Mathematical Biosciences 269 (2015) 10–16

Contents lists available at ScienceDirect

Mathematical Biosciences

journal homepage: www.elsevier.com/locate/mbs

Solving the chemical master equation by a fast adaptive finite state projection based on the stochastic simulation algorithm

R.B. Sidje*, H.D. Vo

Department of Mathematics, The University of Alabama, Tuscaloosa, AL 35487, USA

ARTICLE INFO

Article history:
Received 23 June 2015
Revised 20 August 2015
Accepted 20 August 2015
Available online 28 August 2015

Keywords:
Chemical master equation
Model reduction
Matrix exponential
Stochastic simulation algorithm
Finite state projection

ABSTRACT

The mathematical framework of the chemical master equation (CME) uses a Markov chain to model the biochemical reactions that are taking place within a biological cell. Computing the transient probability distribution of this Markov chain allows us to track the composition of molecules inside the cell over time, with important practical applications in a number of areas such as molecular biology or medicine. However the CME is typically difficult to solve, since the state space involved can be very large or even countably infinite. We present a novel way of using the stochastic simulation algorithm (SSA) to reduce the size of the finite state projection (FSP) method. Numerical experiments that demonstrate the effectiveness of the reduction are included.

1. Introduction

Models of cellular processes promise great benefits in important fields such as molecular biology or medicine. When molecules exist in large numbers, models can be formulated using concentrations. And since concentrations are continuous quantities, this way of doing so allows using reaction rate equations (RREs), which are deterministic, nonlinear ordinary differential equations. Within a cell, however, some key regulatory molecules exist only in small numbers, in which case it becomes more appropriate to formulate the models in a discrete and stochastic setting. The mathematical framework that underpins this is a continuous-time, discrete-state, Markov process, and computing its transient probability distribution amounts to solving the chemical master equation (CME).

While promising many insights, the CME is difficult to solve, especially for large models. Consequently, researchers often resort to simulating trajectories, using most notably Gillespie's stochastic simulation algorithm (SSA) [2] or its improved variants, e.g., [1].

We outline a novel solution technique here, namely an SSA-driven reduction of the state space that builds on the principle that the CME aims at computing a probability vector $\boldsymbol{p} = (p_1, \ldots, p_n)^T \in [0,1]^n$, with components that sum to one, $\sum_{i=1}^{n} p_i = 1$. A key consequence of this is that if numerous nonnegative real numbers sum to one, then some of these numbers are necessarily zero or small. That is, for very large problems, $n \gg 1$, the probability sum condition implies that some of the components must necessarily be zero or negligible, and precisely because the CME has such extremely high dimension, the more of them there will be. As our paper will show, dropping those negligible components allows us to reduce the size dramatically.

We build on the finite state projection (FSP) of Munsky and Kammash [9] and the Krylov-FSP [8] that was an early improvement to the basic FSP. We note that there have been other developments targeted at special problems such as those with a clear separation of timescales [14], or those reducible to one-dimensional processes [15]. Applying the quasi steady state assumption when possible can also reduce the state space in some cases [6,16]. Several other general studies such as [5,12,13] have investigated ways to speed up the calculations. But solving the CME of realistic models with many species and many reactions remains a formidable challenge to all methods. Our new SSA-based reduction drives the FSP in a fast, economical, and adaptive way.

The organization of the paper is as follows. Section 2 motivates this study. Section 3 gives some background on modeling biochemical reactions by way of the CME. Section 4 summarizes the basic FSP method and introduces our proposed SSA-driven approach. Section 5 describes the original Kylov-FSP method and how to incorporate our new reduction. Section 6 shows the performance of our method on some examples. Section 7 finally gives some concluding remarks.

2. Challenge and motivation

What makes the CME so hard to deal with is because it is exhaustive—it enumerates *all* the possible states that the cell can ever have, resulting in a size that is extremely large and difficult to handle.

