Hybrid CME–ODE method for efficient simulation of the galactose switch in yeast

David M. Bianchi¹,², Joseph R. Peterson¹, Tyler M. Earnest¹,³,₄, Michael J. Hallock³,₄, Zaida Luthey-Schulten²,³,⁵,⁶,⁷

¹Department of Chemistry, University of Illinois at Urbana–Champaign, 505 S Mathews Ave, Urbana, USA
²NSF Center for the Physics of Living Cells, 1110 W Green St, MC-704, 320 Loomis Laboratory, Urbana, USA
³NIH Center for Macromolecular Modeling and Bioinformatics, University of Illinois at Urbana–Champaign, 405 N Mathews Ave, Urbana, USA
⁴School of Chemical Sciences, University of Illinois at Urbana–Champaign, 505 S Mathews Ave, Urbana, USA
⁵Center for Biophysics and Quantitative Biology, University of Illinois at Urbana–Champaign, 1110 W Green St, MC-704, Urbana, USA
⁶Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana–Champaign, 405 N Mathews Ave, Urbana, USA
⁷Department of Physics, University of Illinois at Urbana–Champaign, 1110 W Green St, MC-704, Urbana, USA
E-mail: zan@illinois.edu

Abstract: It is well known that stochasticity in gene expression is an important source of noise that can have profound effects on the fate of a living cell. In the galactose genetic switch in yeast, the unbinding of a transcription repressor is induced by high concentrations of sugar particles activating gene expression of sugar transporters. This response results in high propensity for all reactions involving interactions with the metabolite. The reactions for gene expression, feedback loops and transport are typically described by chemical master equations (CME). Sampling the CME using the stochastic simulation algorithm (SSA) results in large computational costs as each reaction event is evaluated explicitly. To improve the computational efficiency of cell simulations involving high particle number systems, the authors have implemented a hybrid stochastic–deterministic (CME–ODE) method. As a test of the implementation, the authors apply the hybrid CME-ODE method to the galactose switch in Saccharomyces cerevisiae, gaining a 10–50× speedup while yielding protein distributions and species traces similar to the pure SSA CME.

1 Introduction

Many processes within living cells, especially gene expression, have characteristics of low particle numbers and a high degree of randomness that leads to stochastic effects, such as heterogeneity in a population of cells [1–4]. The chemical master equation (CME) is a useful formalism for describing the dynamics of stochastic events in biological systems. It describes a chemical process as a continuous-time Markov chain on a state space comprising particle numbers of each chemical species; thus, capturing the discreteness of particles and the random nature of individual chemical reactions. The widely used stochastic simulation algorithm (SSA) of Gillespie [5, 6] provides an effective method for obtaining unbiased realisations of these Markov processes. However, the SSA is limited by the fact that reaction events are accounted for explicitly. Systems with high particle counts – those containing metabolites in millimolar concentrations – or those with large reaction rate constants, have a high propensity (probability per unit time) for a reaction event to occur; thus, they evolve on fast time scales and incur large computational cost (e.g. time to solution).

To overcome the computational expense of solving the SSA for high concentration systems, researchers have devised hybrid approaches that decrease time-to-solution for stochastic simulations while faithfully capturing the results of stochasticity in important chemical species (e.g. transcription factors). For a brief review of methods that improve computational efficiency by reformulating the original SSA scheme in a more economical manner, see Jahnke and Kreim [7]. Notably, Cao et al. [8] describe a method by which the chemical system of interest is separated into a set of reactions with fast rates to be simulated deterministically and a set of slow reactions to be simulated stochastically. Alfonsi et al. [9] present a hybrid model in which a CME Markov jump process describing the dynamics of the species with low particle number is coupled to ordinary differential equations (ODEs) representing the highly abundant species. Jahnke and Kreim [7] validated this technique through a rigorous error analysis of the CME–ODE partitioning which was compared to a CME treatment for a small reaction system. Pájaro and Alonso [10] examined the applicability of approximate methods to modelling genetic circuits, and Smith et al. [11] showed the applicability of hybrid methods to metabolic networks. Alternative frameworks exist, for example coupling CME with the chemical Langevin equation [12]. Algorithms to handle stochastic reaction-diffusion processes typically partition the system into small spatial subvolumes and use the SSA to describe the reaction events within them [1, 2, 13]. The hybrid CME–ODE method discussed in this work will also accelerate spatially resolved simulations of processes described by reaction-diffusion master equations (RDME) over such subvolumes, where reactions within each compartment are treated by the CME.

