A case study on the use of scale separation-based analytic propagators for parameter inference in stochastic gene regulation

A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation

Justin Feigelman[1,2], Nikola Popović[3], and Carsten Marr[1]

[1]*Helmholtz Zentrum München - German Research Center for Environmental Health, Institute of Computational Biology, Ingolstädter Landstraße 1, 85764 Neuherberg Germany*
[2]*Technische Universität München, Center for Mathematics, Chair of Mathematical Modeling of Biological Systems, Boltzmannstraße 3, 85748 Garching, Germany and*
[3]*University of Edinburgh, School of Mathematics and Maxwell Institute for Mathematical Sciences, James Clerk Maxwell Building, King's Buildings, Peter Guthrie Tait Road, Edinburgh EH9 3FD, United Kingdom*

(Dated: October 2, 2015)

Advances in long-term fluorescent time-lapse microscopy have made it possible to study the expression of individual genes in single cells. In a typical setting, the intensity of one or more fluorescently-labeled proteins is measured at regular time intervals. Such time-courses are inherently noisy due to both measurement noise and intrinsic stochasticity of the underlying gene expression regulation. Fitting stochastic models to time-series data remains a difficult task, partly because analytical and tractable expressions for the transition probabilities cannot usually be derived in closed form.

In the present work, we employ a recently developed approach that is based on geometric singular perturbation theory, as applied to the chemical master equation of a simple two-stage gene expression model, to compute parameter likelihoods using synthetic protein time-series. We study the identifiability of model parameters in this simple setting, and compare the performance of the perturbative (uniform) propagator to a previously published, idealized (zeroth-order) propagator that assumes perfect time-scale separation between degradation of mRNA and protein. We find that both propagators are useful for parameter inference when the scale separation is sufficiently large. However, with decreasing separation, the uniform propagator sometimes yields non-physical negative transition probabilities which render parameter inference difficult. Finally, we discuss the utility of both propagators, and possible extensions thereof, for inference. For computational efficiency, the propagators were implemented in C++ and embedded in Matlab; the code is available upon request.

Keywords: Multi-scale gene expression dynamics. Propagator approximation. Parameter inference. Geometric singular perturbation.

1. INTRODUCTION

Gene expression is a complex and highly regulated multi-step process that is responsible for the timely synthesis of proteins necessary for cellular function. At the molecular level, gene expression is inherently stochastic due to random binding events of transcription factors and the transcriptional machinery, which ultimately leads to mRNA transcription with probabilities depending on the concentration of the reaction educts. Protein synthesis requires a chance encounter of mRNA with ribosomes, and mRNA or protein degradation an encounter with the degradation machinery. Thus, models for gene expression have to capture the stochasticity at both mRNA and protein levels.

A simple, "two-stage" model for stochastic gene expression consists of a constitutively active gene from which an mRNA molecule can be transcribed, and protein, the production of which depends on the instantaneous abundance of mRNA (see Fig. 1A). Both mRNA and protein are subjected to stochastic degradation. Such a qualitative model can be described mathematically as a two-dimensional Markov jump process in the copy numbers of mRNA and protein, with reaction probabilities that are functions of the current state only (hence the Markov property), and suitably chosen kinetic constants [1, 2].

While the two-stage model is easily simulated using stochastic simulation algorithms such as Gillespie's algorithm [3], it is nonetheless a difficult task to derive analytical expressions for the evolution of mRNA and protein copy numbers with time. The Markov process itself obeys the chemical master equation (CME), an infinite-dimensional system of ordinary differential equations, for which no exact (closed-form) solutions are known in general. Numerous approaches exist for the approximate solution of the CME, such as the linear noise approximation [4], a second-order Taylor series expansion in the system size of the reaction volume; moment equations and variants thereof [5, 6], which capture an arbitrary number of statistical moments of the stochastic process; finite state projection [7], a truncation of the state-space of possible copy number combinations, and many others (for an overview, see

[8]). We further note that this particular model has been studied using a variety of analytical and computational techniques, see e.g. [9–12] or [13] for a review of related modelling approaches.

An alternative analytical approach was developed by Shahrezaei and Swain [2], wherein it is assumed that mRNA molecules decay much faster than protein, a realistic assumption in many prokaryotic cells. In the limit of a perfect scale separation in which the decay of mRNA is instantaneous, the CME underlying the two-stage model can be solved analytically by the introduction of a generating function. The latter then obeys a first-order linear partial differential equation, the solution of which can be obtained via the method of characteristics. The resulting analytical expression for the general time-dependent joint probability density of mRNA and protein, called the propagator of the system, is of great utility for understanding its dynamics in time. However, it is not valid when the assumption of scale separation is violated, as is commonly the case for eukaryotic cells. In recent work [14], the procedure developed in [2] was extended to capture departure from the assumption of perfect scale separation: the ratio of degradation rates of protein and mRNA, denoted ε, was taken to be small and positive instead of zero, as was the case in [2]. The presence of the (singular) perturbation parameter ε allows for the application of asymptotic techniques, such as geometric singular perturbation theory [15] and matched asymptotic expansions [16].

In the present case study, we explore the utility of this newly developed perturbative approach for propagator-based parameter inference in systems with varying degrees of scale separation. Specifically, our goal is to estimate molecular parameters in the model from observations of protein abundance only. Trajectories are simulated via Gillespie's stochastic simulation algorithm in a parameter regime in which mRNA and protein are produced continuously, *i.e.*, not in translational bursts. The protein time-courses are sampled at regular time intervals, thus mimicking a typical time-lapse fluorescence microscopy setup [17, 18]. While fluorescence microscopy yields only time-series for the intensity, these can nonetheless be converted into absolute protein numbers if a calibration factor of molecules per unit intensity can be estimated, see e.g. [19]. We note that mRNA time-courses are not observed, and that they are hence not used for parameter inference.

The zeroth-order propagator obtained by setting $\varepsilon = 0$ [2] is then compared to a first-order propagator (in $\varepsilon > 0$) that is uniformly valid both on short and on long time-scales [14], in terms of the ability of each to capture the correct parameters – *i.e.*, the kinetic constants in the underlying chemical reaction network – in the two-stage model for gene expression. For comparison, both propagators are also contrasted with an approximate solution of the CME that is computed using a finite state projection. A number of simplifying assumptions are made; notably, we ignore impeding factors such as measurement noise, uncertainty in the conversion from fluorescence intensity to protein numbers or low sampling frequency of fluorescent signal. Rather, our focus in this case study is on assessing the general efficiency and accuracy of the propagator-based approach for parameter inference.

