

Lecture 21. Application in Chemical Reaction Kinetics

Tiejun Li^{1,2}

¹School of Mathematical Sciences (SMS),
&
²Center for Machine Learning Research (CMLR),
Peking University,
Beijing 100871,
P.R. China
tieli@pku.edu.cn

Office: No. 1 Science Building, Room 1376E

Table of Contents

Mathematical Setup

Stochastic Simulation Algorithm (SSA)

Tau-leaping Algorithm

Avoiding Negative Populations

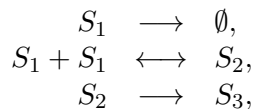
Stiff System

Convergence Analysis

Stationary Distribution

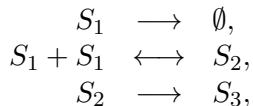
Traditional chemical reaction dynamics — ODE

- ▶ Decaying-dimerizing reaction



Traditional chemical reaction dynamics — ODE

- ▶ Decaying-dimerizing reaction



- ▶ Traditional model — ODEs for the concentration ([Law of Mass Action](#))

$$\begin{aligned} \frac{dx_1}{dt} &= -k_1x_1 - 2k_2x_1^2 + 2k_3x_2 \\ \frac{dx_2}{dt} &= k_2x_1^2 - k_3x_2 - k_4x_2 \\ \frac{dx_3}{dt} &= k_4x_2 \end{aligned}$$

k_1, k_2, k_3, k_4 are reaction rates.

Drawbacks of ODE description

- ▶ Deterministic model describes an average behavior and **is valid for large population**

Drawbacks of ODE description

- ▶ Deterministic model describes an average behavior and **is valid for large population**
- ▶ Species of small population may play important role in biological system

Drawbacks of ODE description

- ▶ Deterministic model describes an average behavior and **is valid for large population**
- ▶ Species of small population may play important role in biological system
- ▶ Examples of stochasticity
 - A. Arkin et al., Genetics 149 (1998), 1633 — Stochastic variations can produce probabilistic pathway selection.
 - M. Elowitz et al., Science 297 (2002), 391 — Gene expression is affected by both extrinsic and intrinsic noise.

Chemical kinetic system (CKS)

Taking into account the stochasticity in biological chemical reactions, this **opens a new way for modeling and simulation!**

Chemical reaction kinetics — stochastic version

- ▶ **Well-stirred** system of N molecular species $\{S_1, S_2, \dots, S_N\}$ interacting through M chemical reaction channels $\{R_1, R_2, \dots, R_M\}$.

Chemical reaction kinetics — stochastic version

- ▶ **Well-stirred** system of N molecular species $\{S_1, S_2, \dots, S_N\}$ interacting through M chemical reaction channels $\{R_1, R_2, \dots, R_M\}$.
- ▶ State of the system

$$\mathbf{X}_t = (X_t^1, X_t^2, \dots, X_t^N).$$

Chemical reaction kinetics — stochastic version

- ▶ **Well-stirred** system of N molecular species $\{S_1, S_2, \dots, S_N\}$ interacting through M chemical reaction channels $\{R_1, R_2, \dots, R_M\}$.
- ▶ State of the system

$$\mathbf{X}_t = (X_t^1, X_t^2, \dots, X_t^N).$$

- ▶ Each reaction channel R_j is characterized by its **propensity function** $a_j(\mathbf{x})$ and its **state change vector**

$$\boldsymbol{\nu}_j = (\nu_j^1, \nu_j^2, \dots, \nu_j^N).$$

Chemical reaction kinetics — stochastic version

- ▶ **Well-stirred** system of N molecular species $\{S_1, S_2, \dots, S_N\}$ interacting through M chemical reaction channels $\{R_1, R_2, \dots, R_M\}$.
- ▶ State of the system

$$\mathbf{X}_t = (X_t^1, X_t^2, \dots, X_t^N).$$

- ▶ Each reaction channel R_j is characterized by its **propensity function** $a_j(\mathbf{x})$ and its **state change vector**

$$\boldsymbol{\nu}_j = (\nu_j^1, \nu_j^2, \dots, \nu_j^N).$$

- ▶ Here $a_j(\mathbf{x})dt$ **gives the probability** that the system will experience an R_j reaction in the next infinitesimal time dt when the current state $\mathbf{X}_t = \mathbf{x}$. ν_j^i **is the change** in the number of S_i molecules caused by one R_j reaction.

Chemical reaction kinetics — stochastic version

- ▶ **Well-stirred** system of N molecular species $\{S_1, S_2, \dots, S_N\}$ interacting through M chemical reaction channels $\{R_1, R_2, \dots, R_M\}$.
- ▶ State of the system

$$\mathbf{X}_t = (X_t^1, X_t^2, \dots, X_t^N).$$

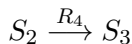
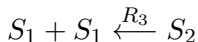
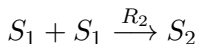
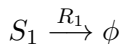
- ▶ Each reaction channel R_j is characterized by its **propensity function** $a_j(\mathbf{x})$ and its **state change vector**

$$\boldsymbol{\nu}_j = (\nu_j^1, \nu_j^2, \dots, \nu_j^N).$$

- ▶ Here $a_j(\mathbf{x})dt$ **gives the probability** that the system will experience an R_j reaction in the next infinitesimal time dt when the current state $\mathbf{X}_t = \mathbf{x}$. ν_j^i **is the change** in the number of S_i molecules caused by one R_j reaction.
- ▶ We will define $a_0(\mathbf{x}) = \sum_{j=1}^M a_j(\mathbf{x})$.

