Dimensionality reduction via path integration for computing mRNA distributions

Jaroslav Albert

Received: 1 July 2020 / Revised: 29 March 2021 / Accepted: 13 October 2021 / Published online: 3 November 2021
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

Abstract
Inherent stochasticity in gene expression leads to distributions of mRNA copy numbers in a population of identical cells. These distributions are determined primarily by the multitude of states of a gene promoter, each driving transcription at a different rate. In an era where single-cell mRNA copy number data are more and more available, there is an increasing need for fast computations of mRNA distributions. In this paper, we present a method for computing separate distributions for each species of mRNA molecules, i.e. mRNAs that have been either partially or fully processed post-transcription. The method involves the integration over all possible realizations of promoter states, which we cast into a set of linear ordinary differential equations of dimension $M \times n_j$, where $M$ is the number of available promoter states and $n_j$ is the mRNA copy number of species $j$ up to which one wishes to compute the probability distribution. This approach is superior to solving the Master equation (ME) directly in two ways: (a) the number of coupled differential equations in the ME approach is $M \times \Lambda_1 \times \Lambda_2 \times \cdots \times \Lambda_L$, where $\Lambda_j$ is the cutoff for the probability of the $j$th species of mRNA; and (b) the ME must be solved up to the cutoffs $\Lambda_j$, which must be selected a priori. In our approach, the equation for the probability to observe $n$ mRNAs of any species depends only on the the probability of observing $n-1$ mRNAs of that species, thus yielding a correct probability distribution up to an arbitrary $n$. To demonstrate the validity of our derivations, we compare our results with Gillespie simulations for ten randomly selected system parameters.

Keywords Gene regulatory networks · Promoter · Probability distributions · Master equation · Path integral · Single-cell RNA data

Mathematics Subject Classification 92-10 · 92C42

Jaroslav Albert
jaroslavalbert81@gmail.com

1 Ronin Institute, Montclair, New Jersey, USA
1 Introduction

In the last decade, single-cell RNA sequencing techniques have advanced to a point where mRNA distributions can be obtained for thousands of genes with high accuracy (Fiers et al. 2018). These type of data offer insights into the stochastic processes that govern gene regulatory networks. For this reason, computational techniques that can interpret these data are in high demand. One of the aspects of gene regulation that single-cell RNA data can shed light on is the promoter architectures for individual genes. Knowing the mRNA distribution associated with a gene, it is in principal possible to reverse-engineer the promoter architecture that gives rise to said distribution. One approach to achieving this goal is to compute the mRNA probability distributions (PD) for a large number of promoter architectures and select the one(s) that best fits the data. However, this requires fast methods of computing mRNA PDs.

The two most conventional methods of computing PDs for gene products (predominantly RNA and protein) are: solving the Master equation (ME) (Van Kampen 2007) and the Gillespie algorithm (GA) (Gillespie 1977). What makes these two methods attractive is that they are derived from first principles; in fact, the GA is derived from the ME, which makes them different sides of the same coin. In practice the ME is useful only when solvable analytically or when it can be numerically integrated. New analytic and numerical techniques for solving the ME are constantly being developed, either by means of improving stochastic simulation algorithms (Gibson and Bruck 2000; Gillespie 2001; Cao et al. 2004, 2005a, b), or by solving the ME exactly/approximately (Jahnke and Huisinga 2007; Albert and Rooman 2016; Albert 2019; Shahrezaei and Swain 2008; Pendar et al. 2013; Bokes et al. 2012a, b; Popović et al. 2016; Veerman et al. 2018), or by a mix of the former two (Burrage et al. 2004; Jahnke and Altintan 2010; Albert 2016b, a; Duso and Zechner 2018; Alfonsi et al. 2005; Kurasov et al. 2018). In this paper, we enlarge this list by one.

Our approach is to reduce the ME for the mRNA and the promoter to a separate ME for each mRNA species. This is accomplished thanks to a theorem we have proven in an earlier paper (Albert 2019), which allows one to write the generating function (an alternative representation of the ME) for the mRNA as a modified ME for the promoter. In this fashion, the individual probability distributions for any one of the species of mRNA can be computed separately.