2. METHODS

2.1. Two-stage Gene Expression Model

We model gene expression as a two-stage process, whereby DNA is transcribed to mRNA, which is then translated into protein (see Fig. 1A). Denoting the probability of observing m molecules of mRNA and n molecules of protein in the system at time τ by $P_{m,n}(\tau)$, we find that the latter evolves according to the non-dimensionalized CME [2, 4]

$$\begin{aligned}\frac{\partial P_{m,n}}{\partial \tau} &= a(P_{m-1,n} - P_{m,n}) \\ &\quad + \gamma b m(P_{m,n-1} - P_{m,n}) \\ &\quad + \gamma[(m+1)P_{m+1,n} - mP_{m,n}] \\ &\quad + [(n+1)P_{m,n+1} - nP_{m,n}]. \end{aligned} \tag{1}$$

Here, m and n denote mRNA and protein copy numbers, respectively, a is the non-dimensional transcription rate and b is the non-dimensional translation rate, while the degradation rates of mRNA and protein are given by γ and 1, respectively (cf. Fig. 1A). Finally, τ denotes a suitably non-dimensionalized time variable.

As in [2, 14], we define the perturbation parameter $\varepsilon = \gamma^{-1}$ here. It follows that for ε sufficiently small, the dynamics of Eq. (1) will vary on two distinct time-scales: the long-term behavior of the system is naturally described on the "slow" τ-scale, while the "fast" transients evolve according to the rescaled time $t := \frac{\tau}{\varepsilon}$.

2.2. Propagator Expressions

In this section, we collect a number of analytical results that underly the present case study; details can be found in [2, 14].

2.2.1. Zeroth-Order Propagator

The zeroth-order propagator for the two-stage gene expression model (Fig. 1A) represents an approximation to the CME, Eq. (1), under the assumption of infinitely fast mRNA degradation. Mathematically speaking, it is obtained in the singular limit of $\gamma \to \infty$, *i.e.*, of $\varepsilon \to 0$. Following [2], we have

$$P_{n|n_0}(\tau,0) = (1-\mathrm{e}^{-\tau})^{n_0}\Big(\frac{1+b\mathrm{e}^{-\tau}}{1+b}\Big)^{a}\Big(\frac{b}{1+b}\Big)^{n}\sum_{k=0}^{n}\Big\{\frac{(-1)^k}{k!(n-k)!}\frac{\Gamma(a+n-k)\Gamma(n_0+1)}{\Gamma(a)\Gamma(n_0-k+1)} \times \Big[\frac{1+b}{b(1-\mathrm{e}^{\tau})}\Big]^{k}{}_2F_1\Big(-n+k,-a,1-a-n+k,\frac{1+b}{\mathrm{e}^{\tau}+b}\Big)\Big\} \tag{2}$$

for the zeroth-order marginal probability $P_{n|n_0}(\tau,0)$ of observing n protein molecules after time τ, given $m_0 = 0$ molecules of mRNA and n_0 molecules of protein initially. Here, ${}_2F_1(a,b,c,z)$ is the Gauss hypergeometric function [20]. We remark that, by construction, $P_{n|n_0}(\tau,0)$ neglects any contributions from the fast t-scale, as the decay of mRNA is instantaneous to leading order in ε.

2.2.2. Uniform (First-Order) Propagator

The uniform propagator, denoted $\mathcal{P}_{n|n_0}(\tau,t,\varepsilon)$, was derived as in [14]. Here, ε denotes the perturbation parameter, as before, while t is the fast time variable. We emphasize that $\mathcal{P}_{n|n_0}$ describes the probability of transitioning from n_0 protein molecules initially to n molecules at time $\tau = \varepsilon t$, uniformly on the two time-scales. After some algebraic rearrangement, we find

$$\mathcal{P}_{n|n_0}(\tau,t,\varepsilon) = P_{n|n_0}(\tau,\varepsilon) + \varepsilon a\Big(\frac{b}{1+b}\Big)^{n-n_0}\frac{1}{(1+b)^2}\times[n-n_0-b-(1+b)t] + \frac{\varepsilon a}{\Gamma(n-n_0+2)}(bt)^{n-n_0}t \times\Big[{}_1F_1(n-n_0+1,n-n_0+2,-(1+b)t)t\Big(1-\frac{n-n_0-b}{1+b}\Big)+\frac{n-n_0+1}{1+b}\mathrm{e}^{-(1+b)t}\Big] \tag{3}$$

to first order in ε; here, ${}_1F_1(a,b,z)$ is the Kummer function of the first kind (or confluent hypergeometric function) [20]. We remark that the transition probability $P_{n|n_0}(\tau,\varepsilon)$ contributes on the slow τ-scale in Eq. (3), while the t-dependent contribution in Eq. (3) accounts for the transient dynamics on the fast time-scale.

Specifically, $P_{n|n_0}(\tau,\varepsilon)$ denotes the marginal probability, up to and including $\mathcal{O}(\varepsilon)$-terms, of observing n protein molecules after time τ given $m_0 = 0$ molecules of mRNA and n_0 molecules of protein initially:

$$P_{n|n_0}(\tau,\varepsilon) = \sum_{m=0}^{\infty} P_{m,n|0,n_0}(\tau,\varepsilon) \tag{4}$$

As shown in [14], the probability of encountering more than 1 molecule of mRNA at time τ is negligible to first order in ε; thus, Eq. (4) reduces to

$$P_{n|n_0}(\tau,\varepsilon) = P_{0,n|0,n_0}(\tau,\varepsilon) + P_{1,n|0,n_0}(\tau,\varepsilon). \tag{5}$$

After some algebraic simplification, the two transition probabilities $P_{0,n|0,n_0}$ and $P_{1,n|0,n_0}$ in the above relation are found to be

$$\begin{aligned}
P_{0,n|0,n_0}(\tau,\varepsilon) &= (1-\mathrm{e}^{-\tau})^{n_0}\Big(\frac{b}{1+b}\Big)^n\Big(\frac{1+b\mathrm{e}^{-\tau}}{1+b}\Big)^a \\
&\quad\times\sum_{k=0}^{n}\frac{1}{(n-k)B(a,n-k)}{}_2F_1\big(-n+k,-a,1-a-n+k,\tfrac{1+b}{\mathrm{e}^{\tau}+b}\big) \\
&\quad\times\Big\{g(n_0,k)-\frac{\varepsilon}{2}\frac{a}{(1+b)^2}(k+1)\times\Big[{}_2F_1\big(-k,-n_0,-1-k,\tfrac{1+b}{b(1-\mathrm{e}^{\tau})}\big) \\
&\quad+\Big(\frac{1+b}{\mathrm{e}^{\tau}+b}\Big)^{k+2}\mathrm{e}^{2\tau}{}_2F_1\big(-k,-n_0,-1-k,\tfrac{\mathrm{e}^{\tau}+b}{b(1-\mathrm{e}^{\tau})}\big)\Big]\Big\},\quad\text{with} \\
g(n_0,k) &= \begin{cases}0 & \text{for } k>n_0\\ (-1)^k\binom{n_0}{k}\big[\frac{(1+b)}{b(1-\mathrm{e}^{\tau})}\big]^k & \text{for } k\le n_0;\end{cases}
\end{aligned}\tag{6}$$

