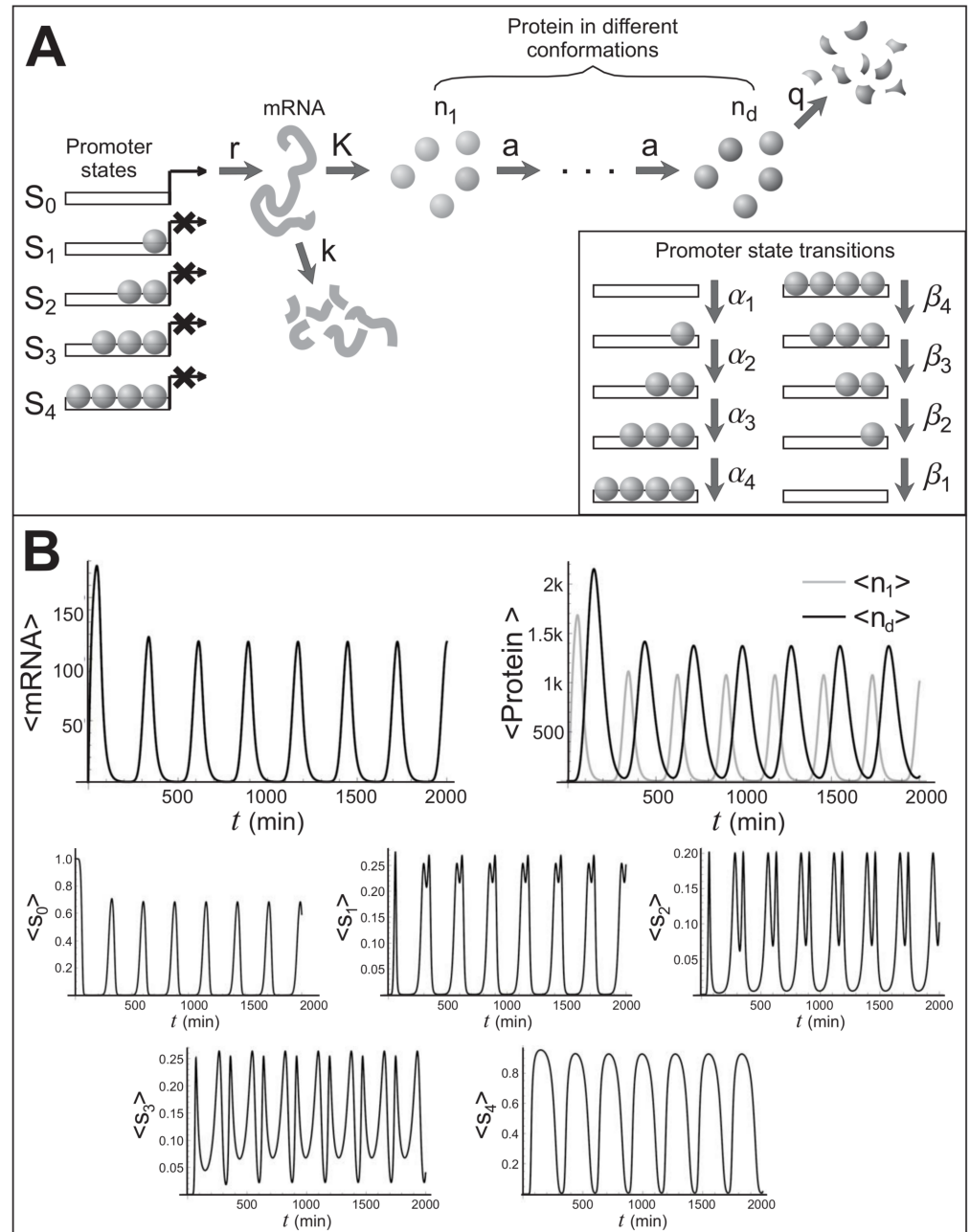


Fig 3. The Griffith model: reactions and dynamics. A) A schematic of the Griffith model. B) Dynamics of average mRNA, m , protein, n , and the five states of promoter, S_0, S_1, S_2, S_3 and S_4 . The chosen reaction frequencies in inverse minutes were: $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.01, \beta_1 = \beta_2 = \beta_3 = \beta_4 = 1, r = 10, K = 1, k = 0.05, q = 0.05, a = 0.1$. The number of protein conformations, d was set to 10.

