Exact Computation of Probability Landscape of Stochastic Networks of Single Input and Coupled Toggle Switch Modules

Anna Terebus[1], Youfang Cao[2] and Jie Liang[3]

Accepted, Conf Proc IEEE Eng Med Biol Soc. 2014

***Abstract*— Gene regulatory networks depict the interactions between genes, proteins, and other components of the cell. These interactions often are stochastic that can influence behavior of the cells. Discrete Chemical Master Equation (dCME) provides a general framework for understanding the stochastic nature of these networks. However solving dCME is challenging due to the enormous state space, one effective approach is to study the behavior of individual modules of the stochastic network. Here we used the finite buffer dCME method and directly calculated the exact steady state probability landscape for the two stochastic networks of Single Input and Coupled Toggle Switch Modules. The first example is a switch network consisting of three genes, and the second example is a double switching network consisting of four coupled genes. Our results show complex switching behavior of these networks can be quantified.**

I. INTRODUCTION

Gene regulatory circuits control essential cellular processes including cellular fate. A well known example is stochastic switch between the lysogenic state and the lytic state in phage lambda [1]. Another example is the transition into and from competence in the *Bacillus subtilis* [2].

Studying stochastic gene regulatory networks is challenging, as reactions often involve low copy number of molecules and may have large separation in time scale. The discrete Chemical Master Equation (dCME) provides a general framework for modeling of stochastic gene networks. However solution of the dCME remains difficult, as analytical solution is generally not possible. Computational methods, on the other hand, encounter the problem of enormous state space. For example for the system with 15 molecular species, each of which has 10 molecules at most, the state space size is 10^{15}, and correspondingly the system of 10^{15} ordinary differential equations has to be solved. Therefore it is necessary to truncate the state space with the hope of maintaining sufficient accuracy. The finite state projection provides a method for directly solving the time evolution, however it cannot be used to calculate steady state distribution because of the introduction of the absorbing state, whose probability increases with time [3].

Here we apply the previously described Finite Buffer method, which allows efficiently enumerating state space according to predefined error tolerance, to directly calculate the steady state solution of dCME for two modules of gene regulatory networks. The first example is the single input module, consisting of three genes in which the product of the first gene inhibits the expression of the two other genes. At the same time the inhibiting gene is activated by these two other genes it inhibits. Another example is that of two coupled toggle switch. The four genes in the system are connected pairwise, so protein product can be produced when corresponding pairs of genes are inhibited. The first model consists of 7 molecular species, and the second system 12 species. Both systems are general network motifs widely found in biological systems. Our results show that the exact steady state probability landscape of two these networks can be obtained using Finite Buffer method.

II. MODELS AND METHODS

A. Discrete Chemical Master Equation

Consider a well-mixed biochemical system with constant volume and temperature. Assume this system contains n molecular species X_i which participate in m reactions R_k with reaction rate constants r_k. The microstate of the system at time t is represented by the non-negative integer column vector of copy numbers of each molecular species: $\boldsymbol{x}(t) = (x_1(t), x_2(t), \cdots, x_n(t))^T$, where ^{T} denotes the transpose. An arbitrary reaction R_k ($k = 1, 2, \cdots, m$) with intrinsic rate r_k takes the general form:

$$c_{1k}X_1 + \cdots + c_{nk}X_n \xrightarrow{r_k} c'_{1k}X_1 + \cdots + c'_{nk}X_n,$$

which brings the system from a microstate $\boldsymbol{x}_i$ to $\boldsymbol{x}_j$. The difference between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is the stoichiometry vector $\boldsymbol{s}_k$ of the reaction R_k: $\boldsymbol{s}_k = \boldsymbol{x}_j - \boldsymbol{x}_i = (s_{1k}, s_{2k}, \cdots, s_{nk})^T = (c'_{1k} - c_{1k}, c'_{2k} - c_{2k}, \cdots, c'_{nk} - c_{nk})^T \in \mathbb{Z}^n$. The stoichiometry matrix $\boldsymbol{S}$ for the reaction network is defined as: $\boldsymbol{S} = (\boldsymbol{s}_1, \boldsymbol{s}_2, \cdots, \boldsymbol{s}_m) \in \mathbb{Z}^{n\times m}$, where each column represents a single reaction. The rate $A_k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ of reaction R_k that transforms microstate from $\boldsymbol{x}_i$ to $\boldsymbol{x}_j$ is determined by the intrinsic rate constant r_k and the combination number of relevant reactants in the current microstate $\boldsymbol{x}_i$: $A_k(\boldsymbol{x}_i, \boldsymbol{x}_j) = A_k(\boldsymbol{x}_i) = r_k \prod_{l=1}^{n} \begin{pmatrix} x_l \\ c_{lk} \end{pmatrix}$.