$$\begin{aligned}
P_{1,n|0,n_0}(\tau,\varepsilon) &= a\varepsilon\Big(\frac{b}{b+1}\Big)^n\frac{1}{b+1}(1-\mathrm{e}^{-\tau})^{n_0}\Big(\frac{1+b\mathrm{e}^{-\tau}}{1+b}\Big)^a \\
&\quad\times\sum_{k=0}^{n}\Big\{\frac{1}{(n-k)B(a,n-k)}{}_2F_1\big(k-n,-a,-a+k-n+1,\tfrac{b+1}{\mathrm{e}^{\tau}+b}\big) \\
&\quad\times\Big[h(n_0,k)+(-1)^{n_0}\Big[\frac{b\mathrm{e}^{\tau}+1}{b(1-\mathrm{e}^{\tau})}\Big]^{n_0}\Big]\Big\},\quad\text{with} \\
h(n_0,k) &= \begin{cases}0 & \text{for } k\ge n_0\\ \binom{n_0}{k+1}\big[\frac{b+1}{b(1-\mathrm{e}^{\tau})}\big]^{k+1}{}_2F_1\big(1,k-n_0+1,k+2,\tfrac{b+1}{b(1-\mathrm{e}^{\tau})}\big) & \text{for } k<n_0.\end{cases}
\end{aligned}\tag{7}$$

Here, $B(a,b):=\frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ is the Beta function, with the proviso that $\frac{1}{(n-k)B(a,n-k)}=1$ when $n=k$.

Finally, Eq. (3) can be simplified by substituting

$$\begin{aligned}
{}_1F_1(n-n_0+1;n-n_0+2;-(1+b)t) &= [(1+b)t]^{-(n-n_0+1)}\Gamma(n-n_0+2) \\
&\quad-(n-n_0+1)\Gamma\big(n-n_0+1,(1+b)t\big)
\end{aligned}\tag{8}$$

to achieve the computationally more tractable formulation

$$\begin{aligned}
\mathcal{P}_{n|n_0}(\tau,t,\varepsilon) &= P_{n|n_0}(\tau,\varepsilon) \\
&\quad+\varepsilon a\Big(\frac{b}{1+b}\Big)^{n-n_0}\frac{1}{(1+b)^2}[n-n_0-b-(1+b)t]+\varepsilon at\Big\{-\Big(\frac{b}{1+b}\Big)^{n-n_0}\frac{1}{(1+b)t} \\
&\quad\times\Big(\frac{b+n_0-n}{1+b}-t\Big)\big[1-Q(n-n_0+1,(1+b)t)\big]+\frac{(bt)^{(n-n_0)}}{1+b}\frac{\mathrm{e}^{-(1+b)t}}{\Gamma(n-n_0+1)}\Big\}.
\end{aligned}\tag{9}$$

Here, $Q(a,x) := \frac{\Gamma(a,x)}{\Gamma(a)}$ denotes the regularized upper incomplete gamma function.

2.3. Special Cases of the Hypergeometric Functions

Care must be taken when evaluating the hypergeometric function ${}_2F_1(a,b,c,z)$. The following special cases are of use [20].

- $a = -k = c$ ($k \in \mathbb{Z}^+$):

$$\begin{aligned} {}_2F_1(a,b,c,z) &= {}_2F_1(-k,b,-k,z) \\ &= \sum_{n=0}^{m} (b)_n \frac{z^n}{n!}, \end{aligned} \tag{10}$$

  where $(x)_n = x(x+1)\dots(x+n-1)$ is the rising factorial of x.

- $a = -k$, $c = -k-1$ ($k \in \mathbb{Z}^+$):

$$\begin{aligned} {}_2F_1(a,b,c,z) &= {}_2F_1(-k,b,-k-1,z) \\ &= \sum_{n=0}^{\min(-a,-b)} (b)_n \frac{z^n}{n!} \frac{a+n-1}{a-1}. \end{aligned} \tag{11}$$

- $a > 0$, $c > 0$, $b = -k$ ($k \in \mathbb{Z}^+$):

$$\begin{aligned} {}_2F_1(a,b,c,z) &= {}_2F_1(a,-k,c,z) \\ &= \sum_{n=0}^{k} \frac{(a)_n (b)_n}{(c)_n} \frac{z^n}{n!}. \end{aligned} \tag{12}$$

2.4. Stochastic Simulation

Stochastic simulations were performed using the StochKit 2.0 [21] simulation framework and the standard stochastic simulation algorithm [3], with a non-dimensionalized transcription rate $a = 20$ and a non-dimensionalized translation rate $b = 2.5$, corresponding to "regime I" as defined in [14]. We considered mRNA degradation rates of $\gamma \in \{10, 20, 50, 100\}$ and a protein degradation rate of 1. Each value of γ was simulated 20 times, and the resulting trajectories were used for computing the probability landscapes of the rescaled model parameters a and b. Protein quantities were observed without measurement noise at intervals of 0.1 time units. All simulation runs assumed zero molecules of mRNA and protein initially, *i.e.*, $m_0 = 0 = n_0$.

2.5. Implementation

Both the zeroth-order propagator $P_{n|n_0}$, Eq. (2), and the uniform propagator $\mathcal{P}_{n|n_0}$, Eq. (3), were implemented in C++ with a Matlab mex-file interface. Special functions were evaluated using the GNU scientific library [22], the `Hyp_2F1` function implementation of the Gauss hypergeometric function [23], and the `Algorithm 910` multiprecision special function library [24]. It proved indispensable to use a high precision numerical library due to several computations involving subtraction of very large numbers. While the difference of such numbers is potentially below a double precision machine error of approximately 10^{-13}, they are nonetheless essential in the correct computation of the transition probabilities. However, our C++ implementation is still inaccurate in some extreme cases, typically for very large protein numbers n, due to numerical differences which are sometimes as small as 10^{-370} in Eq. (7), but which unfortunately cannot be neglected as they are inflated by the remaining terms in the expression. Such inaccuracies are infrequent, though, and generally occur during transitions for which the uniform propagator yields nonphysical values; thus, they do not substantially affect our analysis, or the conclusions obtained in this study.