A challenging and typical scenario arises when species participating in slow reactions are also changed by firings of one or more fast reactions, making the dynamics of the slow reactions dependent on the fast reactions. While it is tempting to assume a partial thermodynamic limit for the fast reactions involving a large number of species and simply rescale the rate constants so the entire system can be treated stochastically, this assumption cannot be made for genetic switches involving nutrients. During the early phase of sugar/inducer/metabolite uptake, the system evolves quickly and errors in the approximation can compound leading to incorrect switching dynamics and even incorrect switched steady states. The study of nutrient-induced metabolic switches is of particular importance in understanding survival fitness. As a cell adapts to a change in the composition of its environment, genes...
that will help in the processing and efficient usage of the new metabolite are upregulated, and genes that are no longer needed are downregulated, in an attempt to optimise its fitness.

2 Test system

We simulate the well-studied galactose switch in *Saccharomyces cerevisiae* [14–16]. This system, composed of 37 species and 75 reactions, is summarised in Fig. 1 and provided in Appendix – Section 10.1. The switch has four feedback loops that respond to the presence of the sugar galactose. The transcription factor G80, in dimer form (G80d), binds to the promoters of the genes for several of the proteins (G1, G2, G3 and G80) involved in the galactose switch process, and represses their expression. When galactose binds to G3 it creates an activated complex (G3i) that can bind to G80Cd (G80d in the cytoplasm). The G3i sequesters G80d in the cytoplasm causing the genes to be in an unpressed, active state. The proteins G2, which transports galactose into the cell, and G1, which metabolises galactose within the cell, also play key roles in the cell's response to changes in the galactose concentration in its environment.

The positive (G1, G2, G3) and negative (G80) feedback loops of the system work in the following manner. When exposed to a galactose rich environment the cell begins to take up sugar which subsequently sequesters the transcription factor G80 in the cytoplasm, allowing the expression of the genes for G1, G2, G3 and G80. The associated mRNAs diffuse out of the nucleus into the cytoplasm where they can be translated into their protein products. As G2 transporter proteins accumulate, they subsequently motivate a flood of more galactose into the cell. As a counter-balance, more G4d transporter proteins accumulate, they subsequently motivate a flood of more galactose into the cell, and G1, which metabolises galactose within the cell, also play key roles in the cell's response to changes in the galactose concentration in its environment.

3 Hybrid CME–ODE algorithm description

The hybrid algorithm is implemented such that the states treated by the CME and ODE solvers (the communication time τ) are exchanged between the CME and ODE solvers (the communication time τ) at intervals set by the user (e.g. the communication time τ, comparing results and run-times to the exact SSA-ODE algorithm where communication occurs between the stochastic and deterministic simulation regimes at every CME reaction event. This hybrid method utilises the lattice microbe (LM) software suite [1], and is implemented using its python interface pyLM [4]. This study showcases the hybrid implementation as a new feature of LM that allows users to easily define and simulate complex biochemical systems and to achieve computational performance that was previously unavailable in LM for systems with chemical species present in millimolar concentrations.
into four binding or unbinding reactions for each gene. These reactions consist of: (i) binding of G4 dimer to the gene, (ii) unbinding of G4 dimer from the gene, (iii) binding of the G80 dimer to the G4 dimer when it is bound to the gene and (iv) unbinding of the G80 dimer:G4 dimer from the gene (see Appendix – Section 10.3.2). These modifications allow the model (hereafter referred to as the ‘reduced model’) to be directly implemented in a spatially-resolved stochastic reaction-diffusion framework (RDME) where the assumptions underlying Hill and Michaelis-Menten reactions do not always hold [17].