* Corresponding author.+12053487007.
E-mail addresses: roger.b.sidje@ua.edu (R.B. Sidje), hvo@crimson.ua.edu (H.D. Vo).

http://dx.doi.org/10.1016/j.mbs.2015.08.010
0025-5564/

In reality, however, at any given time, there could be several states that are unlikely and that could be safely ignored in the calculations. These could be past states that the cell already had and is unlikely to have again, or they could be future states that are too far in the future to be relevant at present. By ignoring those improbable states, we can reduce the size to a tractable level. While this principle is quite promising [6,9,12], it raises the challenging points that we discuss next.

First, states that may be improbable right now can become probable the next instant because the makeup of the cell is continuously changing as a result of the reactions that are taking place within it. Hence the subset of probable states is changing over time, but the merit of our approach hinges on the core principle that this subset remains small compared to the set of all possible states that the cell can ever have and that are encoded in the full blown CME. The key to efficiency is therefore to only track this reduced subset and deliberately ignore unlikey states, in a way that is adjusted dynamically over time.

The second point is related to the first point in the following sense: how then do we distinguish probable states from improbable ones? If we do a good job in the selection, we speed up the calculations. However, if we mistakenly exclude relevant states, we ultimately end up with unreliable answers, or even worse, erroneous or meaningless answers that we may wrongly believe to be right. This is a serious concern that is not simple to resolve because a sorting strategy may come at a significant overhead due to the fact that we deal with extremely large models over possibly thousands of time steps.

Our research is bringing fresh ideas from advanced computing and differential equations into the field of stochastic chemical kinetics. We are investigating inexact (or relaxed) techniques with other novel reduction techniques. What we present in this work is a fundamental cornerstone that is shaping our investigations.

3. The chemical master equation

As mentioned earlier, when there are small numbers of some molecular species, as is the case for many key regulatory elements in biological systems, it is more appropriate to use a discrete and stochastic framework to model such systems. The underlying Markov chain is briefly described in this section, which also serves to outline the terminology and notation.

Consider a biochemical system with $N \geq 1$ different chemical species that are interacting via $M \geq 1$ chemical reactions. The state of the system is a vector $\boldsymbol{x} \equiv (x_1, \ldots, x_N)^T$ of nonnegative integers where x_i is the current population of the ith species. Transitions between states occur when (and only when) a reaction occurs. Each reaction $1 \leq j \leq M$ entails a *stoichiometric* vector $\boldsymbol{\nu}_j$ that represents the change when the reaction occurs; if the system is in state $\boldsymbol{x}$ and reaction j occurs, then the system transitions to state $\boldsymbol{x} + \boldsymbol{\nu}_j$. Each state $\boldsymbol{x}$ entails M *propensities* $\alpha_1(\boldsymbol{x}), \ldots, \alpha_M(\boldsymbol{x})$ that determine the relative chance of each reaction occurring if the system is in state $\boldsymbol{x}$. The propensities are defined by the requirement that, given $\boldsymbol{x}(t) = \boldsymbol{x}$, $\alpha_j(\boldsymbol{x})dt$ is the probability of reaction j occurring in the next infinitesimal time interval $[t, t + dt)$, where the dependence on time has now been made explicit.

Let the probability of being in state $\boldsymbol{x}$ at time t be denoted by $P(\boldsymbol{x}; t)$ and consider the way that this changes over time. With appropriate assumptions, it can be shown, see for example Higham [4] for a readable overview, that for each state $\boldsymbol{x}$, the previous description of the model implies that this probability satisfies the CME, which is the following discrete, parabolic, partial differential equation:

$$\frac{\partial P(\boldsymbol{x}; t)}{\partial t} = \sum_{j=1}^{M} \alpha_j(\boldsymbol{x} - \boldsymbol{\nu}_j) P(\boldsymbol{x} - \boldsymbol{\nu}_j; t) - P(\boldsymbol{x}; t) \sum_{j=1}^{M} \alpha_j(\boldsymbol{x}).$$