All possible microstates that the system can visit from a given initial condition over time t form the state space:

*This work was supported by NIH Grants GM079804, GM086145 and NSF Grants DBI 1062328 and DMS-0800257. This work was also funded by the Chicago Biomedical Consortium with support from the Searle Funds at The Chicago Community Trust

[1]A. Terebus is with the Bioinformatics Program, Department of Bioengineering, University of Illinois at Chicago, Chicago, IL, 60607, USA atereb2@uic.edu

[2]Y. Cao is a Visiting Research Assistant Professor, Department of Bioengineering, University of Illinois at Chicago, Chicago, Chicago, IL, 60607, USA youfang@uic.edu

[3]J. Liang is on Faculty of Department of Bioengineering, University of Illinois at Chicago, Chicago, IL, 60607, USA jliang@uic.edu

$\mathcal{S} = \{\boldsymbol{x}(t)|\boldsymbol{x}(0), t \in (0, \theta)\}$. We denote the probability of each microstate at time t as $p(\boldsymbol{x}(t))$, and the probability distribution at time t over the whole state space as $\boldsymbol{p}(t) = \{(p(\boldsymbol{x}(t))|\boldsymbol{x}(t) \in \mathcal{S})\}$. And, $\boldsymbol{p}(t)$ is also called the *probability landscape* of the network [1].

Discrete chemical master equation (dCME) is a set of linear ordinary differential equations describing the probability changes of each discrete microstate of the system over time. The dCME of an arbitrary microstate $\boldsymbol{x} = \boldsymbol{x}(t)$ is:

$$\frac{dp(\boldsymbol{x})}{dt} = \sum_{\boldsymbol{x}'} [A(\boldsymbol{x}', \boldsymbol{x})p(\boldsymbol{x}') - A(\boldsymbol{x}, \boldsymbol{x}')p(\boldsymbol{x})] \qquad (1)$$

where $\boldsymbol{x}' \neq \boldsymbol{x}$.

The Eqn. (1) can be further represented in matrix form:

$$\frac{d\boldsymbol{p}(t)}{dt} = \boldsymbol{A}^T \boldsymbol{p}(t).$$

For any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{S}$, where $\boldsymbol{A} \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ is called the transition rate matrix formed by the collection of all $A(\boldsymbol{x}_i, \boldsymbol{x}_j)$:

$$A = \|A(\boldsymbol{x}_i, \boldsymbol{x}_j)\| = \begin{cases} -\sum\limits_{\substack{\boldsymbol{x}' \in \mathcal{S}, \\ \boldsymbol{x}' \neq \boldsymbol{x}_i}} A_k(\boldsymbol{x}_i, \boldsymbol{x}'), & \boldsymbol{x}_i = \boldsymbol{x}_j, \\ A_k(\boldsymbol{x}_i, \boldsymbol{x}_j), & \boldsymbol{x}_i \neq \boldsymbol{x}_j. \end{cases}$$

B. Finite Buffer State Space Enumeration Method with Multiple Buffers

We have developed an algorithm previously to optimally enumerate state space of arbitrary biological network, and solve the steady state probability landscape of the dCME, when an initial state is given [1], [4]. When the network is an open system, *i.e.*, containing synthesis and degradation reactions, one finite buffer of virtual molecules is assigned to the network to limit the total copy number of species that can be synthesized.