Rate constants for the newly simplified reactions were fit to recapitulate the cooperative behaviour of the more complex rate equations. ODEs of the original and reduced model were simulated for 750 min and used to fit the rate constants. Briefly, during the fitting procedure the sum of squared differences between concentrations of each species in the original and reduced solutions was minimised. The reduced model generally reproduced the results of the original model <5% error.

5 Agreement with pure SSA CME

The communication times between the stochastic and deterministic descriptions as well as the timesteps for each solver must be evaluated to understand under which conditions the hybrid method is appropriate. In this section, we provide insight into the effect the choice of \( \tau \) has on the behaviour of different chemical species at varying concentrations of extracellular galactose. The hybrid CME–ODE simulation algorithm effectively captures the stochastic dynamics of the genetic switch process at galactose concentrations ranging from micromolar to millimolar. At a small enough communication timestep, both protein distributions and average traces of key species approach pure SSA CME simulation results, while allowing for sufficient performance enhancement to make the method worthwhile. It is important to note, however, that even in the limit in which the communication timestep approaches zero the results from our hybrid implementation should not converge to the pure CME results, but rather to an exact hybrid SSA-ODE model, where every time a reaction fires in the CME the ODE solver is called with updated species counts due to the CME reaction.

Significant error arises in the protein distributions of the unbound G2 transporter (hereafter referred to as G2) when large communication timesteps are used (Fig. 2a). As an example, consider a simulation with an extracellular galactose concentration of 0.055 mM where a 5 min timestep was chosen. This results in an overestimation of the mean and variability in the G2 count. As G2 is affected by reactions in both the CME (gene expression) and ODE (transport), and plays a key role in bringing galactose into the cell, error in the average or the noise in this species could give rise to differences in the switching dynamics or steady-state copy number distributions. This underscores the importance of choosing an appropriate communication timestep between the simulation regimes. As \( \tau \) is decreased from 5 min to 1 s, the protein distribution begins to closely match the pure SSA CME result, with runtime only increasing from \( \sim 25 \) to 45 min in the worst case.

The optimal communication timestep is dependent on the concentration of the external galactose. At an extracellular galactose concentration of 0.055 mM, a 1 min timestep seems to closely approach the CME distribution results (Fig. 2a). However, as demonstrated by the G2 average at 2.0 mM galactose (Fig. 2b), the 1 min timestep is on the order of a few 1000 proteins away from the pure CME result as opposed to hundreds of proteins when using a 1 s timestep at an increase of simulation time from 45 min to \( \sim 1 \) h. We have observed the trend that a smaller communication timestep is often needed to capture stochastic behaviour at higher inducer concentrations. The exact SSA-ODE trace (small dashes in Fig. 2b), which was calculated using 250 replicates, can be used to determine whether moving to a smaller communication timestep will provide an increase in accuracy since smaller timesteps will converge to this result.

The dynamics of a chemical species can also be observed by witnessing the times taken to reach steady-state values. At an extracellular galactose concentration of 2 mM the CME–ODE
The KS test calculates the maximum difference in cumulative statistic for protein distributions from each method was computed. This is an open access article published by the IET under the Creative Commons Attribution-NonCommercial-NoDerivs License

## 6 Computational performance

Having established the fidelity of the hybrid method we now demonstrate the dramatic increase in simulation efficiency it provides. Wall-time required to simulate 750 min of the galactose switch using the hybrid method, exact SSA–ODE and the pure SSA CME are shown in Table 1 along with the relative speedup.