An example

- ▶ Decaying-dimerizing reaction:



The **propensity functions** are given by

$$a_1 = x_1, \quad a_2 = 5x_1(x_1 - 1), \quad a_3 = 1000x_2, \quad a_4 = 0.1x_2$$

and **state change vector**

$$\nu_1 = (-1, 0, 0), \quad \nu_2 = (-2, 1, 0), \quad \nu_3 = (2, -1, 0), \quad \nu_4 = (-1, 0, 1).$$

Initial state $\mathbf{X}(0) = (400, 798, 0)$.

Chemical Master Equation

- ▶ The Chapman-Kolmogorov equation in the time interval $[t, t + dt)$

$$P(\mathbf{x}, t + dt | \mathbf{x}_0, t_0) = \sum_{j=1}^M P(\mathbf{x} - \boldsymbol{\nu}_j, t | \mathbf{x}_0, t_0) a_j(\mathbf{x} - \boldsymbol{\nu}_j) dt + \left(1 - \sum_{j=1}^M a_j(\mathbf{x}) dt\right) P(\mathbf{x}, t | \mathbf{x}_0, t_0),$$

where dt is an infinitesimal time, and we have already omitted the higher order terms in $o(dt)$.

Chemical Master Equation

- ▶ The Chapman-Kolmogorov equation in the time interval $[t, t + dt)$

$$P(\mathbf{x}, t + dt | \mathbf{x}_0, t_0) = \sum_{j=1}^M P(\mathbf{x} - \boldsymbol{\nu}_j, t | \mathbf{x}_0, t_0) a_j(\mathbf{x} - \boldsymbol{\nu}_j) dt + \left(1 - \sum_{j=1}^M a_j(\mathbf{x}) dt\right) P(\mathbf{x}, t | \mathbf{x}_0, t_0),$$

where dt is an infinitesimal time, and we have already omitted the higher order terms in $o(dt)$.

- ▶ With some algebra, we get the well-known *chemical master equation*.

$$\partial_t P(\mathbf{x}, t | \mathbf{x}_0, t_0) = \sum_{j=1}^M a_j(\mathbf{x} - \boldsymbol{\nu}_j) P(\mathbf{x} - \boldsymbol{\nu}_j, t | \mathbf{x}_0, t_0) - a_0(\mathbf{x}) P(\mathbf{x}, t | \mathbf{x}_0, t_0)$$

where $a_0(\mathbf{x}) := \sum_{j=1}^M a_j(\mathbf{x})$ is the total propensity.

Table of Contents

Mathematical Setup

Stochastic Simulation Algorithm (SSA)

Tau-leaping Algorithm

Avoiding Negative Populations

Stiff System

Convergence Analysis

Stationary Distribution

SSA — Gillespie's algorithm

- ▶ SSA (Stochastic Simulation Algorithm) (Gillespie, JCP 22 (1976), 403.)

SSA — Gillespie's algorithm

- ▶ SSA (Stochastic Simulation Algorithm) (Gillespie, JCP 22 (1976), 403.)
 - ▶ Step 1: Sampling the **waiting time** τ as an exponentially distributed random variable (R.V.) with rate $a_0(\mathbf{X}_t)$;

SSA — Gillespie's algorithm

- ▶ SSA (Stochastic Simulation Algorithm) (Gillespie, JCP 22 (1976), 403.)
 - ▶ Step 1: Sampling the **waiting time** τ as an exponentially distributed random variable (R.V.) with rate $a_0(\mathbf{X}_t)$;
 - ▶ Step 2: Sampling an **M point R.V. k** with probability $\frac{a_j(\mathbf{X}_t)}{a_0(\mathbf{X}_t)}$ for the j -th reaction;

SSA — Gillespie's algorithm

- ▶ SSA (Stochastic Simulation Algorithm) (Gillespie, JCP 22 (1976), 403.)
 - ▶ Step 1: Sampling the **waiting time** τ as an exponentially distributed random variable (R.V.) with rate $a_0(\mathbf{X}_t)$;
 - ▶ Step 2: Sampling an **M point R.V.** k with probability $\frac{a_j(\mathbf{X}_t)}{a_0(\mathbf{X}_t)}$ for the j -th reaction;
 - ▶ Step 3: **Update** $\mathbf{X}_{t+\tau} = \mathbf{X}_t + \nu_k$, then return to Step 1.

SSA — Gillespie's algorithm

- ▶ SSA (Stochastic Simulation Algorithm) (Gillespie, JCP 22 (1976), 403.)
 - ▶ Step 1: Sampling the **waiting time** τ as an exponentially distributed random variable (R.V.) with rate $a_0(\mathbf{X}_t)$;
 - ▶ Step 2: Sampling an **M point R.V.** k with probability $\frac{a_j(\mathbf{X}_t)}{a_0(\mathbf{X}_t)}$ for the j -th reaction;
 - ▶ Step 3: **Update** $\mathbf{X}_{t+\tau} = \mathbf{X}_t + \nu_k$, then return to Step 1.
- ▶ Understood as an exact simulation since the waiting time of the j -th reaction firing is distributed as $a_j(\mathbf{x}) \exp(-a_0(\mathbf{x})t)$.