However, to more efficiently enumerate the state space our finite buffer method can be further improved by using multiple buffers and empirically estimating the error of state space truncation for each individual buffer. This novel method has been developed in [5]. Briefly, we can partition reactions into different independent reaction groups (IRG), each of which contains reactions sharing the common species participating in synthesis and degradation type processes. We then assign different buffers to each different IRG. We can estimate the error of the dCME solution due to the finite buffer size by calculating the total probability of boundary states, which are microstates with at least one buffer depleted. Therefore, each IRG can be bounded by a separate buffer, and the minimal buffer size can be determined by comparing the error estimate of the buffer to the desired error tolerance. If the estimated error is larger that the error tolerance, the buffer size needs to be increased. Otherwise, the buffer size can be reduced to save memory space.

The error of each buffer is related to the ratio between synthesis and degradation reaction rate constants in the corresponding IRG. When the ratio is larger, the IRG has larger error with the same buffer size, or equivalently, a larger buffer is required for the IRG to achieve the same error. We develop an approach to estimate the size of each buffer *a priori* as $2 \times s/d$, where s and d are the synthesis and degradation rate constants in the IRG. We then iteratively adjust the buffer size until the pre-defined error tolerance is reached.

In the Results section, we study two important gene regulatory networks using this improved finite buffer method. We show the estimation of buffer sizes based on synthesis/degradation ratio in each IRG, as well as the steady state probability landscapes of the dCME.

III. Results

A. Single Input Network module

Single input network motif can be found in many biological networks, in which multiple genes are regulated by the expression of a single transcription factor [7]. Here we study a simple network of three genes with two of them controlled by a master gene (Fig. 1). The molecular species, reactions and their rate constants are shown below:

$$R_1 : A + GeneB \xrightarrow{k_1} BGeneB,$$

$$R_2 : BGeneB \xrightarrow{k_2} A + GeneB,$$

$$R_3 : A + GeneC \xrightarrow{k_3} BGeneC,$$

$$R_4 : BGeneC \xrightarrow{k_4} A + GeneC,$$

$$R_5 : GeneB + GeneC \xrightarrow{k_5} A + GeneB + GeneC,$$

$$R_6 : A \xrightarrow{k_6} \emptyset,$$

$$R_7 : GeneB \xrightarrow{k_7} B + GeneB,$$

$$R_8 : B \xrightarrow{k_8} \emptyset,$$

$$R_9 : GeneC \xrightarrow{k_9} C + GeneC,$$

$$R_{10} : C \xrightarrow{k_{10}} \emptyset.$$

This model consists of three genes $GeneA$, $GeneB$, and $GeneC$, expressing protein products A, B and C, respectively. Protein monomer A can bind to promoter sites of $GeneB$ and $GeneC$ to form protein-DNA complexes $BGeneB$ and $BGeneC$, respectively, to turn off the expression of the other gene. At the same time, both genes $GeneB$ and $GeneC$ activate the expression of $GeneA$, so protein A can be synthesized if the binding sites of both $GeneB$ and $GeneC$ are not occupied. We take the parameters as: $k_1 = k_3 = 0.005/s$, $k_2 = k_4 = 0.1/s$, $k_5 = 20/s$, $k_7 = 10/s$, $k_9 = 11/s$, $k_6 = k_8 = k_{10} = 1/s$.