2 Promoter dynamics

The system we wish to describe consists of a gene promoter that can be in different states. Figure 1a–c shows three possible promoter states and the processes that cause one state to transform into another. If we let $x_i$ be the state of the empty promoter site $i$, and $y_i^k$ be the state of the promoter site $i$ occupied by transcription factor (TF) $k$, then the reactions that change the state of the promoter can be written as

$$x_i \xrightarrow{\alpha_i^k} y_i^k \quad i = 1, \ldots, M$$
Fig. 1  Example of a promoter with five promoter sites. a Transcription factor 3 binds to and dissociates from the promoter site 1 at the rate $\alpha_3^1$ and $\beta_3^{3,5}$, respectively. b Transcription factor 5 binds to and dissociates from the promoter site 2 at the rate $\alpha_5^2$ and $\beta_5^{5,1}$, respectively. c Transcription factor 1 binds to and dissociates from the promoter site 3 at the rate $\alpha_1^3$ and $\beta_5^{5,1}$, respectively. d Transcription factor 5 binds to and dissociates from the promoter site 2 at the rate $\alpha_3^3$ and $\gamma_5^2$, respectively.

\[
y^k_i \xrightarrow{b^k_i} x_i \quad i = 1, \ldots, M; \quad k = 1, \ldots, N
\]

where $\alpha^k_i$ is the TF association rate and

\[
b^k_i = y^k_i x_{i-1} x_{i+1} + \sum_l \left( \beta_{k,l-1}^i y^l_{i-1} x_{i+1} + \beta_{k,l+1}^i y^l_{i+1} x_{i-1} \right) + \sum_{lp} \tilde{\beta}_{k,l}^{klp} y^l_{i-1} y^p_{i+1},
\]

where $\beta_{k,l-1}^i$ is the dissociation rate of the $k$th TF from the promoter site $i$ when site $i - 1$ is occupied by the $l$th TF; $\beta_{k,l+1}^i$ is the dissociation rate of the $k$th TF from the promoter site $i$ when site $i + 1$ is occupied by the $l$th TF; and $\tilde{\beta}_{k,l}^{klp}$ is the dissociation rate of the $k$th TF from the promoter site $i$ when site $i - 1$ is occupied by the $l$th TF and site $i + 1$ is occupied by the $p$th TF. The variables $x_i$ and $y^k_i$ can only take the values 0 and 1. When $x_i = 1$, the $i$th promoter site is empty; when $x_i = 0$, it is occupied by a TF (any TF). When $y^k_i = 1$, the $i$th promoter site is occupied by the $k$th TF; when $y^k_i = 0$, it is empty. A promoter state is determined by a unique combination of ones and zeros taken by the variables $x_i$ and $y^k_i$, according to the available promoter sites and the number of TFs trying to bind them. For example, for $M = 2$ and $N = 2$, a promoter state where TF 1 is bound to promoter site 2, the set of variables $(x_1, x_2, y^1_1, y^2_1, y^1_2, y^2_2)$ would have the values $(1, 0, 0, 1, 0, 0)$. The ME for
the processes in Eq. (1) is given by

\[
\dot{P} = \sum_i \alpha_i \left[ (x_i + 1) P(x_i + 1, y_i - 1) - x_i P \right] \\
+ \sum_i b_i \left[ (y_i + 1) P(x_i - 1, y_i + 1) - y_i P \right].
\] (3)