doi:10.1371/journal.pone.0149909.g003

(27) leads to differential equations for the λ s:

$$\begin{aligned} \dot{\lambda}_1 &= Km - a\lambda_1, \\ \dot{\lambda}_2 &= a(\lambda_1 - \lambda_2), \\ &\vdots \\ \dot{\lambda}_d &= a\lambda_{d-1} - q\lambda_d, \end{aligned} \tag{55}$$

and another almost identical set for the h s but with q replaced with $\tilde{q} = \alpha_1 S_0 + \alpha_2 S_1 + \alpha_3 S_2 + \alpha_4 S_3$, and

$$\dot{g} = \tilde{K} - qh_d, \tag{56}$$

with $\tilde{K} = \beta_1 S_1 + \beta_2 S_2 + \beta_3 S_3 + \beta_4 S_4 + rS_0 + (K + k)m$. The solutions for the λ s, h s and g are

$$\begin{aligned} \lambda_i(t) &= \frac{mK}{a} - e^{-at} \sum_{j=0}^{i-1} \frac{1}{j!} \left(\frac{mK}{a} - \lambda_{i-j}(0) \right) (at)^j, \quad i < d, \\ \lambda_d(t) &= \sum_{j=0}^{d-2} \frac{e^{-qt}}{j!} \left(\frac{Km}{a} - \lambda_{d-j-1}(0) \right) \left(\frac{a}{a-q} \right)^{j+1} [\Gamma(j+1) - \Gamma(j+1, (a-q)t)] \\ &\quad + \frac{Km}{q} + \left(\lambda_d(0) - \frac{Km}{q} \right) e^{-qt}, \tag{57} \\ h_i(t) &= \lambda_i(t) \Big|_{q \rightarrow \tilde{q}}, \quad \forall i, \\ g(t) &= Kmt - q \int_0^t h_d(t') dt'. \end{aligned}$$

Following the steps detailed in the previous section, we arrive at the same approximation for τ :

$$\tau = \frac{1}{b_1} \ln \frac{1}{\xi_1}, \quad b_1 = (\alpha_1 S_0 + \alpha_2 S_1) \lambda_d(0) + \beta_1 S_1 + \beta_2 S_2 + r_0 + rS_2 + km, \tag{58}$$

with the same error parameter as in Eq (48), τ_1/τ_0 , but now with $b_2 = (\tilde{q} - q)(a\lambda_{d-1}(0) - q\lambda_d(0))$.

The propensities p_1 to p_{10} are given by

$$\begin{aligned}
 p_1 &= S_0 \left[1 - \left\langle \frac{a_1}{n_d + a_1} \right\rangle \right] \\
 p_2 &= S_1 \frac{\beta_1}{\alpha_2} \left\langle \frac{1}{n_d + a_1} \right\rangle \\
 p_3 &= S_1 \left[1 - \left\langle \frac{a_2}{n_d + a_2} \right\rangle \right] \\
 p_4 &= S_2 \frac{\beta_2}{\alpha_3} \left\langle \frac{1}{n_d + a_2} \right\rangle \\
 p_5 &= S_2 \left[1 - \left\langle \frac{a_3}{n_d + a_3} \right\rangle \right] \\
 p_6 &= S_3 \frac{\beta_3}{\alpha_4} \left\langle \frac{1}{n_d + a_3} \right\rangle \\
 p_7 &= S_3 \left[1 - \left\langle \frac{a_4}{n_d + a_4} \right\rangle \right] \\
 p_8 &= S_4 \frac{\beta_4}{\beta_4 + km} \\
 p_9 &= S_0 \frac{r_0}{\alpha_1} \left\langle \frac{1}{n + a_1} \right\rangle \\
 p_{10} &= km \left[\frac{S_4}{km + \beta_4} + \sum_{i=1}^4 \frac{S_{i-1}}{\alpha_i} \left\langle \frac{1}{n_d + a_i} \right\rangle \right],
 \end{aligned} \tag{59}$$

with

$$a_1 = \frac{km}{\alpha_1}, \quad a_2 = \frac{km + \beta_1}{\alpha_2}, \quad a_3 = \frac{km + \beta_2}{\alpha_3}, \quad a_4 = \frac{km + \beta_4}{\alpha_4}, \tag{60}$$

Finally, making the approximation that reactions p_1 - p_8 do not alter $P_{\mathbf{m}}(\mathbf{n}, 0)$ significantly, and expanding the terms $\langle \dots \rangle$ in $n_d - \lambda_d(\tau)$, we can run the CME-GA. The results are shown in Fig 4.