While the pure SSA CME Gillespie direct method takes ∼2 days to simulate a cell introduced to 2 mM external galactose, the hybrid method using \( \tau = 1 \) s runs in <2 h. Even at the lowest concentration, 0.055 mM external galactose, simulations are executed in 25–45 min instead of 120–130 min. This speedup provides researchers with much higher throughput and can assist informing the design of more complex investigations.

The differences in the simulation times achieved when using a 1 min communication timestep versus a 1 s timestep at 2 mM external galactose demonstrate that at times the user must make a choice between accuracy (see Fig. 2a and d) and simulation speedup. While at the lower galactose concentration a 1 min communication interval appears sufficient, the G2 traces and KS statistic at 2 mM galactose (Fig. 2d) show that a 1 s timestep is more appropriate to capture the dynamics of the transporter species. For those considering even coarser timesteps, the run-time for a simulation using a 5 min communication timestep versus a 1 s timestep at 2 mM external galactose is due to the fact that makes orders of magnitude more computationally costly calls to Python to run the ODE Solver than what is made with a fixed timestep.

While the exact SSA-ODE method is an effective tool to determine the accuracy of hybrid methods and the appropriate communication timestep to use, it is much less computationally efficient than comparable fixed timestep runs taking ∼2 days to run at 2 mM external galactose compared to <2 h for \( \tau = 1 \) s. At the lower concentration of 0.055 mM galactose the exact method is still 5–10 times slower than when using 10–1 s timesteps. The increase in simulation time for the exact method is due to the fact that makes orders of magnitude more computationally costly calls to Python to run the ODE Solver than what is made with a fixed timestep. The exact method calls the ODE Solver at every CME reaction and we have observed the time between CME reactions to be on the order of micro to nano seconds at 2 mM external galactose (much smaller than a typical \( \tau = 1 \) s).

## 7 Conclusion

The hybrid CME–ODE algorithm implementation described in this study, now compatible with LM/pyLM, provides an effective method for the simulation of a genetic switch system containing 37 species, 75 reactions, 4 feedback loops and millions of metabolism two samples (i.e. pure CME and hybrid CME–ODE protein distributions). Therefore, identical distributions give a KS statistic of 0.00 and completely non-intersecting distributions would give a value of 1.00. The formula for KS statistic is

\[
KS = \sup_{x \in \mathbb{R}} |F(x)_{\text{CME}} - F(x)_{\text{Hybrid}}|
\]

where sup, is the supremum of the set of distances, \( F(x)_{\text{CME}} \) is the pure CME protein empirical cumulative distribution function and \( F(x)_{\text{Hybrid}} \) is the hybrid CME–ODE protein empirical cumulative distribution function.

For species with reactions in the deterministic regime (i.e. G2) a decrease in communication timestep coincides with a decrease in KS statistic (increase in agreement between protein distributions) although the observed decrease is larger at the higher external galactose concentration of 2 mM as opposed to 0.055 mM (Fig. 2d). The p-values associated with the KS test also decrease from 2 × 10⁻³ to 5 × 10⁻⁷ as timestep is decreased from 1 min to 1 s, showing greater agreement at smaller timesteps. However, for species that participate only in reactions in the stochastic regime such as the reporter protein, KS statistic is relatively constant with respect to communication timestep.

### Table 1

| Model               | Galactose, mM | Time (s) |
|---------------------|---------------|----------|
| CME                 | 0.055         | 47.4     |
| exact SSA-ODE       | 4.7           | 47.9     |
| hybrid (\( \tau = 10 \) s) | 0.4          | 1.1      |
| hybrid (\( \tau = 1 \) s)   | 0.8          | 1.8      |

\( ^a \)Times are presented in the number of hours required to simulate 750 min of cell growth.

\( ^b \)Values in parenthesis indicate the speedup relative to pure CME.

The hybrid method using \( \tau = 1 \) s takes similar times to reach steady-state concentration as the pure CME (Fig. 2c). For G2 the mean and median times to reach 80% of the average steady-state value were <10 min faster for the CME–ODE than the CME. While the hybrid method did seem to evolve slightly faster than the CME for some of the species with reactions handled by the ODE solver the difference is relatively small (generally on the order of 5–10 min faster and in the worst case ∼20 min faster).