For convenience, we define a variable \( s \) that labels different promoter states. For example, we could label the state specified by \((1, 0, 0, 0, 0)\) as \( s = 1 \) and the state specified by \((1, 1, 0, 0, 0, 0)\) as \( s = 2 \). Then, the transition from \( s = 1 \) to \( s = 2 \) would correspond to a process in which the first TF dissociates from the second promoter site. To make a connection between \( s \) and the variables \( x_i \) and \( y_i \), we must define a set

\[
S = \left\{ [x_1, \ldots, x_N], [y_1^1, \ldots, y_N^1], \ldots [y_1^M, \ldots, y_N^M] \right\},
\] (4)

such that \( S^s \) represents \( S \) for particular values of the variables \( x_i \) and \( y_i \). For example, if \( s = 1 \), we might have

\[
S^1 = \{[1, \ldots, 0, \ldots, 1], [0, \ldots, 0], \ldots [0, \ldots, 1, \ldots, 0] \ldots [0, \ldots, 0] \},
\] (5)

which represents a state with the \( k \)th TF bound to the \( i \)th site. The square brackets inside \( S \) are imaginary, serving only as a visual aid; hence, \( S \) can be thought of as a vector. How we index the promoter states is of no consequence, only that every state has a unique index. In terms of \( S \), we can write \( x_i = S^s_i \) and \( y_i^k = S^s_{kM+i} \), where the subscript labels the element of \( S^s \). Defining the probability vector as

\[
P = \begin{bmatrix}
P(S^1) \\
P(S^2) \\
\vdots \\
\vdots \\
\end{bmatrix},
\] (6)

the ME (3) can be written in the desired form:

\[
\frac{dP(S^s)}{dt} = \sum_{s'} M_{ss'} P(S^{s'}),
\] (7)

where

\[
M_{ss'} = \sum_{ik} a^k_i C_{ss}^{ik}(1, -1) + \sum_{ik} b^k_i C_{ss}^{ik}(-1, 1) - \sum_{ik} \left( a^k_i S^s_i + b^k_i S^s_{kM+i} \right),
\] (8)
and the matrices $C^i_{ss'}(1, -1)$ and $C^i_{ss'}(-1, 1)$ are defined as

$$
C^i_{ss'}(1, -1) = \begin{cases} 1, & \text{if } S^s_i = S^s_i + 1, \ S^s_{kM+i} = S^s_{kM+i} - 1 \\ 0, & \text{otherwise} \end{cases}
$$

$$
C^i_{ss'}(-1, 1) = \begin{cases} 1, & \text{if } S^s_i = S^s_i - 1, \ S^s_{kM+i} = S^s_{kM+i} + 1 \\ 0, & \text{otherwise} \end{cases}
$$

Converting $x_i$ and $y^k_i$ into the new variables $S^s$ in the dissociation rate, Eq. (2),

$$
b^k_i = \gamma^k_i S^s_i S^s_{i+1} + \sum_l (\beta^k_{li} S^s_{i+1} + \beta^k_{li+1} S^s_{i+1}) + \sum_{lp} \tilde{\beta}^k_{li} S^s_{i+1} S^s_{pM+i+1},
$$

completes the switch between the two types of variable.

### 3 mRNA dynamics: path integral approach

We are interested in the stochastic evolution of partially and fully processed mRNA copy numbers given the promoter dynamics (1). Such processes can be modeled via these reactions:

$$
\emptyset \overset{R(s)}{\rightarrow} m_1
$$

$$
m_j \overset{f_j}{\rightarrow} m_{j+1} \quad j = 1, \ldots, L - 1
$$

$$
m_j \overset{g_j}{\rightarrow} m_{j-1} \quad j = 2, \ldots, L
$$

$$
m_P \overset{d}{\rightarrow} \emptyset.
$$

(10)

where $m_1$ is the copy number of the newly transcribed mRNA molecules, $m_j$ ($j > 1$) is the copy number of those mRNA molecules that have undergone $j - 1$ post-transcription processes, with $m_P$ being the copy number of the fully processed mRNAs; $R(s)$ is the promoter state-dependent transcription rate, $f_j$ is the rate of conversion from mRNA species $j$ to mRNA species $j + 1$, $g_j$ is the rate of the conversion from mRNA species $j$ to mRNA species $j - 1$, and $d$ is the degradation rate of mRNA species $L$. The master equation for the entire system, reactions (1) and (10) reads

$$
\frac{d}{dt} P = MP + R[\{P(m_1 - 1) - P]\]
$$

$$
+ \sum_{j=1}^{p-1} f_j [(m_j + 1)P(m_j + 1, m_{j+1} - 1) - m_j P]
$$
Fig. 2 An example of a path \( s(t) \) for a promoter with two promoter sites acted upon by one TF. The integer values of \( s \) correspond to the following promoter states:

- \( s = 1 \) — both promoter sites are unoccupied;
- \( s = 2 \) — only promoter site A is occupied;
- \( s = 3 \) — only promoter site B is occupied;
- \( s = 4 \) — both A and B are occupied.

\[
+ \sum_{j=2}^{P} g_j \left[ (m_j + 1)P(m_j + 1, m_{j-1} - 1) - m_j P \right] \\
+ d \left[ (m_L + 1)P(m_L + 1) - m_L P \right].
\]

(11)

We have employed a short hand notation in which \( P \) is short for \( P(m_1, m_2, \ldots, m_L, t) \), \( P(m_j + 1, m_{j+1} - 1) \) is short for \( P(m_1, \ldots, m_j + 1, m_{j+1} - 1, \ldots, m_L, t) \), etc. The elements of the vector \( P_s \), are the probabilities to observe a specific set of copy numbers \((m_1, m_2, \ldots, m_P)\) and the promoter state \( s \). The matrix \( M \) is the one defined in Eq. (8).

In principal, Eq. (11) can be solved numerically by imposing upper bounds, or cutoffs, on all the variables \( m_j \). Methods for estimating such cutoffs that guarantee accuracy can be found in Munsky and Khammash (2006) and Gupta et al. (2017). Once the cutoffs are determined, the number of equations that must be solved is \( \Lambda_1 \times \Lambda_2 \times \cdots \times \Lambda_L \times M \), where \( \Lambda_j \) is the cutoff for the \( j \)th species of mRNA, and \( M \) is the dimension of \( M \) which equals the number of possible promoter states. Given a large enough \( L \), and large enough \( \Lambda_j \)s, the task of solving Eq. (11) directly may become computationally infeasible. In what follows, we present a different way of solving Eq. (11), one that reduces the dimension of the problem to \( n_j \times M \), where \( n_j \) is the copy number for the \( j \)th species of mRNA up to which we wish to know the probability distribution of \( m_j \).

Suppose that we are able to observe the state of the promoter in real time but not the stochastic evolution of the mRNA molecules. We could then write down a master equation for the variables \( m_j \) in which the transcription rate would be a known function of time:

\[
\frac{d}{dt} P = R(t) [P(m_1 - 1) - P]
\]
\[
\sum_{j=1}^{L-1} f_j \left[(m_j + 1) P(m_j + 1, m_{j+1} - 1) - m_j P\right] \\
+ \sum_{j=2}^{L} g_j \left[(m_j + 1) P(m_j + 1, m_{j-1} - 1) - m_j P\right] \\
+ d \left[(m_L + 1) P(m_L + 1) - m_L P\right],
\]  

(12)

where \( R(t) \) depends on time through the variable \( s \): \( R(t) = R(s(t)) \). Figure 2 shows an example of what \( s(t) \) might look like for a simple promoter with two binding sites and one TF. The evolution of \( s \) is sometimes referred to as “path”. In what follows, it will be more convenient to work with a generating function (GF), defined as

\[
F(\xi_1, \ldots, \xi_L, t) = \sum_{m_1} \cdots \sum_{m_L} \left(\xi_1^{m_1} \cdots \xi_L^{m_L}\right) P(m, t).
\]  

(13)

Knowing the GF, one can recover the joined PD from this relation

\[
P(m_1, \ldots, m_L, t) = \left[\frac{1}{m_1! \cdots m_L!} \frac{\partial^{m_1} \cdots \partial^{m_L}}{\partial \xi_1^{m_1} \cdots \partial \xi_L^{m_L}} F(\xi_1, \ldots, \xi_L, t)\right]_{\xi=0}.
\]  

(14)