SSA — Gillespie's algorithm

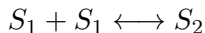
- ▶ SSA (Stochastic Simulation Algorithm) (Gillespie, JCP 22 (1976), 403.)
 - ▶ Step 1: Sampling the **waiting time** τ as an exponentially distributed random variable (R.V.) with rate $a_0(\mathbf{X}_t)$;
 - ▶ Step 2: Sampling an **M point R.V.** k with probability $\frac{a_j(\mathbf{X}_t)}{a_0(\mathbf{X}_t)}$ for the j -th reaction;
 - ▶ Step 3: **Update** $\mathbf{X}_{t+\tau} = \mathbf{X}_t + \nu_k$, then return to Step 1.
- ▶ Understood as an exact simulation since the waiting time of the j -th reaction firing is distributed as $a_j(\mathbf{x}) \exp(-a_0(\mathbf{x})t)$.
- ▶ It is essentially the BKL algorithm (Bortz-Kalos-Lebowitz) or KMC in condensed matter physics.

Shortcomings of SSA

- ▶ When the **population of molecules is very large**, the reaction will fire very frequently, which is quite time consuming.

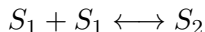
Shortcomings of SSA

- ▶ When the **population of molecules is very large**, the reaction will fire very frequently, which is quite time consuming.
- ▶ When the **reaction rate is very large for a reversible reaction**, the reactions will fire back and forth very frequently, but cause very little change of the state.



Shortcomings of SSA

- ▶ When the **population of molecules is very large**, the reaction will fire very frequently, which is quite time consuming.
- ▶ When the **reaction rate is very large for a reversible reaction**, the reactions will fire back and forth very frequently, but cause very little change of the state.



- ▶ How to accelerate the simulation process?

Table of Contents

Mathematical Setup

Stochastic Simulation Algorithm (SSA)

Tau-leaping Algorithm

Avoiding Negative Populations

Stiff System

Convergence Analysis

Stationary Distribution

Tau-leaping Algorithm

- ▶ The philosophy is to accelerate SSA at the cost of accuracy.

Tau-leaping Algorithm

- ▶ The philosophy is to accelerate SSA at the cost of accuracy.
- ▶ Gillespie proposed the following condition for accelerating the simulation:

“*Leap Condition*: Require the leap time τ to be small enough that the change in the state during $[t, t + \tau)$ will be so slight that no propensity function will suffer an appreciable (i.e., macroscopically noninfinitesimal) change in its value.”

Tau-leaping Algorithm

- ▶ The philosophy is to accelerate SSA at the cost of accuracy.
- ▶ Gillespie proposed the following condition for accelerating the simulation:
 “*Leap Condition*: Require the leap time τ to be small enough that the change in the state during $[t, t + \tau)$ will be so slight that no propensity function will suffer an appreciable (i.e., macroscopically noninfinitesimal) change in its value.”
- ▶ This means that we set $a_j(\mathbf{X}_t)$ fixed, and leap with time stepsize δt .

Tau-leaping Algorithm

- ▶ The philosophy is to accelerate SSA at the cost of accuracy.
- ▶ Gillespie proposed the following condition for accelerating the simulation:
 “*Leap Condition*: Require the leap time τ to be small enough that the change in the state during $[t, t + \tau)$ will be so slight that no propensity function will suffer an appreciable (i.e., macroscopically noninfinitesimal) change in its value.”
- ▶ This means that we set $a_j(\mathbf{X}_t)$ fixed, and leap with time stepsize δt .
- ▶ Then the number of j -th reaction will be $\mathcal{P}(a_j(\mathbf{X}_t)\delta t)$, which is a Poisson random variable with distribution $\lambda^k/k! \exp(-\lambda)$. Here $\lambda = a_j(\mathbf{X}_t)\delta t$.

Tau-leaping Algorithm

- ▶ The philosophy is to accelerate SSA at the cost of accuracy.
- ▶ Gillespie proposed the following condition for accelerating the simulation:
 “*Leap Condition*: Require the leap time τ to be small enough that the change in the state during $[t, t + \tau)$ will be so slight that no propensity function will suffer an appreciable (i.e., macroscopically noninfinitesimal) change in its value.”
- ▶ This means that we set $a_j(\mathbf{X}_t)$ fixed, and leap with time stepsize δt .
- ▶ Then the number of j -th reaction will be $\mathcal{P}(a_j(\mathbf{X}_t)\delta t)$, which is a Poisson random variable with distribution $\lambda^k/k! \exp(-\lambda)$. Here $\lambda = a_j(\mathbf{X}_t)\delta t$.
- ▶ So we have the tau-leaping scheme

$$\mathbf{X}_{t+\delta t} = \mathbf{X}_t + \sum_{j=1}^M \nu_j \mathcal{P}(a_j(\mathbf{X}_t)\delta t)$$

Tau-leaping Algorithm: Stepsize Selection

- ▶ The procedure for selecting tau (time stepsize) is as follows.

Tau-leaping Algorithm: Stepsize Selection

- ▶ The procedure for selecting tau (time stepsize) is as follows.
- ▶ Note that the state after τ -leaping is

$$\mathbf{X} \rightarrow \mathbf{X} + \sum_{j=1}^M \nu_j a_j(\mathbf{X}) \tau := \mathbf{X} + \tau \boldsymbol{\xi}.$$

Tau-leaping Algorithm: Stepsize Selection

- ▶ The procedure for selecting tau (time stepsize) is as follows.
- ▶ Note that the state after τ -leaping is

$$\mathbf{X} \rightarrow \mathbf{X} + \sum_{j=1}^M \nu_j a_j(\mathbf{X}) \tau := \mathbf{X} + \tau \boldsymbol{\xi}.$$

- ▶ Then the leap condition will be

$$|a_j(\mathbf{X} + \tau \boldsymbol{\xi}) - a_j(\mathbf{X})| \leq \epsilon a_0(\mathbf{X}),$$

where $0 < \epsilon < 1$ is a specified value.