Discussion

The method presented herein provides a means of stochastically simulating a reaction sub-network. Because most of the information about the rest of the network is lost, its usefulness is limited to cases where partial information about a network is sufficient. The two examples discussed above illustrate the accuracy of this method. In terms of efficiency, Table 1 shows the computation times of the CME-GA and five other stochastic simulation algorithms that were used to simulate the two models. It is clear that the CME-GA is significantly more efficient than any of the other algorithms.

Important to notice is the relation between the speed of CME-GA and the abundance of those molecular species that appear in the CME Eq (20). Since the speed of the GA scales with the number of species and their abundance, and the CME does not (at least when it can be solved exactly), different partitions will lead to different speeds. If, for instance, we had chosen to partition either of the example systems such that the mRNA appeared in the CME Eq (20) instead of the protein, and it was the protein that was simulated via the GA, the computational time would have been drastically increased. Therefore, a network to be partitioned must be

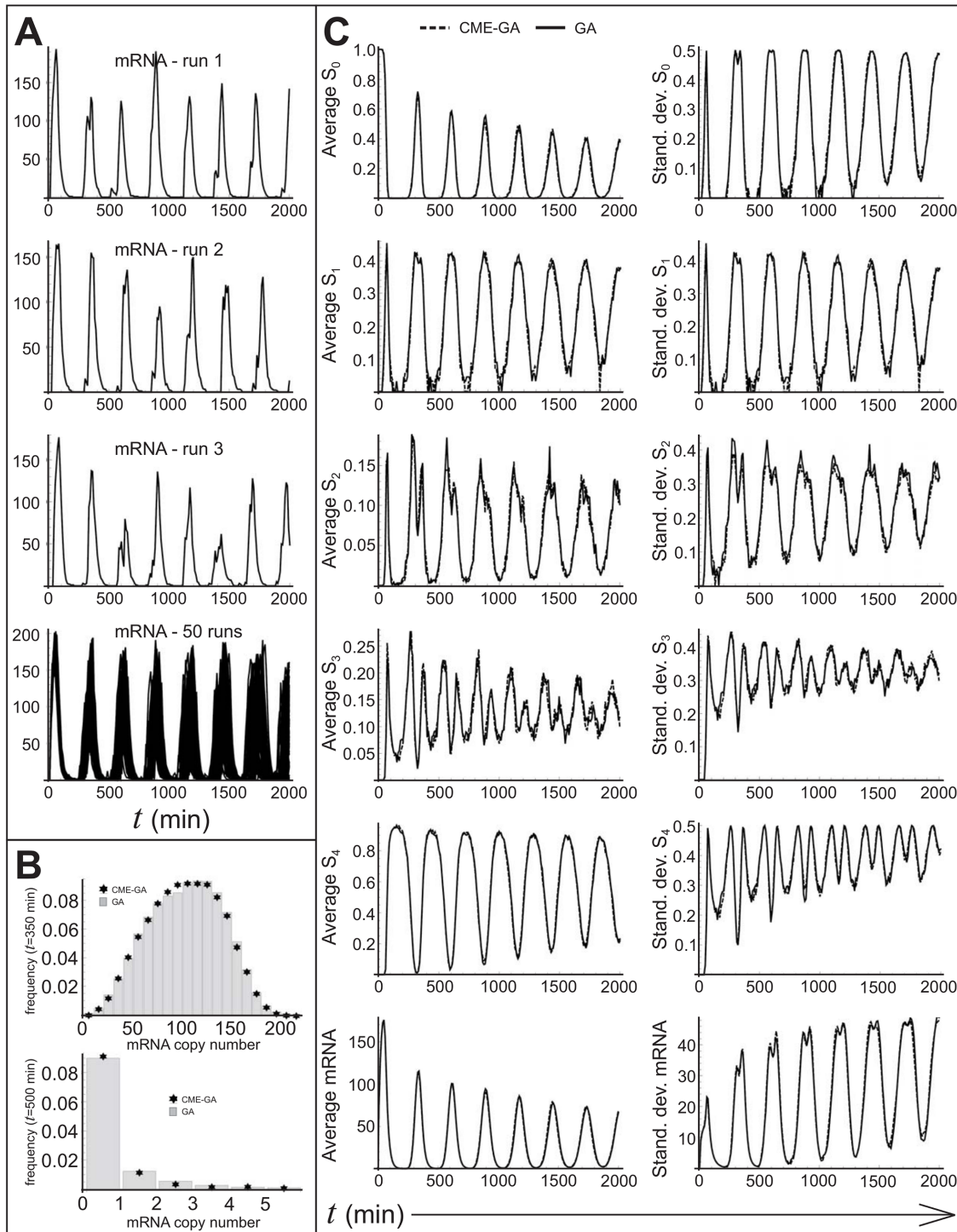


Fig 4. Comparison of the CME-GA with the GA—the Griffith model. **A)** Graphs 1–3 show individual realizations generated by the GA. The fourth graph shows a superposition of 50 realizations. **B)** Probability distributions for m at $t = 350$ min and $t = 500$ min, showing the match between the CME-GA (asterisk) and the GA (bar) constructed from an ensemble size of 10000. **C)** Comparison between the CME-GA and the GA of the averages and standard deviations of m , S_0 , S_1 , S_2 , S_3 and S_4 . The ensemble size was 1000.

doi:10.1371/journal.pone.0149909.g004

Table 1. State of the art vs. CME-GA.

Algorithm	Genetic switch	Griffith model
Gillespie direct	37	300
Gibson-Bruck	60	297
Tau-leaping simple	60	304
Tau-leaping complex	44	126
CME-GA	2	14

Comparison of computing times (rounded to nearest second) of the four standard algorithms with the CME-GA hybrid. For both, the genetic switch and the Griffith model, the ensemble size was 1000. The stopping time was: 500 for the former and 2000 for the latter.

doi:10.1371/journal.pone.0149909.t001