Here, we are interested in computing PDs for each variable separately; hence, we will work with a single variable GF, defined as

\[
F_j(\xi, t) = \sum_{m_1} \cdots \sum_{m_L} \xi^{m_j} P(m, t),
\]  

(15)

from which the single variable PD can be recovered:

\[
P(m_j, t) = \left[\frac{1}{m_j!} \frac{\partial^{m_j}}{\partial \xi^{m_j}} F(\xi, t)\right]_{\xi=0}.
\]  

(16)

In reference (Albert 2019), we have shown that for a system governed by Eq. (12)

\[
F_j(\xi, t) = G_j(\xi, t) \exp \left[\int_0^t (\xi - 1) K_j(t, t') R(t') dt'\right],
\]  

(17)

where

\[
G_j(\xi, t) = \sum_{n_1} \cdots \sum_{n_L} P(n, 0) \prod_{l=1}^M \left[(\xi - 1) \sum_{i} U_{ji} U_{il}^{-1} e^{S_l t} + 1\right]^{n_i}.
\]  

(18)
\( P(n, 0) \) is the initial joint PD for all variables \( m \), \( S_i \) are the eigenvalues of the matrix

\[
S = \begin{bmatrix}
-g_1 & f_2 & 0 & \ldots & 0 \\
 g_1 & -(g_2 + f_2) & f_3 & 0 & \ldots & 0 \\
 0 & g_2 & -(g_3 + f_3) & f_4 & 0 & \ldots & 0 \\
 0 & 0 & g_3 & -(g_4 + f_4) & f_5 & \ldots & 0 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
 0 & \ldots & \ldots & \ldots & \ldots & \ldots & 0 \ g_{L-1} & -(f_L + d)
\end{bmatrix}, \tag{19}
\]

and \( \mathbf{U} \) is the unit matrix that satisfies \( [\mathbf{U}^{-1} \mathbf{S}]_{il} = S_i \delta_{il} \). In the exponent of Eq. (17), the integration kernel \( K_j(t, t') \) is given by

\[
K_j(t, t') = \sum_{i=1}^{L} e^{S_i(t-t')} \left( \mathbf{u}_j^T \mathbf{U} \mathbf{B}_i \mathbf{U}^{-1} \mathbf{u}_1 \right), \tag{20}
\]

where the elements of the diagonal matrices \( \mathbf{B}_i \) are \( [\mathbf{B}_i]_{lp} = \delta_{ii} \delta_{pi} \), and \( \mathbf{u}_j \) is the \( j \)th unit vector of the bases \( \mathbf{u}_i \). The exponential term in Eq. (17) corresponds to the Poisson solution starting from zero initial conditions, and \( G_j(\xi, t) \) encompasses the tailing off of the initial molecules.

Equation (17) is valid only for a specific path taken by \( s \). In order to obtain the PD for the variable \( m_j \), regardless of the promoter states, we must multiply Eq. (17) by the probability of observing a specific path, and then integrate over all possible paths. Thus,

\[
\mathcal{F}_j(\xi, t) = G_j(\xi, t) Q_j(\xi, t), \tag{21}
\]

where

\[
Q_j(\xi, t) = \exp \left[ \int_0^t (\xi - 1) K_j(t, t') R(s(t')) dt' \right]. \tag{22}
\]

In Albert (2019) we have proven that

\[
Q_j(\xi, t) = \left[ \sum_{i=1}^{M} \mathbf{u}_i \cdot Q_j(\xi, t', t) \right]_{t'=t}, \tag{23}
\]

where \( Q_j(\xi, t', t) \) satisfies this equation:

\[
\frac{dQ_j(\xi, t', t)}{dt'} = M \mathbf{Q}_j(\xi, t', t) + (\xi - 1) K_j(t, t') R \mathbf{Q}_j(\xi, t', t) \quad \text{for} \ t' \leq t, \tag{24}
\]