Tau-leaping Algorithm: Stepsize Selection

- ▶ The procedure for selecting tau (time stepsize) is as follows.
- ▶ Note that the state after τ -leaping is

$$\mathbf{X} \rightarrow \mathbf{X} + \sum_{j=1}^M \nu_j a_j(\mathbf{X}) \tau := \mathbf{X} + \tau \boldsymbol{\xi}.$$

- ▶ Then the leap condition will be

$$|a_j(\mathbf{X} + \tau \boldsymbol{\xi}) - a_j(\mathbf{X})| \leq \epsilon a_0(\mathbf{X}),$$

where $0 < \epsilon < 1$ is a specified value.

- ▶ The Taylor expansion of the LHS gives $\tau |\boldsymbol{\xi} \cdot \nabla a_j| \leq \epsilon a_0(\mathbf{X})$, which gives one stepsize selection strategy

$$\tau \approx \min_{j=1, \dots, M} \left\{ \frac{\epsilon a_0(\mathbf{X})}{|\boldsymbol{\xi} \cdot \nabla a_j|} \right\}.$$

Tau-leaping Algorithm: Stepsize Selection

- ▶ The procedure for selecting tau (time stepsize) is as follows.
- ▶ Note that the state after τ -leaping is

$$\mathbf{X} \rightarrow \mathbf{X} + \sum_{j=1}^M \nu_j a_j(\mathbf{X}) \tau := \mathbf{X} + \tau \boldsymbol{\xi}.$$

- ▶ Then the leap condition will be

$$|a_j(\mathbf{X} + \tau \boldsymbol{\xi}) - a_j(\mathbf{X})| \leq \epsilon a_0(\mathbf{X}),$$

where $0 < \epsilon < 1$ is a specified value.

- ▶ The Taylor expansion of the LHS gives $\tau |\boldsymbol{\xi} \cdot \nabla a_j| \leq \epsilon a_0(\mathbf{X})$, which gives one stepsize selection strategy

$$\tau \approx \min_{j=1, \dots, M} \left\{ \frac{\epsilon a_0(\mathbf{X})}{|\boldsymbol{\xi} \cdot \nabla a_j|} \right\}.$$

- ▶ Many more robust stepsize selection strategies are also proposed.

Some Remarks

Remark

The total propensity is $a_0(\mathbf{X})$. So the expected waiting time for one reaction fires is $\mathcal{O}(1/a_0(\mathbf{X}))$. If

$$\tau \leq m/a_0(\mathbf{X}), \quad m \sim \mathcal{O}(1)$$

One will use SSA instead.

Some Remarks

Remark

The total propensity is $a_0(\mathbf{X})$. So the expected waiting time for one reaction fires is $\mathcal{O}(1/a_0(\mathbf{X}))$. If

$$\tau \leq m/a_0(\mathbf{X}), \quad m \sim \mathcal{O}(1)$$

One will use SSA instead.

Remark

Compare with the forward Euler step $x_{n+1} = x_n + f(x_n)\delta t$ for ODE

$$\dot{x} = f(x).$$

We actually fix $f(x)$ as a constant $f(x_n)$ in $[t_n, t_{n+1})$ with a similar idea. One will find more connections along this direction.

Tau-leaping Algorithm: Multiscale Picture

Multi-scale picture:

The multi-scale picture from tau-leaping is charming.

- ▶ From tau-leaping to Chemical Langevin Equation:

When $a_j(\mathbf{X}_t)\tau \gg 1$, $P(a_j(\mathbf{X}_t)\tau) \approx N(a_j(\mathbf{X}_t)\tau, a_j(\mathbf{X}_t)\tau)$ by Central Limit Theorem

$$\mathbf{X}_{t+\tau} \approx \mathbf{X}_t + \sum_{j=1}^M \nu_j a_j(\mathbf{X}_t)\tau + \sum_{j=1}^M \nu_j \sqrt{a_j(\mathbf{X}_t)\tau} N(0, 1)$$

which corresponds to CLE

$$d\mathbf{X}_t = \sum_{j=1}^M \nu_j a_j(\mathbf{X}_t) dt + \sum_{j=1}^M \nu_j \sqrt{a_j(\mathbf{X}_t)} d\mathbf{W}_t$$

Tau-leaping Algorithm: Multiscale Picture

- ▶ From Chemical Langevin Equation to Reaction Rate Equation:
When $a_j(\mathbf{X}_t)\tau \rightarrow +\infty$, $N(a_j(\mathbf{X}_t)\tau, a_j(\mathbf{X}_t)\tau) \approx a_j(\mathbf{X}_t)\tau$ by Law of Large Numbers

$$\mathbf{X}_{t+\tau} \approx \mathbf{X}_t + \sum_{j=1}^M \nu_j a_j(\mathbf{X}_t)\tau$$

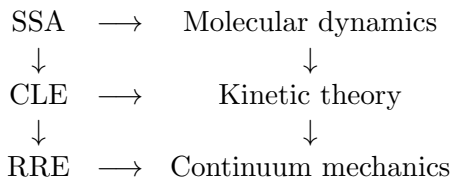
which corresponds to RRE

$$\frac{d\mathbf{X}_t}{dt} = \sum_{j=1}^M \nu_j a_j(\mathbf{X}_t)$$

Tau-leaping bridges all of the equations in different scales with a seamless way!