As a quantitative measure of disagreement between the hybrid method and the pure CME, the Kolmogorov–Smirnov (KS) statistic for protein distributions from each method was computed. The KS test calculates the maximum difference in cumulative probability between empirical cumulative distribution functions of

\[
\text{Data: CME} = \text{set of CME Reactions} \\
\text{ODE} = \text{set of ODE Reactions} \\
\text{Species (CME)} = \text{set of CME Species} \\
\text{Species (ODE)} = \text{set of ODE Species} \\
x_{\text{CME}} = \text{initial CME species counts} \\
x_{\text{ODE}} = \text{initial ODE species counts} \\
R_{\text{update}} = \text{Reactions updated in CME} \\
\tau = \text{communication timestep} \\
T = \text{target simulation length} \\
t_{\text{now}} \leftarrow 0 \\
t_{\text{comm}} = 0 \\
\text{while } t_{\text{now}} \leq T \text{ do} \\
\text{while } t_{\text{now}} < t_{\text{comm}} + \tau \text{ do} \\
[\text{ Gillespie Direct SSA Step}] \\
\Delta x, \Delta t \leftarrow \text{CME}(x_{\text{CME}}, \tau) \\
t_{\text{now}} \leftarrow t_{\text{now}} + \Delta t \\
x_{\text{CME}} \leftarrow x_{\text{CME}} + \Delta x \\
\text{end} \\
\text{end} \\
\text{ Update ODE particle counts} \\
\text{for } s \in \text{ODE do} \\
x_{\text{ODE}}(s) \leftarrow x_{\text{CME}}(s) \\
\text{end} \\
\text{ Run LSODA forward for } \tau \\
x_{\text{ODE}} \leftarrow \text{ODE Solve}(R_{\text{ODE}}, x_{\text{ODE}}, \tau) \\
\text{ Update CME particle counts} \\
\text{for } s \in \text{ODE do} \\
x_{\text{CME}}(s) \leftarrow \text{round}(x_{\text{ODE}}(s)) \\
\text{end} \\
\text{ Update CME reaction rates} \\
\text{for } \tau \in R_{\text{update}} \text{ do} \\
\text{ Update reaction rate for } R_{\text{CME}}(\tau) \text{ based on dependent species in } x_{\text{ODE}} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end}
particles within a eukaryotic cell. The 10–50-fold computational performance increase relative to a pure SSA CME simulation for sugar concentrations ranging from micromolar to millimolar makes this simulation method an intriguing option for researchers in the field of computational biology. However, the user must determine a suitable communication timestep between the stochastic and deterministic regimes to ensure that data is passed with enough frequency to maintain the protein distributions and stochastic effects that are observed when using a pure SSA CME implementation. The results gained from these efficient hybrid CME–ODE simulations can be used to inform simulation setup conditions (communication timestep etc.) for hybrid simulations of much more computationally expensive, spatially resolved whole cell RDME studies. Hybrid simulations can utilise LM features developed for RDME simulations, such as multi-GPU computation [2] and optimised propensity calculation [18], without any further work on the user’s part. LM provides the fastest method for RDME simulations to date and with the addition of this hybrid stochastic–deterministic method, simulations of systems with chemical species present in millimolar concentrations (such as those found in nutrient based genetic switches) are now computationally accessible.

8 Acknowledgments

D.M.B. thanks Piyush Labsetwar for helping in simplifying the galactose switch system model. J.R.P. thanks support from the National Science Foundation (NSF) Graduate Fellowship Program under grant DGE-1144245. This research was supported in part by NSF grants: PoLS (PHY-1505008, PHY-1026550), MCB-1244570 and CPLC (PHY-1430124) as well as NIH grants NIH-9P41GM10460123 and by NASA grant no. NNA13AA91A. This research is part of the Blue Waters sustained-petascale computing project at NCSA and the University of Illinois Urbana-Champaign and is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois.