such that \( Q_j(\xi, 0, t) = \mathbf{P}(0) \), with \( \mathbf{P}(0) \) being the probability distribution for \( s \) at \( t = 0 \). The elements or the vector \( \mathbf{R} \) correspond to: \( R(s = 1), R(s = 2), \ldots \).
Note that $t'$ is a dummy variable; it is an artefact of the integral in Eq. (22). The vector function $Q_j(\xi, t', t)$ acquires physical meaning only when $t'$ is set to $t$, as stipulated by Eq. (23) (for detailed analysis see Albert 2019). In order to relate $Q_j(\xi, t', t)$ to the GF (21), we multiply Eq. (24) by $G_j(\xi, t)$ and define $	ilde{Q}_j(\xi, t', t) = G_j(\xi, t)Q_j(\xi, t', t)$, to get

$$\frac{d\tilde{Q}_j(\xi, t', t)}{dt'} = M\tilde{Q}_j(\xi, t', t) + (\xi - 1)K_j(t, t')R\tilde{Q}_j(\xi, t', t) \quad \text{for } t' \leq t,$$

such that $\tilde{Q}_j(\xi, 0, t) = G_j(\xi, t)P(0)$. Finally, the GF reads:

$$\mathcal{F}_j(\xi, t) = \left[ \sum_{i=1}^{M} u_i \cdot \tilde{Q}_j(\xi, t', t) \right]_{t'=t}. \quad (26)$$

Solving Eq. (25) is not possible; however, we can convert it into an equation for the PD for $m_j$ by applying the operator $1/(m!)\partial^m/\partial \xi^m$ and then setting $\xi = 0$. The result is this:

$$\frac{d\tilde{P}_j(m, t', t)}{dt'} = [M - RK_j(t, t')]\tilde{P}_j(m, t', t) + RK_j(t, t')\tilde{P}_j(m - 1, t', t), \quad (27)$$

where

$$\tilde{P}_j(m, t', t) = \frac{1}{m!} \left. \frac{\partial^m}{\partial \xi^m} \tilde{Q}_j(\xi, t', t) \right|_{\xi=0}. \quad (28)$$

Equation (27) must be solved for the initial conditions

$$\tilde{P}_j(m, 0, t) = \left[ P(0) \frac{1}{m!} \frac{\partial^m}{\partial \xi^m} G_j(\xi, t) \right]_{\xi=0}. \quad (29)$$

To work out Eq. (29), we can invoke Cauchy’s integral formula, which states that

$$\frac{1}{m!} \frac{d^m f(\xi)}{d\xi^m} = \frac{1}{2\pi i} \oint \frac{dz}{(z - \xi)^{m+1}} \frac{f(z)}{z}, \quad (30)$$

where $f(z)$ is analytic at the point $\xi$. The integral over the complex variable $z$ must enclose $\xi$ but is otherwise arbitrary. Replacing $f(z)$ in Eq. (30) with $G_j(\xi, t)$, setting $\xi = 0$ and performing the integration over a unit circle centered at $z = 0$, we obtain
\[
\int_0^{2\pi} \frac{d\theta}{2\pi} e^{-mi\theta} G_j(e^{i\theta}, t) = \sum_n P(n, 0) \sum_{q_1=0}^{n_1} \cdots \sum_{q_L=0}^{n_L} \prod_{l=1}^{M} (q_l^j)^{n_l} (1 - h_{jl})^{n_l - q_l} \times \int_0^{2\pi} \frac{d\theta}{2\pi} \exp \left[ i \left( \sum_{\mu=1}^{M} q_{\mu} - m \right) \theta \right] = \sum_n P(n, 0) \left[ \sum_{q_1=0}^{n_1} \cdots \sum_{q_L=0}^{n_L} \prod_{l=1}^{M} (q_l^j)^{n_l} (1 - h_{jl})^{n_l - q_l} \delta_{m, \bar{q}} \right],
\]

where

\[h_{jl} = \sum_i U_{ji} U_{il}^{-1} e^{S_i t}\]

and \(\bar{q} = \sum_{\mu} q_{\mu}\). Hence, the initial conditions for \(\tilde{P}_j(m, t', t)\) are

\[
\tilde{P}_j(m, 0, t) = \sum_n P(n, 0) \left[ \sum_{q_1=0}^{\tilde{m}_1} \cdots \sum_{q_L=0}^{\tilde{m}_L} \prod_{l=1}^{M} (q_l^j)^{n_l} (1 - h_{jl})^{n_l - q_l} \delta_{m, \bar{q}} \right] P(0).
\]