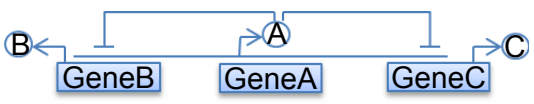

Fig. 1. Single Input Network module

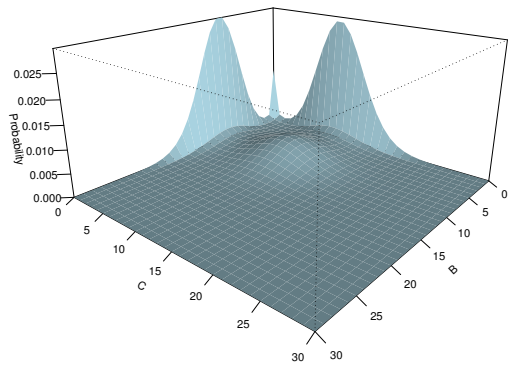

Fig. 2. Steady state probability landscape for proteins B and C

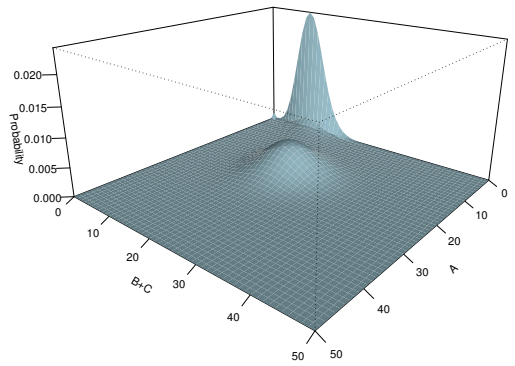

Fig. 3. Steady state probability landscape for proteins A and B+C

Three IRGs are identified for this model as $\mathcal{R}_1^{IRG} = \{R_1, \cdots, R_6\}$, $\mathcal{R}_2^{IRG} = \{R_7, R_8\}$, and $\mathcal{R}_3^{IRG} = \{R_9, R_{10}\}$. Each is assigned a separate buffer.

We predefine the error tolerance for all of the buffers to be 1×10^{-5}. When estimating the error for $\mathcal{R}_1^{IRG}$, we consider the extreme cases in which protein A is synthesized at the maximum rate, but degraded at the minimum rate. This corresponds to the case in which $GeneB$ and $GeneC$ are constantly turned off. We pre-estimate the buffer size of $\mathcal{R}_1^{IRG}$ as $2 \times k_5/k_6 = 40$. We further reduce the error by increasing the buffer size by 1 at a time, if the boundary probability is larger than the tolerance 1×10^{-5}. Otherwise, if the boundary probability is smaller than the tolerance, we decrease the buffer size by 1 at a time to achieve further saves on memory space. We obtain the final minimal buffer size for $\mathcal{R}_1^{IRG}$ to be 44. Similarly we obtain the minimal buffer sizes for the other two IRGs $\mathcal{R}_2^{IRG}$ and $\mathcal{R}_3^{IRG}$ to be 27 and 28, respectively. The enumerated state space consists of $142,912$ states. The sparse transition rate matrix contains a total of $1,016,135$ non-zero elements.

For this switch network, the steady state probability was computed and shown on Fig. 2, 3, with errors for each IRG as: $\mathcal{R}_1^{IRG}$: 8.1739×10^{-7}, $\mathcal{R}_2^{IRG}$: 3.3932×10^{-6} and $\mathcal{R}_3^{IRG}$: 6.4288×10^{-6}, which are all smaller than the predefined error tolerance 1×10^{-5}.

The computed steady state probability landscape of species B and C is plotted on Fig. 2, in which the switch between proteins B and C can be seen. When $GeneB$ ($GeneC$) is bound, protein A synthesis is suppressed, which leads to the reduction of its concentration. The probability of $GeneC$ ($GeneB$) to be repressed decreases, and the number of molecules of protein C (B) increases. Fig. 3 shows the expression level of $GeneA$ versus the total expression level of $GeneB$ and $GeneC$. Oscillating behaviors can be inferred of this network. Probability of the expression of $GeneB$ and $GeneC$ is high, when the concentration of the protein A is low. When both genes $GeneB$ and $GeneC$ are unbound, the concentration of the protein A increases. We can therefore infer the following scenario: the increase of the protein A leads to the increase of the probability of $GeneB$ or $GeneC$ to be bound, but once one of them is inhibited, it leads to the immediate reduction of the amount of molecules of the protein A.

B. Two coupled toggle switch network

Toggle switches are an important class of biological networks playing critical roles in many biological processes, such as cell fate determination [4]. Here we studied the behavior of a biological network consisting of two coupled toggle switches (Fig. 4). The molecular species, reactions and their rate constants are shown below:

$$R_1 : C + GeneA \xrightarrow{k_1} BGeneA,$$
$$R_2 : BGeneA \xrightarrow{k_2} C + GeneA,$$
$$R_3 : D + GeneB \xrightarrow{k_3} BGeneB,$$
$$R_4 : BGeneB \xrightarrow{k_4} D + GeneB,$$
$$R_5 : A + GeneC \xrightarrow{k_5} BGeneC,$$
$$R_6 : BGeneC \xrightarrow{k_6} A + GeneC,$$
$$R_7 : B + GeneD \xrightarrow{k_7} BGeneD,$$
$$R_8 : BGeneD \xrightarrow{k_8} B + GeneD,$$
$$R_9 : GeneD + GeneA \xrightarrow{k_9} A + GeneD + GeneA,$$
$$R_{10} : A \xrightarrow{k_{10}} \emptyset,$$
$$R_{11} : GeneB + GeneA \xrightarrow{k_{11}} B + GeneB + GeneA,$$
$$R_{12} : B \xrightarrow{k_{12}} \emptyset,$$
$$R_{13} : GeneB + GeneC \xrightarrow{k_{13}} C + GeneB + GeneC,$$
$$R_{14} : C \xrightarrow{k_{14}} \emptyset,$$
$$R_{15} : GeneC + GeneD \xrightarrow{k_{15}} D + GeneC + GeneD,$$
$$R_{16} : D \xrightarrow{k_{16}} \emptyset.$$

This model consists of four genes $GeneA$, $GeneB$, $GeneC$, and $GeneD$, expressing protein products A, B, C, and D, respectively, in the way that $GeneA$ and $GeneC$, $GeneB$ and $GeneD$ repress each other pairwise. Namely $GeneA$ ($GeneB$) product monomer A (B) turns off the expression

of $GeneC$ ($GeneD$), when forming protein-DNA complex $BGeneC$ ($BGeneD$), analogously $GeneC$ ($GeneD$) product monomer C (D) turns off the expression of $GeneA$ ($GeneB$), when forming protein-DNA complex $BGeneA$ ($BGeneB$). In the same time protein A can be synthesized, if the binding sites of both $GeneA$ and $GeneD$ are not occupied, protein B can be synthesized if the binding sites of both $GeneA$ and $GeneB$ are not occupied, protein C can be synthesized if the binding sites of both $GeneC$ and $GeneB$ are not occupied, protein D can be synthesized if the binding sites of both $GeneD$ and $GeneA$ are not occupied. We take the parameters as: $k_1 = k_3 = k_5 = k_7 = 0.006/s$, $k_2 = k_4 = k_6 = 0.1/s$, $k_9 = k_{13} = 3/s$, $k_{11} = k_{15} = 4/s$, $k_{10} = k_{12} = k_{14} = k_{16} = 1/s$. Four IRGs