Multiscale Picture

The comparison with fluid mechanics (upscaling) will be instructive.



Negative Populations

- ▶ Because of unboundedness of Poisson R.V., negative populations may appear.

Negative Populations

- ▶ Because of unboundedness of Poisson R.V., negative populations may appear.
- ▶ One choice to avoid N.P. is by binomial tau-leaping.

Negative Populations

- ▶ Because of unboundedness of Poisson R.V., negative populations may appear.
- ▶ One choice to avoid N.P. is by binomial tau-leaping.
- ▶ Note that Poisson distribution may be viewed as a limit of binomial distribution $B(n, p)$ when $n \rightarrow \infty$ with $\lambda = np$ fixed. That is

$$B(k; n, p) = C_n^k p^k (1 - p)^{n-k} \rightarrow \mathcal{P}(\lambda).$$

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*
- ▶ *Step 2: Sequentially for $j = 1, 2, \dots, M$ do:*

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*
- ▶ *Step 2: Sequentially for $j = 1, 2, \dots, M$ do:*
 - ▶ *Find $k_{\max}^{(j)} \sim$ Maximal admissible number of j -th reactions according to $\tilde{\mathbf{X}}$;*

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*
- ▶ *Step 2: Sequentially for $j = 1, 2, \dots, M$ do:*
 - ▶ *Find $k_{\max}^{(j)} \sim$ Maximal admissible number of j -th reactions according to $\tilde{\mathbf{X}}$;*
 - ▶ *Define $p = a_j \tau / k_{\max}^{(j)}$;*

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*
- ▶ *Step 2: Sequentially for $j = 1, 2, \dots, M$ do:*
 - ▶ *Find $k_{\max}^{(j)} \sim$ Maximal admissible number of j -th reactions according to $\tilde{\mathbf{X}}$;*
 - ▶ *Define $p = a_j \tau / k_{\max}^{(j)}$;*
 - ▶ *Sample binomial distribution R.V. $k_j \sim B(k_{\max}^{(j)}, p)$;*

Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*
- ▶ *Step 2: Sequentially for $j = 1, 2, \dots, M$ do:*
 - ▶ *Find $k_{\max}^{(j)} \sim$ Maximal admissible number of j -th reactions according to $\tilde{\mathbf{X}}$;*
 - ▶ *Define $p = a_j \tau / k_{\max}^{(j)}$;*
 - ▶ *Sample binomial distribution R.V. $k_j \sim B(k_{\max}^{(j)}, p)$;*
 - ▶ *Firing j -th reaction k_j times:*



Binomial Tau-leaping

Algorithm (Binomial Tau-leaping)

Avoiding negative populations.

- ▶ *Step 1: At time t , set $\tilde{\mathbf{X}} = \mathbf{X}_t$, select τ ;*
- ▶ *Step 2: Sequentially for $j = 1, 2, \dots, M$ do:*
 - ▶ *Find $k_{\max}^{(j)} \sim$ Maximal admissible number of j -th reactions according to $\tilde{\mathbf{X}}$;*
 - ▶ *Define $p = a_j \tau / k_{\max}^{(j)}$;*
 - ▶ *Sample binomial distribution R.V. $k_j \sim B(k_{\max}^{(j)}, p)$;*
 - ▶ *Firing j -th reaction k_j times:*



- ▶ *Step 3: The iteration is repeated until the final time T is achieved.*

Simple Analysis of Binomial Tau-leaping

- ▶ A simple analysis of binomial tau-leaping as follows. Note that the number of j -th reactions:

$$\text{Poisson : } a_j \tau \pm \sqrt{a_j \tau}$$

$$\text{Binomial : } a_j \tau \pm \sqrt{a_j \tau \left(1 - a_j \tau / k_{\max}^{(j)}\right)^{\frac{1}{2}}}$$

Simple Analysis of Binomial Tau-leaping

- ▶ A simple analysis of binomial tau-leaping as follows. Note that the number of j -th reactions:

$$\text{Poisson : } a_j \tau \pm \sqrt{a_j \tau}$$

$$\text{Binomial : } a_j \tau \pm \sqrt{a_j \tau \left(1 - a_j \tau / k_{\max}^{(j)}\right)^{\frac{1}{2}}}$$

- ▶ In the law of rare events limit ($a_j \tau \ll k_{\max}^{(j)}$), they give same result; in the finite size case, the noise is different!

Stiff System: Model System 1

- ▶ Chemical reactions are usually stiff.

Stiff System: Model System 1

- ▶ Chemical reactions are usually stiff.

Example (Reversible reaction)

Consider the reversible reaction system



when C_1 and C_2 are both large.

Stiff System: Model System 1

- ▶ Chemical reactions are usually stiff.

Example (Reversible reaction)

Consider the reversible reaction system



when C_1 and C_2 are both large.

- ▶ Define $C_1 + C_2 = \lambda$, $X_t^1 + X_t^2 = X^T$ (total number). Here $\lambda \gg 1$. Then we have

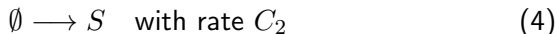
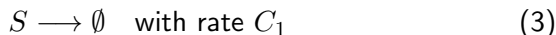
$$\mathbb{E}X_t^1 = \frac{C_2 X^T}{\lambda} (1 - e^{-\lambda t}) + e^{-\lambda t} X_0^1$$

$$\mathbb{E}X_t^2 = \frac{C_1 X^T}{\lambda} (1 - e^{-\lambda t}) + e^{-\lambda t} X_0^2$$

Stiff System: Model System 2

Example (Fast decaying)

Consider the following system



when $C_1 - C_2$ is large.