Thus, finally, the probability distribution for the variable \(m_j\) is given by

\[
P(m_j, t) = \sum_{i=1}^{M} u_i \cdot \tilde{P}_j(m, t', t) \bigg|_{t'=t}.
\]

4 Results and discussion

In order to test the validity of Eq. (27), we generated ten random samples for each of the parameter sets, \(a_i^k, b_i^k, f_i, g_i\) and \(R_{ij}\) for \(M = 2, N = 3\) and \(L = 3\). For each of the ten cases, we chose initial condition \(P(m, 0) = \delta_{m_1, \tilde{m}_1} \delta_{m_2, \tilde{m}_2} \delta_{m_3, \tilde{m}_3}\), where \(\tilde{m}_j\) was randomly selected from square distributions of integers ranging from 0 to 50. Equation (27) was solved numerically on Mathematica using the NDSolve package for \(t = 0.5, 5\) and \(t = 10\). For each parameter set, initial conditions and \(t = 0.5, 5, 10\), we generated an ensemble of 100k realizations using the GA, from which we constructed the PD for each variable. The results are presented in Fig. 3; the parameter ranges are given in the figure captions.

The advantage of the method presented herein is that it allows one to decouple the PDs for the mRNA species. As a result, our method takes us from computationally expensive or infeasible to highly efficient. One drawback of this method is that Eq. (27) must be integrated over what we termed “dummy time” from zero to the real time,
Fig. 3 Probability distributions for ten randomly selected parameter sets for $M = 2$, $N = 3$ and $L = 3$ at $t = 0.5$ min (square), $t = 5$ min (circle) and $t = 10$ min (triangle) for (a) $m_1$, (b) $m_2$, and (c) $m_3$. The parameters were sampled from square distributions with the range: $R_{ij} = [1, 50]$ min$^{-1}$, $f_i = [0.05, 0.5]$ min$^{-1}$, $g_i = [0.05, 0.5]$ min$^{-1}$, $\gamma_i^k = [0.001, 0.01]$ min$^{-1}$, $\beta_{kl,i-1} = \beta_{kl,i+1} = \gamma_i^k / \kappa_1$ min$^{-1}$, $\beta_{klp,i} = \gamma_i^k / \kappa_2$ min$^{-1}$, $K_1 = (1, 2, 3, 4)$ and $K_2 = (1, 2, 3, 4)$. The initial conditions were drawn from square distribution of integers with the range: $m_1 = [0, 50]$, $m_2 = [0, 50]$, $m_3 = [0, 50]$.

which must be set beforehand. This means that unlike the solution to the ME, which, if numerically solvable, gives us a pseudo-continuous solution in time, our method does not. To obtain a pseudo-continuous solution in time with our method, one must solve Eq. (27) for a suitable number of time points and then interpolate the solutions. However, in practice, data on probability distributions are usually available only for a
few time points; thus, in the context of single-cell mRNA data, our method is preferable to the ME or the GA.

5 Conclusion

We have presented an alternative approach to the ME for a system of an arbitrarily complex promoter and a set of mRNA species that have either partially or fully undergone the post-transcription processing. The approach consists of obtaining the generating function (GF) for the mRNAs only as a functional of a particular realization of the promoter state, and then integrating over all possible promoter states. As a result,
we derived an alternative equation for the GF, which we then converted into separate equations for the probability distribution for each species of mRNA for arbitrary initial conditions. We have demonstrated the validity of our derivations by comparing the results obtained via our method to those of Gillespie simulations. This method is highly efficient compared to other methods when the number of mRNA species is greater than one. In practice, this method lends itself to the reverse-engineering of promoter architectures based on single-cell RNA data.