The finite state projection algorithm was implemented in Matlab, assuming no more than 2 mRNA molecules and no more than 200 protein molecules, in agreement with simulation.

3. RESULTS AND DISCUSSION

To assess the applicability of the zeroth-order propagator $P_{n|n_0}(\tau, 0)$, Eq. (2), and the uniform propagator $\mathcal{P}_{n|n_0}(\tau, t, \varepsilon)$, Eq. (3), for parameter inference in the two-stage gene expression model, we simulated time-series with a specific parameter pair (a^*, b^*). Then, we computed the likelihood of the observed data set on the basis of the two propagators for a range of values for the parameters a and b. For simplicity, we assumed the scale separation γ between mRNA and protein lifetimes to be known (see Methods for definitions).

3.1. Protein Time-Courses Simulated With Gillespie's Algorithm

We simulated mRNA and protein time-series for the two-stage gene expression model (Fig. 1A) using Gillespie's algorithm [3] (see Methods for details). Simulations were initialized with $m_0 = 0$

mRNA molecules and $n_0 = 0$ protein molecules, although the propagator-based approach is equally applicable to any initial number of proteins, as shown in [14].

The generated protein time-courses were sampled at $N = 101$ points in time, with fixed increments of $\Delta t = 0.1$ to mimic the measurement of protein abundance with time-lapse microscopy, see Fig. 1B. For each transition in the observed time-series, we computed the approximate probability on the basis of the analytical propagators $P_{n|n_0}$ and $\mathcal{P}_{n|n_0}$; cf. Fig. 1B, inset. Notably, we ignored measurement noise throughout, *i.e.*, we only investigated the suitability of the two propagators for synthetic "ideal" data (see Discussion for possible extensions).

We note, moreover, that the expressions in Eqs. (2) and (3) can be used to visualize the likelihood of various sample paths in the underlying stochastic networks for a given set of parameters and conditional on the initial condition; see Fig. 1C.

3.2. Parameter Inference

Given the propagators $P_{n|n_0}$ and $\mathcal{P}_{n|n_0}$, we computed the log-likelihood $L(a, b)$ of the simulated trajectories for a range of parameter values (a, b) in the subspace $(a, b) \in [10^{-1}, 10^3] \times [10^{-3}, 10^3]$. Here, the log-likelihood is defined as

$$L(a, b) = \sum_{i=1}^{N} \log P^*_{n_i|n_{i-1}}, \qquad (13)$$

where either $P^*_{n_i|n_{i-1}} = P_{n_i|n_{i-1}}$ as in Eq. (2) for the zeroth-order propagator or $P^*_{n_i|n_{i-1}} = \mathcal{P}_{n_i|n_{i-1}}$ as in Eq. (9) for the uniform propagator. We note that both propagators depend on the parameters a and b; moreover, the parameter $\varepsilon = \gamma^{-1}$ is assumed to be known. The term n_i represents the number of proteins at measurement time t_i. Thus, we compute the logarithm of the probability of each transition, from n_{i-1} protein molecules at time t_{i-1} to n_i molecules at time t_i, in the sequence of observed measurements (see Fig. 1B, inset).

In order to estimate the parameters a and b from simulated protein time-courses, Eq. (13) has to be evaluated very frequently. We thus developed a numerically stable expression for the uniform propagator $\mathcal{P}_{n|n_0}$ (see Section 2 2.2), and we used an efficient implementation in C++ for both propagators that results in reasonable runtimes; see Section 2 2.5 for details.

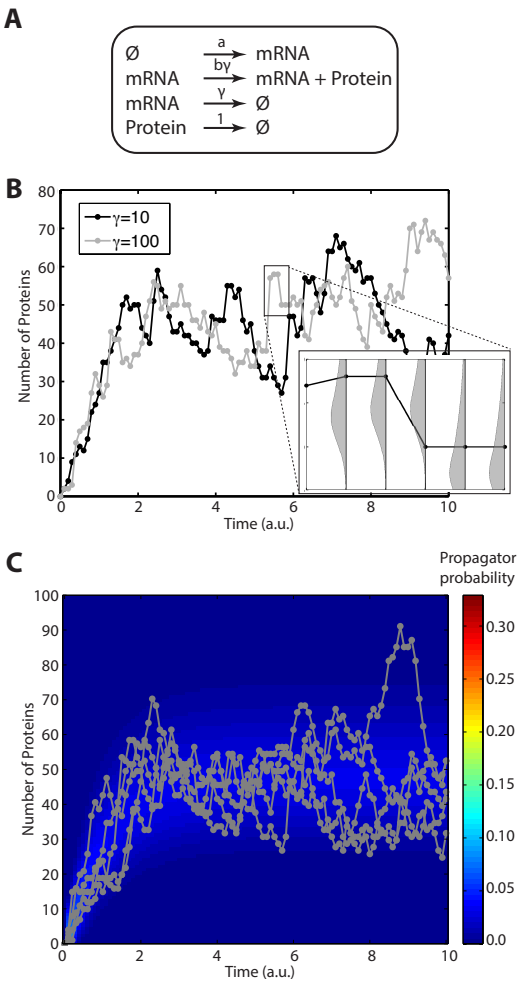

FIG. 1: A. The two-stage model for gene expression, which captures stochastic birth and death of mRNA and protein, with non-dimensionalized parameters a for transcription, b for translation, and γ for mRNA degradation. B. Time-courses were simulated using the stochastic simulation algorithm, shown here for $a = 20$, $b = 2.5$, and $\gamma = 10$ or $\gamma = 100$. Probabilities can be computed for each protein transition from the analytical two-stage propagators given in Eqs. (2) and (9) (inset, probability distributions shown in gray). C. Analytical propagators can be used to compute the probability of observing a particular number of protein molecules at arbitrary points in time, conditional on the initial conditions. The prediction from the uniform propagator, Eq. (3) (blue background), shows good qualitative agreement with stochastic simulation (gray lines), as illustrated for $\gamma = 10$ here.