Define $\lambda = C_1 - C_2$, we have $\mathbb{E}X_t = e^{-\lambda t} X_0$.

Stability Analysis

- ▶ Now we perform the stability analysis for the stiff reversible reaction system. Suppose the explicit tau-leaping is applied.

$$\begin{cases} X_{n+1}^1 &= X_n^1 - \mathcal{P}(C_1 X_n^1 \delta t) + \mathcal{P}(C_2 X_n^2 \delta t) \\ X_{n+1}^2 &= X_n^2 - \mathcal{P}(C_2 X_n^2 \delta t) + \mathcal{P}(C_1 X_n^1 \delta t) \end{cases}$$

Stability Analysis

- ▶ Now we perform the stability analysis for the stiff reversible reaction system. Suppose the explicit tau-leaping is applied.

$$\begin{cases} X_{n+1}^1 &= X_n^1 - \mathcal{P}(C_1 X_n^1 \delta t) + \mathcal{P}(C_2 X_n^2 \delta t) \\ X_{n+1}^2 &= X_n^2 - \mathcal{P}(C_2 X_n^2 \delta t) + \mathcal{P}(C_1 X_n^1 \delta t) \end{cases}$$

- ▶ We have $X_n^1 + X_n^2 = X^T = \text{Const.}$. So we have

$$X_{n+1}^1 = X_n^1 - \mathcal{P}(C_1 X_n^1 \delta t) + \mathcal{P}(C_2 (X^T - X_n^1) \delta t).$$

Stability Analysis for the Mean

- ▶ Taking expectation we obtain

$$\mathbb{E}X_{n+1}^1 = (1 - \lambda\delta t)\mathbb{E}X_n^1 + C_2 X^T \delta t.$$

Stability Analysis for the Mean

- ▶ Taking expectation we obtain

$$\mathbb{E}X_{n+1}^1 = (1 - \lambda\delta t)\mathbb{E}X_n^1 + C_2 X^T \delta t.$$

- ▶ The stability condition is

$$|1 - \lambda\delta t| \leq 1 \implies \delta t \leq \frac{2}{\lambda}.$$

Stability Analysis for the Mean

- ▶ Taking expectation we obtain

$$\mathbb{E}X_{n+1}^1 = (1 - \lambda\delta t)\mathbb{E}X_n^1 + C_2 X^T \delta t.$$

- ▶ The stability condition is

$$|1 - \lambda\delta t| \leq 1 \implies \delta t \leq \frac{2}{\lambda}.$$

- ▶ When $\lambda \gg 1$, we have $\delta t \ll 1$. That is the stiffness!

Stability Analysis for the Mean

- ▶ Taking expectation we obtain

$$\mathbb{E}X_{n+1}^1 = (1 - \lambda\delta t)\mathbb{E}X_n^1 + C_2 X^T \delta t.$$

- ▶ The stability condition is

$$|1 - \lambda\delta t| \leq 1 \implies \delta t \leq \frac{2}{\lambda}.$$

- ▶ When $\lambda \gg 1$, we have $\delta t \ll 1$. That is the stiffness!
- ▶ As $n \rightarrow \infty$, we have

$$\mathbb{E}X_n^1 \rightarrow \frac{C_2}{\lambda} X^T,$$

which is the correct limit state.

Stability Analysis for the Variance

- ▶ Now consider the variance. At first we have

$$\text{Var}(Y) = \mathbb{E}(\text{Var}(Y|X)) + \text{Var}(\mathbb{E}(Y|X)).$$

Stability Analysis for the Variance

- ▶ Now consider the variance. At first we have

$$\text{Var}(Y) = \mathbb{E}(\text{Var}(Y|X)) + \text{Var}(\mathbb{E}(Y|X)).$$

- ▶ Then

$$\begin{aligned}\text{Var}(X_{n+1}^1) &= C_1 \delta t \mathbb{E}X_n^1 + C_2 \delta t \mathbb{E}(X^T - X_n^1) \\ &+ \text{Var}\left(X_n^1 - C_1 X_n^1 \delta t + C_2 (X^T - X_n^1) \delta t\right) \\ &= (1 - \lambda \delta t)^2 \text{Var}(X_n^1) + (C_1 - C_2) \delta t \mathbb{E}X_n^1 + C_2 \delta t X^T.\end{aligned}$$

Stability Analysis for the Variance

- ▶ Now consider the variance. At first we have

$$\text{Var}(Y) = \mathbb{E}(\text{Var}(Y|X)) + \text{Var}(\mathbb{E}(Y|X)).$$

- ▶ Then

$$\begin{aligned}\text{Var}(X_{n+1}^1) &= C_1 \delta t \mathbb{E} X_n^1 + C_2 \delta t \mathbb{E}(X^T - X_n^1) \\ &+ \text{Var}\left(X_n^1 - C_1 X_n^1 \delta t + C_2 (X^T - X_n^1) \delta t\right) \\ &= (1 - \lambda \delta t)^2 \text{Var}(X_n^1) + (C_1 - C_2) \delta t \mathbb{E} X_n^1 + C_2 \delta t X^T.\end{aligned}$$

- ▶ As $n \rightarrow \infty$, we have

$$\text{Var}(X_n^1) \rightarrow \frac{2}{2 - \lambda \delta t} \frac{C_1 C_2 X^T}{(C_1 + C_2)^2} = \frac{2}{2 - \lambda \delta t} \text{Var}(X_\infty^1) \geq \text{Var}(X_\infty^1).$$