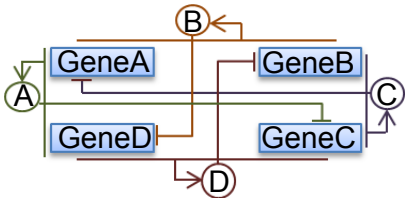

Fig. 4. Two coupled toggle switch network

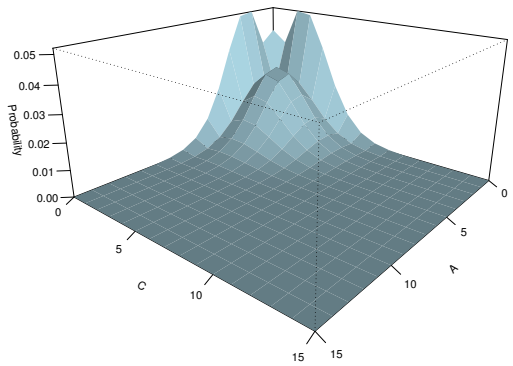

Fig. 5. Steady state probability landscape for proteins A and C

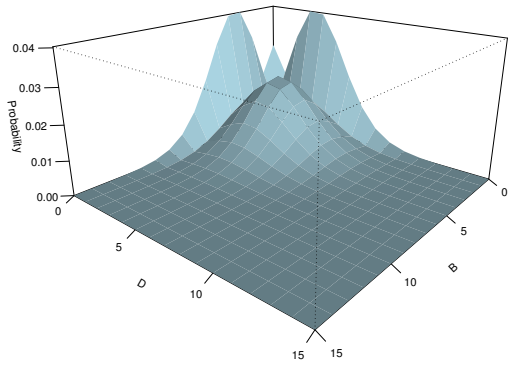

Fig. 6. Steady state probability landscape for proteins B and D

are identified for this model: $\mathcal{R}_1^{IRG} = \{R_5, R_6, R_9, R_{10}\}$, $\mathcal{R}_2^{IRG} = \{R_7, R_8, R_{11}, R_{12}\}$, $\mathcal{R}_3^{IRG} = \{R_1, R_2, R_{13}, R_{14}\}$, $\mathcal{R}_4^{IRG} = \{R_3, R_4, R_{15}, R_{16}\}$, and each is assigned a separate buffer.

We predefine the error threshold for all of the buffers to be equal 1×10^{-5}. When estimating the error for $\mathcal{R}_1^{IRG}$, we consider the extreme cases in which A is synthesized at the maximum rate, but degraded at the minimum rate. This corresponds to the $GeneB$ is constantly turned off. Following the same approach as in the first example, we determine the minimal buffer sizes that can satisfy the predefined error tolerance $\epsilon = 10^{-5}$ for all four IRGs $\mathcal{R}_1^{IRG}$, $\mathcal{R}_2^{IRG}$, $\mathcal{R}_3^{IRG}$, and $\mathcal{R}_4^{IRG}$ to be $15, 17, 15$, and 17, respectively. The enumerated state space consists of $1,177,225$ states. The sparse transition rate matrix contains a total of $11,339,253$ non-zero elements.

For this switch network, the steady state probability was found and shown on Fig. 5, 6 with errors for each IRG: $\mathcal{R}_1^{IRG}$: 5.8808×10^{-7}, $\mathcal{R}_2^{IRG}$: 9.7086×10^{-7}, $\mathcal{R}_3^{IRG}$: 5.8808×10^{-7}, and $\mathcal{R}_2^{IRG}$: 9.7086×10^{-7} which are smaller than the predefined error tolerance 10^{-5}.

The steady state probability landscape of species A and C is plotted on Fig. 5. The steady state probability landscape of species B and D is plotted on Fig. 6. Switching behavior of each gene pairs is shown in both plots. For example, for the pair of proteins A and C (Fig. 5), we can observe that the increase of the concentration of the protein C leads to the increase of the probability of $GeneA$ to be bound, as well as the reduction of the concentration of the protein A. There is an opposite effect as well: the increase of the concentration of protein A leads to the decrease of the concentration of the protein C.

IV. Conclusion

Here we present results of exact calculation of steady state probability landscape of two stochastic network modules that are widely found in biological circuits [7]. Our results show that their probability landscape can be studied in details using the Finite Buffer dCME Method.

V. Acknowledgments

Support from NIH Grants GM079804, GM086145 and NSF Grants DBI 1062328 and the Chicago Biomedical Consortium are gratefully acknowledged.

References

[1] Cao Y, Lu H-M and Liang J (2010) Probability landscape of heritable and robust epigeneric state of lysogeny in phage lambda. *Proceedings of the National Academy of Sciences of the United States of America*, **107**(43):18445–18450.

[2] Schultz D, Jacob E-B, Onuchic J. N. and Wolynes G (2007) Molecular level stochastic model for competence cycles in *Bacillus subtilis*. *Proceedings of the National Academy of Sciences of the United States of America*, **30**(1):17582–17587.

[3] Munsky B, and Khamash M (2006) The finite state projection algorithm for the solution of the chemical master equation. *The Journal of Chemical Physics*, **124**(4), 044104.

[4] Cao Y, and Liang J (2008) Optimal enumeration of state space of finitely buffered stochastic molecular networks and exact computation of steady state landscape probability. *BMC Systems Biology*, **2**(1):30.

[5] Cao Y, Terebus A and Liang J (2014) Direct Solution of Discrete Chemical Master Equation For Both Time Evolution and Steady State Using Multi-Finite Buffer State Space Enumeration Method *Manuscript*.

[6] Tian J P, Kannan D (2006) Lumpability and commutativity of Markov processes. *Stochastic analysis and Applications*, **24**(3), 585-702.
[7] Alon U. (2007) Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, **(**8), 450-461.