3.3. Comparison of Propagator Accuracy and Efficiency

We scanned the space of parameter values (a, b) on a logarithmically spaced 44×45-grid with

$10^{-1} \leq a \leq 10^3$ and $10^{-3} \leq b \leq 10^3$. For each pair (a, b), we computed the log-likelihood $L(a, b)$, thus obtaining a likelihood landscape that should ideally have its maximum, the maximum likelihood estimator (MLE), at the true parameter values (a^*, b^*). We immediately encountered the obstacle that the uniform propagator $\mathcal{P}_{n|n_0}$ yields negative transition probabilities, or even probabilities larger than one, for some choices of (a, b). The resulting non-physicality is discussed in [14], and is due to the fact that $\mathcal{P}_{n|n_0}$ is derived from an asymptotic approximation; nonetheless, it is problematic when computing the overall log-likelihood, as the definition in Eq. (13) becomes meaningless. Thus we introduce an "averaged log-likelihood", $\bar{L}(a, b)$, which removes all non-physical values (*i.e.*, those that are larger than one or less than or equal to zero):

$$\bar{L}(a,b) = \frac{\sum_{i=1}^{N} \mathbb{1}\{0 < P^*_{n_i|n_{i-1}} \leq 1\} \left[\log P^*_{n_i|n_{i-1}}\right]}{\sum_{i=1}^{N} \mathbb{1}\{0 < P^*_{n_i|n_{i-1}} \leq 1\}}, \tag{14}$$

with $P^*_{n_i|n_{i-1}}$ defined as in (13). The averaged log-likelihood represents the average log-likelihood for a set of parameters (a, b), after removal of all non-physical transition densities. The averaging compensates for the fact that the number of non-physical transitions may vary greatly for different values of (a, b). Since each retained transition only decreases the overall log-likelihood of the time-series, the log-likelihood estimate without normalization would inherently be biased towards regions of (a, b)-space for which many transitions were omitted.

Using (14), we compute the log-likelihood landscapes (shown as contour plots) for the zeroth-order and uniform propagators, obtained from a single time-course simulated with $\gamma = 100$, observed at $N = 101$ points in time at intervals of $\Delta t = 0.1$. Computing the MLE, we find that it deviates from the true parameter values $(a^*, b^*) = (20, 2.5)$ in (a, b)-space, both for the zeroth-order propagator $P_{n|n_0}$ (Fig. 2A) and for the uniform propagator $\mathcal{P}_{n|n_0}$ (Fig. 2B). For comparison, we also generated a finite state projection approximation (FSP) to the log-likelihood landscape (Fig. 2C), which was computed by solving the CME (1), assuming that mRNA has at most two copies (in agreement with simulation), and that the number of proteins does not exceed 200; see [7] for details on the FSP.

The log-likelihood landscape generated using each approach shows some bias in the MLE when using only a single trajectory (Figs. 2A-C). However, for all three approaches, the MLE converges to the true model parameters (a^*, b^*) as the number of simulation runs used increases from one to 20; see Figs. 2D-F, wherein we depict the sum of the averaged log-likelihoods over each of the trajectories. This convergence suggests that the bias is largely due to the inherent stochasticity of the system, which is averaged out as more data are incorporated. Thus, we conclude that for $\gamma = 100$, both analytical propagators provide a good approximation to the underlying transition density, and may hence be of use for parameter inference. However, the FSP yields a log-likelihood landscape that is more tightly peaked around (a^*, b^*), as is seen from a comparison of contour lines in Figs. 2D-F; the propagator-based approaches are hence less able to distinguish between combinations of a and b which lead to approximately equal dynamics in the observed time-series.

The approximation provided by the propagators $P_{n|n_0}$ and $\mathcal{P}_{n|n_0}$ deteriorates as γ decreases, *i.e.*, if the perturbation parameter $\varepsilon = \gamma^{-1}$ is not sufficiently small. Thus, in the case of $\gamma = 10$, the uniform propagator generates many non-physical transition probabilities which heavily distort the log-likelihood landscape, see Fig. 3A. These distortions lead to a severe bias of the MLE with respect to the true model parameters (a^*, b^*).

To understand the origins of this bias, it is helpful to examine a representative time-series. In Fig. 3B, a typical protein time-course with $\gamma = 10$ is shown (top), along with the log-likelihood (bottom) obtained from the uniform propagator $\mathcal{P}_{n|n_0}$, Eq. (3), for the true parameter values (black), and for the MLE (cyan). Transitions for which $\mathcal{P}_{n|n_0}$ yields non-physical values are shown as white squares within the colored bars at the bottom of Fig. 3B. We indicate one such transition with arrows in Fig. 3B, and compute the corresponding transition probability distribution using the uniform propagator, Fig. 3C. In this example, the protein time-course transitions from 55 to 57 molecules within one time interval. Examining the propagator evaluated for the true model parameters (a^*, b^*) with initially 55 protein molecules, *i.e.*, calculating $\mathcal{P}_{57|55}$, we see that the propagator becomes negative for $57 \leq n \leq 60$ (Fig. 3C, arrow). We note that the corresponding negative values are of order $\mathcal{O}(\gamma^{-2})$, and thus within the error incurred by the expansion in Eq. (3), which is accurate to $\mathcal{O}(\gamma^{-1})$.

Using the uniform propagator $\mathcal{P}_{n|n_0}$, we computed a portion of the "transition matrix", *i.e.*, the probability of all transitions from $n(t) \in \{0, \ldots, 100\}$ to $n(t + \Delta t) \in \{0, \ldots, 100\}$, evaluated at $(a^* b^*)$, see Fig. 3D. From that plot, it is obvious that large regions of the transition space yield non-physical values, shown in gray. Similar distortions were also found for $\gamma \in \{20, 50\}$.