Stability Analysis for the Variance

- ▶ Now consider the variance. At first we have

$$\text{Var}(Y) = \mathbb{E}(\text{Var}(Y|X)) + \text{Var}(\mathbb{E}(Y|X)).$$

- ▶ Then

$$\begin{aligned}\text{Var}(X_{n+1}^1) &= C_1 \delta t \mathbb{E}X_n^1 + C_2 \delta t \mathbb{E}(X^T - X_n^1) \\ &+ \text{Var}\left(X_n^1 - C_1 X_n^1 \delta t + C_2 (X^T - X_n^1) \delta t\right) \\ &= (1 - \lambda \delta t)^2 \text{Var}(X_n^1) + (C_1 - C_2) \delta t \mathbb{E}X_n^1 + C_2 \delta t X^T.\end{aligned}$$

- ▶ As $n \rightarrow \infty$, we have

$$\text{Var}(X_n^1) \rightarrow \frac{2}{2 - \lambda \delta t} \frac{C_1 C_2 X^T}{(C_1 + C_2)^2} = \frac{2}{2 - \lambda \delta t} \text{Var}(X_\infty^1) \geq \text{Var}(X_\infty^1).$$

- ▶ In order to get the right variance, we need $\lambda \delta t \rightarrow 0$. Since $\lambda \gg 1$, we need $\delta t \rightarrow 0$, which is a strict constraint.

Stiff System

Strategy: Implicit method to overcome stiffness.

- ▶ The first choice is

$$X_{n+1}^1 = X_n^1 - \mathcal{P}(C_1 X_{n+1}^1 \delta t) + \mathcal{P}(C_2 (X^T - X_{n+1}^1) \delta t).$$

Stiff System

Strategy: Implicit method to overcome stiffness.

- ▶ The first choice is

$$X_{n+1}^1 = X_n^1 - \mathcal{P}(C_1 X_{n+1}^1 \delta t) + \mathcal{P}(C_2 (X^T - X_{n+1}^1) \delta t).$$

- ▶ But the problem is how to sample $\mathcal{P}(C_1 X_{n+1}^1 \delta t)$. If we apply the iteration

$$X_{n+1}^{1,k+1} = X_n^1 - \mathcal{P}(C_1 X_{n+1}^{1,k} \delta t) + \mathcal{P}(C_2 (X^T - X_{n+1}^{1,k}) \delta t),$$

there will be no fixed point because of randomness.

Stiff System

Strategy: Implicit method to overcome stiffness.

- ▶ The first choice is

$$X_{n+1}^1 = X_n^1 - \mathcal{P}(C_1 X_{n+1}^1 \delta t) + \mathcal{P}(C_2 (X^T - X_{n+1}^1) \delta t).$$

- ▶ But the problem is how to sample $\mathcal{P}(C_1 X_{n+1}^1 \delta t)$. If we apply the iteration

$$X_{n+1}^{1,k+1} = X_n^1 - \mathcal{P}(C_1 X_{n+1}^{1,k} \delta t) + \mathcal{P}(C_2 (X^T - X_{n+1}^{1,k}) \delta t),$$

there will be no fixed point because of randomness.

- ▶ The second choice is semi-implicit method as

$$\begin{aligned} X_{n+1}^1 &= X_n^1 - C_1 X_{n+1}^1 \delta t + C_2 (X^T - X_{n+1}^1) \delta t \\ &\quad - \left[\mathcal{P}(C_1 X_n^1 \delta t) - C_1 X_n^1 \delta t \right] \\ &\quad + \left[\mathcal{P}(C_2 (X^T - X_n^1) \delta t) - C_2 (X^T - X_n^1) \delta t \right]. \end{aligned}$$

Stiff System

- ▶ Similar analysis as before shows the stability condition

$$\left| \frac{1}{1 + \lambda t} \right| \leq 1.$$

So the stiffness is resolved!

Stiff System

- ▶ Similar analysis as before shows the stability condition

$$\left| \frac{1}{1 + \lambda t} \right| \leq 1.$$

So the stiffness is resolved!

- ▶ But the variance

$$\text{Var}(X_n^1) \rightarrow \frac{2}{2 + \lambda \delta t} \text{Var}(X_\infty^1) \leq \text{Var}(X_\infty^1).$$

because of the damping effect of implicit method.

Stiff System

- ▶ Similar analysis as before shows the stability condition

$$\left| \frac{1}{1 + \lambda t} \right| \leq 1.$$

So the stiffness is resolved!

- ▶ But the variance

$$\text{Var}(X_n^1) \rightarrow \frac{2}{2 + \lambda \delta t} \text{Var}(X_\infty^1) \leq \text{Var}(X_\infty^1).$$

because of the damping effect of implicit method.

- ▶ Trapezoidal method is a good choice for linear problem. But the story goes on for nonlinear stiff problem!

Convergence Analysis

- ▶ Issue: the chemical reaction kinetics is a pure jump process with **state dependent intensity**.

Convergence Analysis

- ▶ Issue: the chemical reaction kinetics is a pure jump process with **state dependent intensity**.
- ▶ Construct jump process with state dependent intensity from constant jump intensity process (P. Protter, 1983) — Acceptance rejection method

$$\mu(dt) = \int_0^A 1_{\{0 < x \leq a_0(\mathbf{X}_t)\}} \lambda(dt \times dx).$$

$\lambda(dt \times dx)$ is the Poisson random measure generated from a constant jump intensity process. $\mu(dt)$ has intensity $a_0(\mathbf{X}_t)$.