It may be written in an equivalent matrix–vector form by enumerating all the states. In this case, if there are n possible states, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, the CME takes the form of a system of linear ordinary differential equations (ODEs)

$$\begin{cases} \boldsymbol{p}'(t) = \boldsymbol{A}\boldsymbol{p}(t), & t \in [0,\ t_f] \\ \boldsymbol{p}(0) = \boldsymbol{p}_0, \end{cases} \tag{1}$$

where the probability vector $\boldsymbol{p} = (p_1, \ldots, p_n)^T$ is such that each component $p_i = P(\boldsymbol{x}_i, t) = \text{Prob}\{\boldsymbol{x}(t) = \boldsymbol{x}_i\}$, the probability of being at state $\boldsymbol{x}_i$ at time t, for $i = 1, \ldots, n$. It is customary to equivalently identify a state $\boldsymbol{x}_i$ just by its index i in the enumeration. We shall use both identifications interchangeably. The vector $\boldsymbol{p}_0 = \boldsymbol{p}(0)$ is an initial probability distribution and $\boldsymbol{A}$ is a sparse n-by-n matrix representing the infinitesimal generator of the Markov chain that underpins the CME. For $i \neq j$, the entry a_{ij} holds the propensity for the system to transition to state i from state j, or 0 if unreachable. Thus the a_{ij}, for $i \neq j$, are nonnegative. And to be a valid infinitesimal generator, the diagonal terms are defined as $a_{jj} = -\sum_i a_{ij}$, which means that $\boldsymbol{A}$ has zero column sum and so probability is conserved. The solution of (1) at time t is

$$\boldsymbol{p}(t) = \exp(t\boldsymbol{A})\,\boldsymbol{p}_0, \tag{2}$$

where $\exp(t\boldsymbol{A}) = \sum_{n=0}^{\infty} \frac{(t\boldsymbol{A})^n}{n!}$ is the usual matrix exponential represented in Taylor series (among other equivalent definitions). Since there is no particular structure common to all the models, calculations cannot rely on generic simplifications.

4. Model reduction

4.1. FSP method

With the CME prone to have a huge state space and in theory, even a potentially countably infinite set of states, a crucial goal is how to effectively reduce the state space in a meaningful way that still captures enough of the cell dynamics. The finite state projection (FSP) algorithm of Munsky and Khammash [9] is one such model reduction method that we summarize here since we build on it. The principle is to truncate the system to a smaller subsystem that captures enough information while remaining tractable. Let $J = \{1, \ldots, k\}$, then the matrix in (2) is replaced by $\boldsymbol{A}_J$ where

$$\boldsymbol{A} = \left(\begin{array}{c|c} \boldsymbol{A}_J & * \\ \hline * & * \end{array}\right) \in \mathbb{R}^{n \times n},$$

i.e., with $k = |J|$ being the cardinality of J, $\boldsymbol{A}_J$ is a $k \times k$ submatrix of the true operator $\boldsymbol{A}$. The states indexed by J then form the *finite state projection*. The FSP method takes

$$\boldsymbol{p}(t_f) \approx \exp(t_f \boldsymbol{A}_J)\,\boldsymbol{p}_J(0). \tag{3}$$

Note the subscript J that characterizes the truncation just described and note that the initial distribution is truncated similarly. In fact, more generally, $\boldsymbol{A}_J$ need not simply be a *principal* submatrix. Rather, assume that J is an arbitrary subset of $\{1, \ldots, n\}$ and that for consistency $\boldsymbol{A}_J$ is defined with the same size as the matrix $\boldsymbol{A} = [a_{ij}]$ using

$$(\boldsymbol{A}_J)_{ij} = \begin{cases} a_{ij} & \text{if } i, j \in J \\ 0 & \text{if } i \notin J \text{ or } j \notin J. \end{cases}$$

Similarly, $\boldsymbol{p}_J$ is defined from $\boldsymbol{p} = (p_1, \ldots, p_n)^T$ using

$$(\boldsymbol{p}_J)_i = \begin{cases} p_i & \text{if } i \in J \\ 0 & \text{otherwise.} \end{cases}$$

The FSP approximation still takes the form (3). Obviously, all computations are done in practice on the effectively truncated matrix, justifying why the FSP is a reduction method. Conceptually, such a truncation can be understood as an implicit reordering $\boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P}$ where $\boldsymbol{P}$ is an appropriate permutation matrix that makes the effectively truncated matrix a principal submatrix. Munsky and Khammash [9]