To quantify the frequency of these non-physical

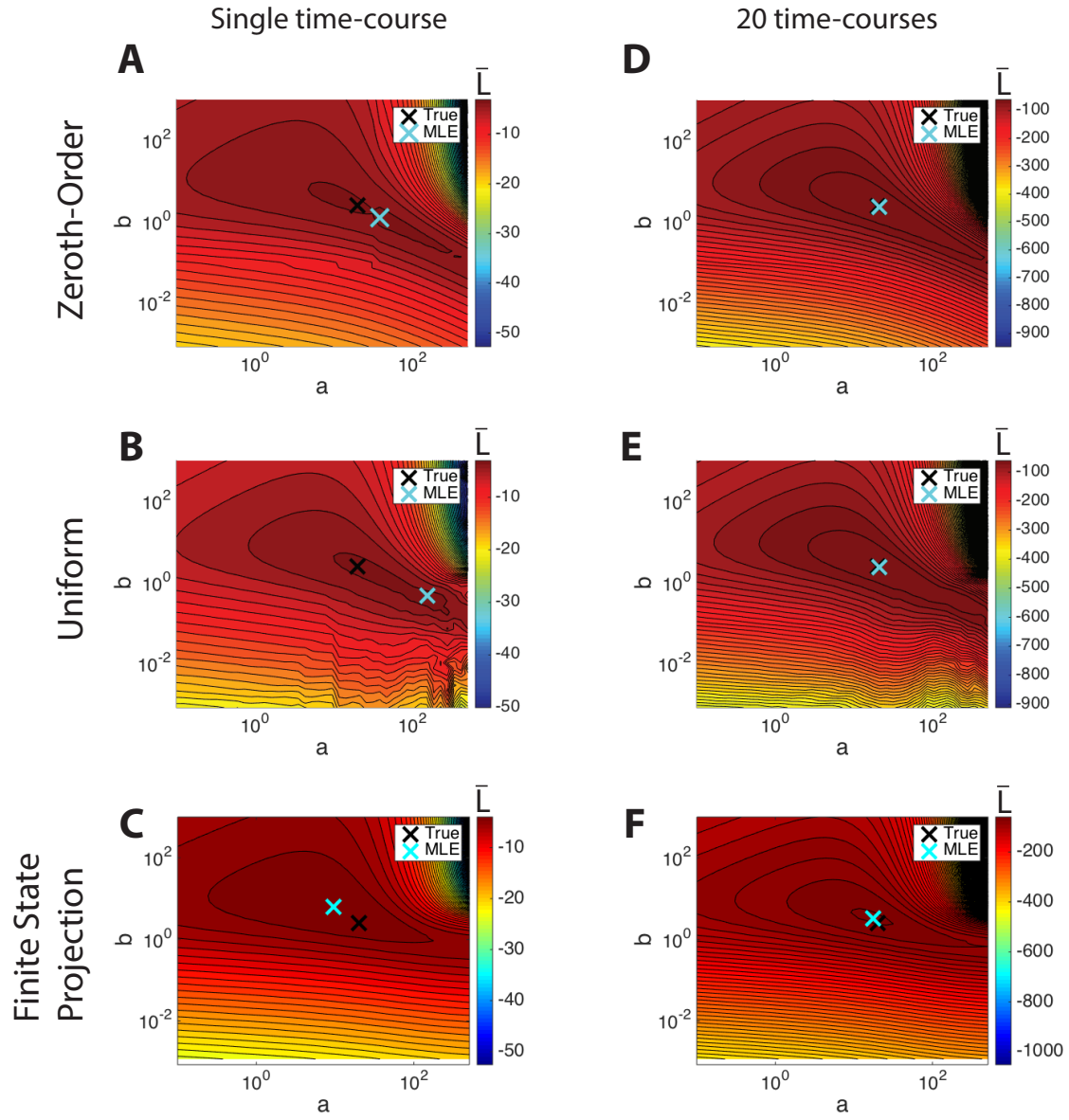

FIG. 2: Averaged simulated log-likelihood landscapes for $(a, b) = (20, 2.5)$ and $\gamma = 100$. Landscapes for single time-courses (left) are shown with contour lines drawn at intervals of 1 unit; contours for landscapes obtained from 20 time-courses (right) are drawn at intervals of 10 units. The averaged log-likelihood landscapes generated using a single time-course for the zeroth-order propagator, Eq. (2), (A), the uniform propagator, Eq. (3), (B), and a finite state projection approximation (C) for a single time-course each display a bias of the MLE with respect to the true model parameters (a^*, b^*). Notably, the landscape of the uniform propagator (B) shows distortions arising from non-physical transition probabilities for some parameter pairs (a, b). As the number of trajectories is increased to 20, the MLE converges to (a^*, b^*) for each of the zeroth-order propagator (D), uniform propagator (E), and the finite state projection (F). The averaged log-likelihood resulting from the finite state projection seems to be most tightly-peaked around the true parameter values (a^*, b^*).

transitions, we calculated a "computability score"

$$C(a,b) = \frac{1}{N \cdot N_{\text{traj}}} \times \sum_{k=1}^{N_{\text{traj}}} \sum_{i=1}^{N} \mathbb{1}\{0 < P_{n_i^k | n_{i-1}^k} \leq 1\}, \quad (15)$$

where the superscript k indicates the index of the simulated trajectory. Thus, $C(a, b)$ captures the fraction of evaluated transitions for a given pair (a, b) which were physically admissible (between zero and one) for the uniform propagator $\mathcal{P}_{n|n_0}$. A

plot of the computability score reveals that certain regions of the parameter space suffer from low computability, *i.e.*, that they yield many non-physical values, which are apparent as dark regions, see Fig. 3E. By contrast, $\mathcal{P}_{n|n_0}$ provides a better approximation to the true transition probability when evaluated in the so-called "regime II" defined in [14], with $(a, b) = (0.5, 100)$, which corresponds to continuous protein synthesis. Correspondingly, examining the transition matrix, Fig. 3F, we found that all transitions were computable and physically admissible, as opposed to the transition matrix obtained in regime I, for $(a, b) = (20, 2.5)$.

Thus, we conclude that the uniform propagator may provide a useful approximation to the stochastic propagator in certain regions of parameter space, in particular for low values of a and high ones of b, such as in regime II. However, it breaks down in large regions of parameter space for which the computability is low. In such regions, the remaining transitions may in fact have a higher likelihood than the true model parameters (a^*, b^*) (see Fig. 3B), which can lead to a biased estimate of the model parameters, as in Fig. 3A.

4. CONCLUSION

In this work, we have investigated the utility of a propagator-based approach for approximating the transition probabilities in a simple two-stage gene expression model by attempting parameter inference from protein time-series. The latter can be derived, *e.g.*, from time-lapse microscopy of fluorescently-labeled proteins in single cells, and are thus of interest for the study of regulation in gene expression. Here, we only used simulated time-series measured at regular intervals, without measurement noise. The simulations were initialized with zero molecules of both mRNA and protein; this simplification, as compared to a typical biological setting, does not affect the subsequent analysis.

We compared a newly developed uniform propagator, Eq. (3), which was derived in [14] by application of geometric singular perturbation techniques, to a previously proposed propagator, Eq. (2) [2], which corresponds to the singular limit as the perturbation parameter in the model is decreased to zero. The comparison was performed on the basis of the probability landscapes of the two relevant model parameters a and b, which represent rescaled transcription and translation rates, respectively. For reference, the two propagators were also compared against another approximate solution of the CME, corresponding to the finite state projection (FSP). The FSP is a numerical method, and is *a priori* restricted to a subspace of the possible configurations of the system; nonetheless, it shows very good identifiability of the model parameters given sufficiently many observed trajectories (see Fig. 2F).