done so wisely. This of course may be in conflict with the user’s desire to simulate a particular set of species. Consequently, if the CME-GA is to remain superior in speed to others, it must be limited to such partitions where the species with large molecular numbers are placed in the non-simulated sub-network.

Although Eqs (20) and (27), which are necessary for the CME-GA to run, were derived exactly, without any assumptions, in practice they may not always be tractable and will require approximations. However, bisecting a system into two groups as proscribed above necessarily renders the CME less complex (Eq (1) vs. Eq (20)) and hence more manageable. It should be noted that as the simulated reaction network grows larger, the time between reactions becomes shorter. This means that Eqs (20) and (27), given a large sub-network (i. e. G_1), will only need to be solved for very short times. Another relief may come from moment closure methods [29–31]: since the propensities in Eq (28) can be expressed as a sum of statistical moments (see Eq (50)), one needs only to solve the set of equations for a few moments, instead of the full CME for the sub-system G_2 ; and same goes for $Q_m(\mathbf{n}, t)$. This last approach might in fact be the most promising way of extending our algorithm to more complex systems.

Lastly, it bears mentioning that although the information about the sub-system G_2 is lost, it may not always be completely lost for all types of systems. In both examples discussed above, when the original CME, Eq (1), is multiplied by the variables in G_2 and summed over all variables, one ends up with

$$\dot{\bar{n}} = K\bar{m} - q\bar{n}, \tag{61}$$

for the genetic switch, and

$$\begin{aligned} \dot{\bar{n}}_1 &= K\bar{m} - a\bar{n}_1, \\ \dot{\bar{n}}_2 &= a(\bar{n}_1 - \bar{n}_2), \\ &\vdots \\ &\vdots \\ \dot{\bar{n}}_d &= a\bar{n}_{d-1} - q\bar{n}_d \end{aligned} \tag{62}$$

for the Griffith model. And because \bar{m} is computed via the CME-GA, all the above equations have a closed form.

Acknowledgments

This research was supported by: the Interuniversity Attraction Poles program of the Belgian Science Policy Office, under grant IAP P7-35 (www.belspo.be); and the Onderzoeksraad of the Vrije Universiteit Brussel (www.vub.ac.be) through the strategic research program “Applied Physics and Systems Biology”. I would like to thank Ekaterina Ejikova for her technical support.