9 References

[1] Roberts, E., Stone, J. E., Luthey-Schulten, Z.: ‘Lattice microbes: High-performance stochastic simulation method for the reaction-diffusion master equation’, J. Comput. Chem., 2013, 34, (3), pp. 245–255
[2] Hallock, M. J., Stone, J. E., Roberts, E., et al.: ‘Simulation of reaction diffusion processes over biologically relevant size and time scales using multi-GPU workstations’, Parallel Comput., 2014, 40, (5–6), pp. 86–99
[3] Stone, J. E., Hallock, M. J., Phillips, J. C., et al.: ‘Evaluation of emerging energy-efficient heterogeneous computing platforms for biomolecular and cellular simulation workloads’, 2016 IEEE Int. Parallel and Distributed Processing Symp. Workshops (IPDPSW), Chicago, IL, USA, May 2016
[4] Peterson, J. R., Hallock, M. J., Cole, J. A., et al.: ‘A problem solving environment for stochastic biological simulations’. PyHPC’13: Proc. of the 3rd Workshop on Python for High-Performance and Scientific Computing, 2013
[5] Gillespie, D. T.: ‘Exact stochastic simulation of coupled chemical reactions’, J. Phys. Chem., 1977, 81, (25), pp. 2340–2360
[6] Gillespie, D. T.: ‘Approximate accelerated stochastic simulation of chemically reacting systems’, J. Chem. Phys., 2001, 115, (4), pp. 1716–1733
[7] Jahnke, T., Kreim, M.: ‘Error bound for piecewise deterministic modeling stochastic reaction systems’, Multiscale Model. Simul., 2012, 10, (4), pp. 1119–1147
[8] Cao, Y., Gillespie, D. T., Petzold, L. R.: ‘The slow-scale stochastic simulation algorithm’, J. Chem. Phys., 2005, 122, (1), pp. 014116
[9] Alfonsi, A., Cancès, E., Turinici, G., et al.: ‘Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems’, ESAIM: Proc., 2005, 14, pp. 1–13
[10] Pijara, M., Alonso, A.: ‘On the applicability of deterministic approximations to model genetic circuits’, IFAC-PapersOnLine, 2016, 49, (7), pp. 206–211
[11] Smith, S., Cianci, C., Grima, R.: ‘Model reduction for stochastic chemical species present in millimolar concentrations (such as those found in nutrient based genetic switches) are now computationally accessible.

10 Appendix

10.1 Galactose switch model

The Python PyLM model for the galactose switch system as well as the code for using the hybrid algorithm through the ‘hookSimulation’ feature of LM is available at http://www.scs.illinois.edu/schulten/software/ODE_CME.tar.gz. The reaction model is included below for completeness. All rates are stochastic rates (i.e. the volume of the cell has been factored into reaction rate) and are presented in min⁻¹ and molecules⁻¹ min⁻¹.

10.2 Species

DG1
DG1_G4d
DG1_G4d_G80d
DG2
DG2_G4d
DG2_G4d_G80d
DG3
DG3_G4d
DG3_G4d_G80d
DG80
DG80_G4d
DG80_G4d_G80d
DGrep
DGrep_G4d
DGrep_G4d_G80d
reporter_mRNA

DG1 gene encoding Gal1 with nothing bound
DG1 gene encoding Gal1 bound to G4 dimer
DG2 gene encoding Gal2 with nothing bound
DG2 gene encoding Gal2 bound to G4 dimer
DG3 gene encoding Gal3 with nothing bound
DG3 gene encoding Gal3 bound to G4 dimer
DG80 gene encoding Gal80 with nothing bound
DG80 gene encoding Gal80 bound to G4 dimer
DG80 gene encoding Gal80 bound to the Gal4 dimer and Gal80 dimer
DGrep gene encoding the reporter protein (YFP) with nothing bound
DGrep gene encoding reporter protein bound to G4 dimer
DGrep gene encoding reporter bound to the Gal4 dimer and Gal80 dimer