The results of our investigation indicate that both propagators perform well when the value of γ — the non-dimensionalized mRNA degradation rate — is sufficiently large. In the case of $\gamma = 100$, both capture the true model parameters almost exactly, as long as there are sufficiently many time-courses. In our simulations, 20 time-courses — about 2000 observed transitions — were needed before convergence to the true parameter values, a number which is attainable in a real biological experiment. However, for smaller values of γ, that is, assuming a decrease in scale separation between mRNA and protein degradation, the uniform propagator becomes inconsistent, in that it generates negative transition probabilities for many segments of the protein time-course. This loss of positivity is a general feature of asymptotic expansions for probability distributions, which *a priori* only satisfy the non-negativity required of the distributions provided the corresponding perturbation parameter is sufficiently small. While the occurrence of negative probabilities for transient times, *i.e.*, on the fast time-scale, is irrelevant for the evaluation of the steady state of the system, it is of extreme relevance to the utility of the propagator for parameter inference. Although the zeroth-order propagator is thus inherently less accurate in an asymptotic sense, it may somewhat counter-intuitively still prove more useful for parameter inference, as it does not yield negative transition densities under any circumstances.

Since the majority of time-courses contained transitions for which the calculated probabilities were negative, it was necessary to devise a better measure which utilized as much information as possible. We thus discarded all negative transitions, and used the remaining non-negative transitions, normalized by their numbers in each time-course, to obtain an averaged likelihood for each pair (a, b) in the parameter space. While this approach retains the maximum information possible from the trajectories, it nonetheless seemingly introduces distortions into the probability landscapes of the parameter space (see Fig. 3A). These distortions proved sufficient to shift the MLE away from the true parameter values (a^*, b^*), thus limiting the utility of the uniform propagator for inference in regime I.

In the current analysis, we have restricted ourselves to computing the log-likelihood landscape, *i.e.*, the approximate averaged log-likelihood $\bar{L}(a, b)$, for all parameter pairs (a, b) on a discrete grid that was sampled uniformly in log-space (see Methods). This approach is useful for visualiz-

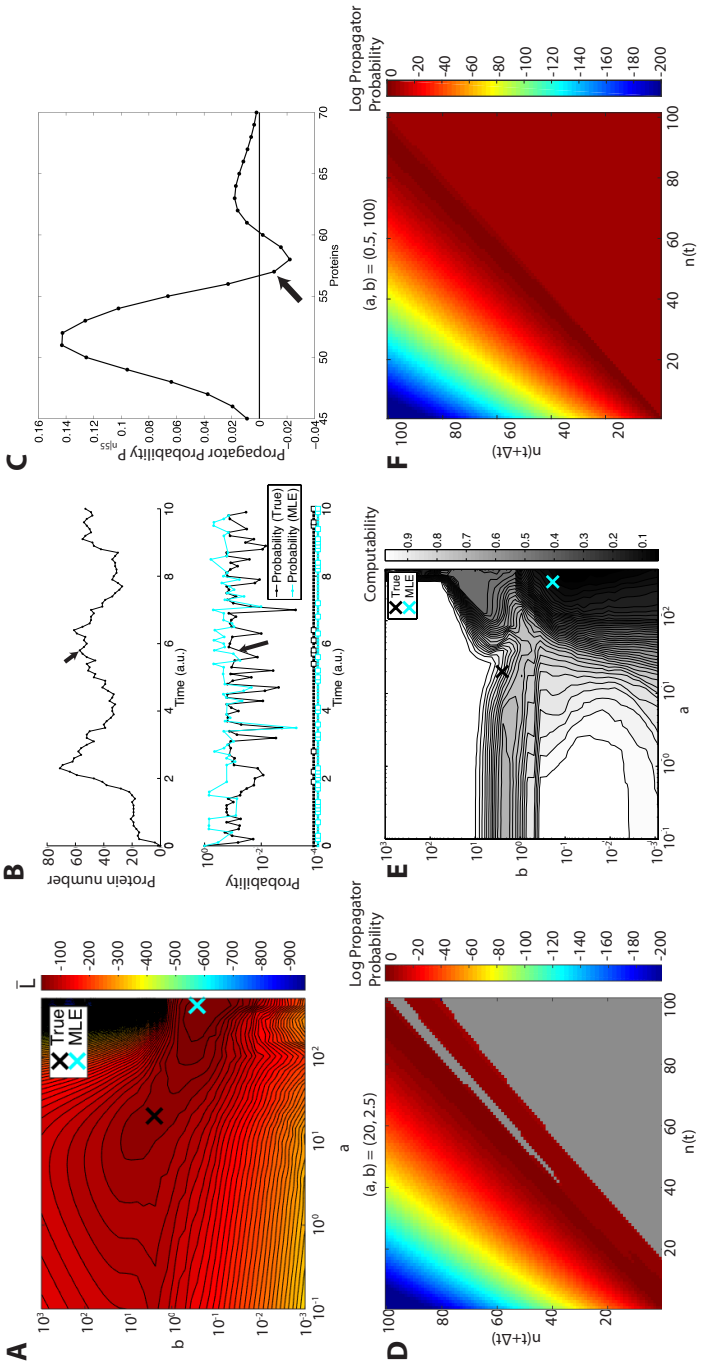

FIG. 3: A. The averaged log-likelihood landscape of (a, b) for $\gamma = 10$ of the uniform propagator $\mathcal{P}_{n|n_0}$ shows prominent distortions in the contours caused by frequent non-computable transitions. The MLE (cyan) exhibits an obvious bias with respect to the true model parameters (a^*, b^*) (black). B. Inspection of a single time-course (shown on top), evaluated at (a^*, b^*) and at the MLE (bottom), reveals more non-computable transitions (indicated with white boxes below) for the MLE than for the true parameters; however, for those points that can be computed, the MLE probability is higher than for the true parameters, leading to a higher averaged probability and thus to a biased estimate of the parameters (a, b). C. Transition probability in regime I, with $(a, b) = (20, 2.5)$. The transition marked with arrows in (B), from 55 to 57 molecules, results in a negative transition probability. D. The transition matrix for the uniform propagator in regime I from $n(t)$ to $n(t+\Delta t)$ proteins reveals a large region of non-computable transitions, shown in gray. E. The computability score $C(a, b)$ shows that the MLE is biased towards the region with the lowest computability, for which most transitions are omitted from the averaged log-likelihood score $\bar{L}(a, b)$. F. By contrast, the transition matrix is fully computable in regime II, with $(a, b) = (0.5, 100)$, corresponding to the region of bursty protein synthesis, *i.e.*, to translational bursting.

ing the probability landscape, but is not ideal for parameter inference. In a more realistic setting, one would compute the maximum likelihood estimator via numerical optimization, *e.g.*, by applying a finite-differencing scheme in conjunction with a gradient descent algorithm; see, *e.g.*, [25]. Alternatively, one could use Markov Chain Monte Carlo (MCMC) techniques to sample directly from the posterior in order to obtain the log-likelihood landscape [26]. The MCMC approach is particularly advantageous when the scale separation parameter γ is not known *a priori*, as was assumed in the current analysis, since the number of parameter combinations increases exponentially with the number of unknown parameters.