Author Contributions

Conceived and designed the experiments: JA. Performed the experiments: JA. Analyzed the data: JA. Contributed reagents/materials/analysis tools: JA. Wrote the paper: JA.

References

1. Doob JL. Topics in the Theory of Markoff Chains. Transactions of the American Mathematical Society 1945; 58 (3), 455–473 doi: [10.2307/1990339](https://doi.org/10.2307/1990339)
2. Gillespie DT. Exact Stochastic Simulation of Coupled Chemical Reactions. J. Phys. Chem. 1977; 81 (25), 2340–2361 doi: [10.1021/j100540a008](https://doi.org/10.1021/j100540a008)
3. Gibson MA, Bruck J. Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels. J. Phys. Chem. 2000; 104(9), 1876–1889 doi: [10.1021/jp993732q](https://doi.org/10.1021/jp993732q)
4. Cao Y, Li H, Petzold L. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. J. Chem. Phys. 2004; 121, 4059 PMID: [15332951](https://pubmed.ncbi.nlm.nih.gov/15332951/)
5. Gillespie DT. Approximate accelerated stochastic simulation of chemically reacting systems. J. Chem. Phys. 2001; 115(4), 1716 doi: [10.1063/1.1378322](https://doi.org/10.1063/1.1378322)
6. Cao Y, Gillespie DT, Petzold LR. Avoiding negative populations in explicit Poisson tau-leaping. J. Chem. Phys. 2005; 123(5), 054104 doi: [10.1063/1.1992473](https://doi.org/10.1063/1.1992473) PMID: [16108628](https://pubmed.ncbi.nlm.nih.gov/16108628/)
7. Cao Y, Gillespie DT, Petzold LR. Efficient step size selection for the tau-leaping simulation method. J. Chem. Phys. 2005; 124(4), 044109 doi: [10.1063/1.2159468](https://doi.org/10.1063/1.2159468)
8. Rathinam M, Samadpour HE. Reversible-equivalent-monomolecular tau: A leaping method for small number and stiff $\tau_{1/2}$ stochastic chemical systems. J. Comp. Phys. 2007; 224(2), 897–923 doi: [10.1016/j.jcp.2006.10.034](https://doi.org/10.1016/j.jcp.2006.10.034)
9. Walczak AM, Mugler A, Wiggins CH. Analytic methods for modeling stochastic regulatory networks. Methods Mol Biol. 2012; 880, 273–322 doi: [10.1007/978-1-61779-833-7_13](https://doi.org/10.1007/978-1-61779-833-7_13) PMID: [23361990](https://pubmed.ncbi.nlm.nih.gov/23361990/)
10. Ramsey S, Orrell D, Bolouri H. Dizzy: stochastic simulation of large-scale genetic regulatory networks. J. Bioinform. Comput. Biol. 2005; 3(2), 415–36 doi: [10.1142/S0219720005001144](https://doi.org/10.1142/S0219720005001144) PMID: [15852513](https://pubmed.ncbi.nlm.nih.gov/15852513/)
11. Mugler A, Walczak AM, Wiggins CH. Spectral solutions to stochastic models of gene expression with bursts and regulation. Phys. Rev. E 2011; 80, 041921 doi: [10.1103/PhysRevE.80.041921](https://doi.org/10.1103/PhysRevE.80.041921)
12. Wolf V, Goel R, Mateescu M, Henzinger TA. Solving the chemical master equation using sliding windows. BMC Systems Biology 2010; 4, 42 doi: [10.1186/1752-0509-4-42](https://doi.org/10.1186/1752-0509-4-42) PMID: [20377904](https://pubmed.ncbi.nlm.nih.gov/20377904/)
13. Albert J, Rooman M. Probability distributions for multimeric systems. J. Math. Biol. 2011; 10.1007/s00285-015-0877-0
14. Munsky B, Khammash M. A multiple time interval finite state projection algorithm for the solution to the chemical master equation. J. Comp. Phys. 2007; 226(1), 818–835
15. Munsky B, Khammash M. Transient analysis of stochastic switches and trajectories with applications to gene regulatory networks. IET Syst Biol. 2008; 2(5), 323–33 doi: [10.1049/iet-syb:20070082](https://doi.org/10.1049/iet-syb:20070082) PMID: [19045827](https://pubmed.ncbi.nlm.nih.gov/19045827/)
16. MacNamara S, Burrage K, Sidje RB. Multiscale Modeling of Chemical Kinetics via the Master Equation. Multiscale Model. Simul. 2007; 6(4), 1146–1168 doi: [10.1137/060678154](https://doi.org/10.1137/060678154)
17. Jahnke T. On Reduced Models for the Chemical Master Equation Multiscale Model. Simul. 2011; 9(4), 1646–1676.
18. Smith S, Cianci C, Grima R. Model reduction for stochastic chemical systems with abundant species. J. Chem. Phys. 2015; 143, 214105 doi: [10.1063/1.4936394](https://doi.org/10.1063/1.4936394) PMID: [26646867](https://pubmed.ncbi.nlm.nih.gov/26646867/)
19. Alfonsi A, Cancès E, Turinici G, Ventura BD, Huisinga W. Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems. ESAIM: Proc. 2005; 14, 1–13
20. Jahnke T, Altmann D. Efficient simulation of discrete stochastic reaction systems with a splitting method. BIT Num Math 2010; 50(4), 797–822 doi: [10.1007/s10543-010-0286-0](https://doi.org/10.1007/s10543-010-0286-0)