R1 mRNA for Gal1
R2 mRNA for Gal2
R3 mRNA for Gal3
R4 mRNA for Gal4
R80 mRNA for Gal80
reporter_mRNA mRNA for the reporter gene

G1 Gal1; galactokinase that metabolises galactose
G2 Gal2; galactose transporter
G3 Gal3; galactose sensing transcription factor
G3i Gal3 bound to a galactose molecule
G4 Gal4; a monomer of the Gal4 transcriptional repressor
G4d Gal4 dimer; the transcriptional repressor dimer in the nucleus
G80 Gal80; nuclear; the monomer of the transcriptional repressor
G80C Gal80; cytoplasmic; the monomer of the transcriptional repressor in the cytoplasm
G80d Gal80 dimer; nuclear; a dimer of the transcriptional repressor
G80Cd Gal80 dimer; cytoplasmic; a dimer of the transcriptional repressor in the cytoplasm
G80G3i Gal80 dimer bound to Gal3i; the transcriptional repressor sequestered in the cytoplasm
G4I intracellular galactose
G4AE extracellular galactose
G2GAI galactose bound to the Gal2 transporter on the intracellular side
G2GAE galactose bound to the Gal2 transporter on the extracellular side
G1GAI galactose bound to the Gal2 transporter on the extracellular side
reporter a yellow fluorescence reporter protein (YFP)

10.3 Reaction model

10.3.1 Transcription:

\[
\begin{align*}
D_G1\_G4d & \xrightarrow{k_1} R_1 + D_G1\_G4d \\
R_1 & \xrightarrow{k_2} \emptyset \\
D_G2\_G4d & \xrightarrow{k_3} R_2 + D_G2\_G4d \\
R_2 & \xrightarrow{k_4} \emptyset \\
D_G3\_G4d & \xrightarrow{k_5} R_3 + D_G3\_G4d \\
R_3 & \xrightarrow{k_6} \emptyset \\
D_Grep\_G4d & \xrightarrow{k_7} \text{reporter}\_r\_na + D_Grep\_G4d \\
\emptyset & \xrightarrow{k_8} \emptyset \\
D_G80\_G4d & \xrightarrow{k_9} R_80 + D_G80\_G4d \\
R_80 & \xrightarrow{k_10} \emptyset
\end{align*}
\]

10.3.2 DNA regulation:

\[
\begin{align*}
D_G1 \_G4d & \xrightarrow{k_1} D_G1 \_G4d \\
D_G1 \_G4d & \xrightarrow{k_2} D_G1 \_G4d \\
D_G1 \_G4d \_G80d & \xrightarrow{k_3} D_G1 \_G4d \_G80d \\
D_G1 \_G4d \_G80d & \xrightarrow{k_4} D_G1 \_G4d \_G80d \\
D_G2 \_G4d & \xrightarrow{k_5} D_G2 \_G4d \\
D_G2 \_G4d \_G80d & \xrightarrow{k_6} D_G2 \_G4d \_G80d \\
D_G2 \_G4d \_G80d & \xrightarrow{k_7} D_G2 \_G4d \_G80d \\
D_G3 \_G4d & \xrightarrow{k_8} D_G3 \_G4d \\
D_G3 \_G4d \_G80d & \xrightarrow{k_9} D_G3 \_G4d \_G80d \\
D_G3 \_G4d \_G80d & \xrightarrow{k_10} D_G3 \_G4d \_G80d \\
D_G3 \_G4d \_G80d & \xrightarrow{k_11} D_G3 \_G4d \_G80d \\
D_G3 \_G4d \_G80d & \xrightarrow{k_12} D_G3 \_G4d \_G80d \\
D_G80 \_G4d & \xrightarrow{k_13} D_G80 \_G4d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_14} D_G80 \_G4d \_G80d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_15} D_G80 \_G4d \_G80d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_16} D_G80 \_G4d \_G80d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_17} D_G80 \_G4d \_G80d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_18} D_G80 \_G4d \_G80d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_19} D_G80 \_G4d \_G80d \\
D_G80 \_G4d \_G80d & \xrightarrow{k_20} D_G80 \_G4d \_G80d
\end{align*}
\]