Thus far, we have not considered the effects of measurement noise. In order to obtain the correct parameter likelihoods in the presence of noisy measurements, one would have to marginalize over all possible paths, weighted by the probability of observing the measured values at each point along the sampled path, according to an error model such as normal or log-normal measurement noise. The variance of the noise then constitutes an additional unknown parameter σ which would have to be inferred. Integrating over all possible sample paths is of course computationally intractable due to the enormity of the number of such paths, even if some truncation of the possible path space is made, *e.g.*, by neglecting paths for which the probability of observing the measured data points lies below some arbitrarily small threshold. Alternatively, rather than integrating over all possible paths to obtain the true marginal parameter likelihoods, one could apply a variant of the expectation maximization algorithm [27] in which case the most likely parameter set (a, b, γ, σ) is inferred along with the "true" latent paths for mRNA and protein, respectively. A similar approach was employed by Suter, *et al.* [28], wherein the zeroth-order approximation presented in [2] is used along with simplifying assumptions in order to perform parameter inference from protein time-series.

To improve the utility of the uniform propagator for parameter inference, it is necessary to eliminate the non-physical transition probabilities, which can possibly be achieved via the inclusion of higher-order terms in the perturbation parameter ε in the corresponding asymptotic expansion, as the current approximation in Eq. (3) is accurate only up to and including first order terms in ε. Alternatively, the "fast" and "slow" propagators that were derived separately in [14], at first order in ε, could be "patched" at some suitable point in time so that positivity is ensured throughout. Further improvement is likely possible for specific parameter regimes (a, b, γ) in which the relative orders of magnitude of the three parameters naturally suggest a γ-dependent rescaling of a or b. Another possible application of the uniform propagator would be to combine it with other techniques, such as moment equations, in order to perform approximate parameter inference by attempting to match simultaneously the predicted steady-state distributions and autocorrelation functions of the model to empirical observations. The uniform propagator provides a more accurate approximation of the steady-state distribution in the two-stage model for gene expression, as is shown in [14], and is thus potentially well suited to such an approximate inference scheme.

Acknowledgments

The authors thank Peter Swain for stimulating discussions. Grant support is acknowledged from the Moray Endowment Fund, as well as from the Engineering and Physical Sciences Research Council through MAXIMATHS, an initiative by the School of Mathematics at the University of Edinburgh aimed at maximizing the impact of mathematics in science and engineering. We also acknowledge the European Research Council for generous funding support.

References

[1] M. Thattai and A. van Oudenaarden, Proceedings of the National Academy of Sciences **98**, 8614 (2001).

[2] V. Shahrezaei and P. S. Swain, Proceedings of the National Academy of Sciences **105**, 17256 (2008).

[3] D. T. Gillespie, The Journal of Physical Chemistry **81** (1977).

[4] N. G. Van Kampen, *Stochastic Processes in Physics and Chemistry* (Elsevier, Amsterdam, 2011).

[5] S. Engblom, Applied Mathematics and Computation **180**, 498 (2006).

[6] J. Hasenauer, V. Wolf, and A. Kazeroonian, Journal of Mathematical Biology **69** (2013).

[7] B. Munsky and M. Khammash, The Journal of Chemical Physics **124**, 044104 (2006).

[8] D. J. Wilkinson, Nature Reviews Genetics **10**, 122 (2009).

[9] P. Bokes, J. R. King, A. T. A. Wood, and M. Loose, Journal of Mathematical Biology **64**, 829 (2011).
[10] P. Bokes, J. R. King, A. T. A. Wood, and M. Loose, Journal of Mathematical Biology **65**, 493 (2011).
[11] M. Kaern, T. C. Elston, W. J. Blake, and J. J. Collins, Nature Reviews Genetics **6**, 451 (2005).
[12] M. Thattai and A. van Oudenaarden, Proceedings of the National Academy of Sciences **98**, 8614 (2001).
[13] J. Paulsson, Physics of Life Reviews **2**, 157 (2005).
[14] N. Popović, C. Marr, and P. S. Swain, Journal of Mathematical Biology (in press) (2015).
[15] C. K. R. T. Jones, in *Dynamical systems (Montecatini Terme, 1994)* (Springer, Berlin, 1995), vol. 1609 of *Lecture Notes in Math.*, pp. 44–118.
[16] P. A. Lagerstrom, *Matched asymptotic expansions*, vol. 76 of *Applied Mathematical Sciences* (Springer-Verlag, New York, 1988), ISBN 0-387-96811-3, ideas and techniques.
[17] P. S. Swain, M. B. Elowitz, and E. D. Siggia, Proceedings of the National Academy of Sciences **99**, 12795 (2002).
[18] J. W. Young, J. C. W. Locke, A. Altinok, N. Rosenfeld, T. Bacarian, P. S. Swain, E. Mjolsness, and M. B. Elowitz, Nature Protocols **7**, 80 (2011).
[19] Y. Taniguchi, P. J. Choi, G.-W. Li, H. Chen, M. Babu, J. Hearn, A. Emili, and X. S. Xie, Science **329**, 533 (2010).
[20] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions* (National Bureau of Standards, Dover, 1972).
[21] K. R. Sanft, S. Wu, M. Roh, J. Fu, R. K. Lim, and L. R. Petzold, Bioinformatics **27**, 2457 (2011).
[22] *GNU Scientific Library Reference Manual* (2013).
[23] N. Michel and M. V. Stoitsov, Computer Physics Communications **178**, 535 (2008).
[24] C. Kormanyos, ACM Transactions on Mathematical Software **37**, 1 (2011).
[25] J. Snyman, *Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms* (Springer, New York, 2005).
[26] D. J. C. Mackay, *Information theory, inference and learning algorithms* (Cambridge University Press, Cambridge, 2003).
[27] A. P. Dempster, N. M. Laird, and D. B. Rubin, Journal of the Royal Statistical Society Series B (Methodological) **39** (1977).
[28] D. M. Suter, N. Molina, D. Gatfield, K. Schneider, U. Schibler, and F. Naef, Science **332**, 472 (2011).