References

Albert J (2016a) Stochastic simulation of reaction subnetworks: exploiting synergy between the chemical master equation and the Gillespie algorithm. In: AIP conference proceedings, vol 1790(1). p 150026
Albert J (2016b) A hybrid of the chemical master equation and the Gillespie algorithm for efficient stochastic simulations of sub-networks. PLoS ONE 11(3):e0149909
Albert J (2019) Path integral approach to generating functions for multistep post-transcription and post-translation processes and arbitrary initial conditions. Authors. J Math Biol 79(6–7):2211–2236
Albert J, Rooman M (2016) Probability distributions for multimeric systems. J Math Biol 72(1–2):157–169
Alfonsi A, Cancès E, Turinici G, Ventura BD, Huisinga W (2005) Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems. ESAIM Proc 14:1–13
Bokes P, King JR, Wood ATA, Loose M (2012a) Exact and approximate distributions of protein and mRNA levels in the low-copy regime of gene expression. J Math Biol 64(5):829–854
Bokes P, King JR, Wood ATA, Loose M (2012b) Multiscale stochastic modelling of gene expression. J Math Biol 65(3):493–520
Burrage K, Tian T, Burrage P (2004) A multi-scaled approach for simulating chemical reaction systems. Prog Biophys Mol Biol 85:217–234
Cao Y, Li H, Petzold L (2004) Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. J Chem Phys 121:4059
Cao Y, Gillespie DT, Petzold LR (2005a) Avoiding negative populations in explicit Poisson tau-leaping. J Chem Phys 123(5):054104
Cao Y, Gillespie DT, Petzold LR (2005b) Efficient step size selection for the tau-leaping simulation method. J Chem Phys 124(4):044109
Duso L, Zechner C (2018) Selected-node stochastic simulation algorithm. J Chem Phys 148:164108
Fiers MWEJ, Minnoye L, Aïbar S, Gonzalez-Blas CB, Atak ZK, Aerts S (2018) Mapping gene regulatory networks from single-cell omics data. Brief Funct Genom 17(4):246–254
Gibson MA, Bruck J (2000) Efficient exact stochastic simulation of chemical systems with many species and many channels. J Phys Chem 104(9):1876–1889
Gillespie DT (1977) Exact stochastic simulation of coupled chemical reactions. J Phys Chem 81(25):2340–2361
Gillespie DT (2001) Approximate accelerated stochastic simulation of chemically reacting systems. J Chem Phys 115(4):1716
Gupta A, Mikelson J, Khammash M (2017) A finite state projection algorithm for the stationary solution of the chemical master equation. J Chem Phys 147(15):154101
Jahnke T, Altıntan D (2010) Efficient simulation of discrete stochastic reaction systems with a splitting method. BIT Num Math 50(4):797–822
Jahnke T, Huisinga W (2007) Solving the chemical master equation for monomolecular reaction systems analytically. J Math Biol 54(1):1–26
Kurasov P, Lück A, Magnolo D, Wolf V (2018) Stochastic hybrid models of gene regulatory networks. Math Biosci 305:170–177
Munsky B, Khammash M (2006) The finite state projection algorithm for the solution of the chemical master equation. J Chem Phys 124:044104
Pendar H, Platini T, Kulikarni RV (2013) Exact protein distributions for stochastic models of gene expression using partitioning of Poisson processes. Phys Rev E 87:042720
Popovic N, Marz C, Swain PS (2016) A geometric analysis of fast-slow models for stochastic gene expression. J Math Biol 72(1–2):87–122
Shahrezaei V, Swain PS (2008) Analytical distributions for stochastic gene expression. PNAS 105(45):17256–17261
Van Kampen NG (2007) Stochastic processes in physics and chemistry 3rd print. North Holland, Amsterdam Veerman F, Marr C, Popović N (2018) Time-dependent propagators for stochastic models of gene expression: an analytical method. J Math Biol 77(2):261–312

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.