10.3.3 Translation:

\[
\begin{align*}
R_1 & \xrightarrow{k_1} G_1 + R_1 \\
G_1 & \xrightarrow{k_2} \emptyset \\
R_2 & \xrightarrow{k_3} G_2 + R_2 \\
G_2 & \xrightarrow{k_4} \emptyset \\
R_3 & \xrightarrow{k_5} G_3 + R_3 \\
G_3 & \xrightarrow{k_6} \emptyset \\
R_4 & \xrightarrow{k_7} G_4 + R_4 \\
G_4 & \xrightarrow{k_8} \emptyset \\
\text{reporter}\_r\_na & \xrightarrow{k_9} \text{reporter} + \text{reporter}\_r\_na \\
\emptyset & \xrightarrow{k_10} \emptyset \\
R_80 & \xrightarrow{k_11} G_80 + R_80 \\
G_80 & \xrightarrow{k_12} \emptyset
\end{align*}
\]

10.3.4 Transport and enzymatic:

\[
\begin{align*}
G_2GAI & \xrightarrow{k_1} G_2GAE \\
G_2GAE & \xrightarrow{k_2} G_2GAI \\
G_2GAE & \xrightarrow{k_3} G_2 \\
G_2 & \xrightarrow{k_4} \emptyset \\
G_2GAE & \xrightarrow{k_5} \emptyset \\
G_2 & \xrightarrow{k_6} \emptyset \\
G_2GAI & \xrightarrow{k_7} G_2GAI
\end{align*}
\]
10.3.5 Transcription factor:

\[ G3 + GAI \quad k_1 = 7.45 \times 10^{-7} \quad \rightarrow \quad G3i \]

\[ G3i \quad k_2 = 890 \quad \rightarrow \quad G3 + GAI \]

\[ G3i \quad k_3 = 0.01155 \quad \rightarrow \quad GAI \]

\[ G80G3i \quad k_4 = 0.0159616 \quad \rightarrow \quad G3i + G80Cd \]

\[ G3i + G80Cd \quad k_5 = 0.025716 \quad \rightarrow \quad G80G3i \]

\[ G80G3i \quad k_6 = 0.005775 \quad \rightarrow \quad \emptyset \]

\[ G4 + G4 \quad k_7 = 100 \quad \rightarrow \quad G4d \]

\[ G4d \quad k_8 = 0.001 \quad \rightarrow \quad G4 + G4 \]

\[ G4d \quad k_9 = 0.0069315 \quad \rightarrow \quad \emptyset \]

\[ G80 + G80 \quad k_{10} = 100 \quad \rightarrow \quad G80d \]

\[ G80d \quad k_{11} = 0.001 \quad \rightarrow \quad G80 + G80 \]

\[ G80d \quad k_{12} = 0.0069315 \quad \rightarrow \quad \emptyset \]

\[ G80C + G80C \quad k_{13} = 100 \quad \rightarrow \quad G80Cd \]

\[ G80Cd \quad k_{14} = 0.001 \quad \rightarrow \quad G80C + G80C \]

\[ G80Cd \quad k_{15} = 0.0069315 \quad \rightarrow \quad \emptyset \]

\[ G80Cd \quad k_{16} = 50 \quad \rightarrow \quad G80d \]

\[ G80d \quad k_{17} = 50 \quad \rightarrow \quad G80Cd \]

\[ G80Cd \quad k_{18} = 50 \quad \rightarrow \quad G80C \]

\[ G80C \quad k_{19} = 50 \quad \rightarrow \quad G80 \]