21. Hellander A, Lotstedt P. Hybrid method for the chemical master equation. *J. Comp. Phys.* 2007; 227(1), 100–122 doi: [10.1016/j.jcp.2007.07.020](https://doi.org/10.1016/j.jcp.2007.07.020)
22. Burrage K, Tian T, Burrage P. A multi-scaled approach for simulating chemical reaction systems. *Progress in Biophysics & Molecular Biology* 2004; 85, 217–234 doi: [10.1016/j.pbiomolbio.2004.01.014](https://doi.org/10.1016/j.pbiomolbio.2004.01.014)
23. Salis H, Kaznessis Y. Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *J. Chem. Phys.* 2005; 122, 054103 doi: [10.1063/1.1835951](https://doi.org/10.1063/1.1835951)
24. Doi M. Stochastic theory of diffusion-controlled reaction. *Journal of Physics A: Mathematical and General* 1976; 9, 1465 doi: [10.1088/0305-4470/9/9/009](https://doi.org/10.1088/0305-4470/9/9/009)
25. Zel'Dovich YB, Ovchinnikov AA. The mass action law and the kinetics of chemical reactions with allowance for thermodynamic fluctuations of the density. *Soviet Journal of Experimental and Theoretical Physics* 1978; 47, 829
26. Peliti L. Renormalisation of fluctuation effects in the A+A to A reaction. *Journal of Physics A: Mathematical and General* 1986; 19, L365 doi: [10.1088/0305-4470/19/6/012](https://doi.org/10.1088/0305-4470/19/6/012)
27. Albert J, Rooman M. Design Principles of a Genetic Alarm Clock. *PLoS ONE* 2012; 7(11), e47256 doi: [10.1371/journal.pone.0047256](https://doi.org/10.1371/journal.pone.0047256) PMID: [23144809](https://pubmed.ncbi.nlm.nih.gov/23144809/)
28. Griffith JS. Mathematics of Cellular Control Processes. I. Negative feedback to one gene. *J. Theor. Biol.* 1968; 20, 202–208 doi: [10.1016/0022-5193\(68\)90189-6](https://doi.org/10.1016/0022-5193(68)90189-6) PMID: [5727239](https://pubmed.ncbi.nlm.nih.gov/5727239/)
29. Lee CH, Kim K-H, Kim P. A moment closure method for stochastic reaction networks. *J. Phys. Chem.* 2009; 130, 134107 doi: [10.1063/1.3103264](https://doi.org/10.1063/1.3103264)
30. Barzel B., Biham O. Stochastic analysis of dimerization systems. *Phys. Rev. E.* 2009; 80, 031117 doi: [10.1103/PhysRevE.80.031117](https://doi.org/10.1103/PhysRevE.80.031117)
31. Grima R. A study of the accuracy of moment-closure approximations for stochastic chemical kinetics. *J. Phys. Chem.* 2012;13680, 154105