Convergence Analysis

- ▶ Issue: the chemical reaction kinetics is a pure jump process with **state dependent intensity**.
- ▶ Construct jump process with state dependent intensity from constant jump intensity process (P. Protter, 1983) — Acceptance rejection method

$$\mu(dt) = \int_0^A 1_{\{0 < x \leq a_0(\mathbf{X}_t)\}} \lambda(dt \times dx).$$

$\lambda(dt \times dx)$ is the Poisson random measure generated from a constant jump intensity process. $\mu(dt)$ has intensity $a_0(\mathbf{X}_t)$.

- ▶ The SDE form for the CME

$$d\mathbf{X}_t = \sum_{j=1}^M \int_0^A \nu_j c_j(x; \mathbf{X}_{t-}) \lambda(dt \times dx),$$

where

$$c_j(x; \mathbf{X}_t) = \begin{cases} 1, & \text{if } x \in (\sum_{i=1}^{j-1} a_i(\mathbf{X}_t), \sum_{i=1}^j a_i(\mathbf{X}_t)], \\ 0, & \text{otherwise.} \end{cases}$$

Tau-leaping is an explicit Euler scheme

► Decomposition

$$\begin{aligned}d\mathbf{X}_t &= \sum_{j=1}^M \int_0^A \boldsymbol{\nu}_j c_j(x; \mathbf{X}_{t-}) m(dt \times dx) \\ &+ \sum_{j=1}^M \int_0^A \boldsymbol{\nu}_j c_j(x; \mathbf{X}_{t-}) (\lambda - m)(dt \times dx) \\ &= \mathbf{P}_1 + \mathbf{P}_2.\end{aligned}$$

Tau-leaping is an explicit Euler scheme

- ▶ Decomposition

$$\begin{aligned}d\mathbf{X}_t &= \sum_{j=1}^M \int_0^A \boldsymbol{\nu}_j c_j(x; \mathbf{X}_{t-}) m(dt \times dx) \\ &+ \sum_{j=1}^M \int_0^A \boldsymbol{\nu}_j c_j(x; \mathbf{X}_{t-}) (\lambda - m)(dt \times dx) \\ &= \mathbf{P}_1 + \mathbf{P}_2.\end{aligned}$$

- ▶ We call \mathbf{P}_1 the **drift term** and \mathbf{P}_2 is the **jump term**.

Tau-leaping is an explicit Euler scheme

- ▶ Decomposition

$$\begin{aligned}d\mathbf{X}_t &= \sum_{j=1}^M \int_0^A \boldsymbol{\nu}_j c_j(x; \mathbf{X}_{t-}) m(dt \times dx) \\ &+ \sum_{j=1}^M \int_0^A \boldsymbol{\nu}_j c_j(x; \mathbf{X}_{t-}) (\lambda - m)(dt \times dx) \\ &= \mathbf{P}_1 + \mathbf{P}_2.\end{aligned}$$

- ▶ We call \mathbf{P}_1 the **drift term** and \mathbf{P}_2 is the **jump term**.
- ▶ **Explicit Euler scheme — tau-leaping method!**

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \sum_{j=1}^M \boldsymbol{\nu}_j \mathcal{P}(a_j(\mathbf{X}_n) \delta t_n)$$

Other tau-leaping schemes

- ▶ Implicit tau-leaping: semi-implicit Euler of SDEs

Other tau-leaping schemes

- ▶ Implicit tau-leaping: semi-implicit Euler of SDEs
- ▶ Stochastic theta methods:

$$\begin{aligned}\mathbf{X}_{n+1} &= \mathbf{X}_n + \sum_{j=1}^M \theta \boldsymbol{\nu}_j \left(a_j(\mathbf{X}_{n+1}) - a_j(\mathbf{X}_n) \right) \delta t_n \\ &+ \sum_{j=1}^M \boldsymbol{\nu}_j \mathcal{P}(a_j(\mathbf{X}_n) \delta t_n).\end{aligned}$$

Other tau-leaping schemes

- ▶ Implicit tau-leaping: semi-implicit Euler of SDEs
- ▶ Stochastic theta methods:

$$\begin{aligned}\mathbf{X}_{n+1} &= \mathbf{X}_n + \sum_{j=1}^M \theta \boldsymbol{\nu}_j \left(a_j(\mathbf{X}_{n+1}) - a_j(\mathbf{X}_n) \right) \delta t_n \\ &+ \sum_{j=1}^M \boldsymbol{\nu}_j \mathcal{P}(a_j(\mathbf{X}_n) \delta t_n).\end{aligned}$$

- ▶ Milstein scheme: Not directly imply any implementable scheme!

Convergence Theorem

Theorem (Mean square convergence)

With the assumptions before we have

$$\sup_{n \leq N_T} \mathbb{E} |\mathbf{X}_n - \mathbf{X}_{t_n}|^2 \leq C\tau,$$

where $\tau = \max_n \delta t_n$.

Convergence Theorem

Theorem (Mean square convergence)

With the assumptions before we have

$$\sup_{n \leq N_T} \mathbb{E} |\mathbf{X}_n - \mathbf{X}_{t_n}|^2 \leq C\tau,$$

where $\tau = \max_n \delta t_n$.

Theorem (Weak convergence)

Under the assumptions, for any continuous function $g(\mathbf{x})$ satisfying exponential growth condition

$$|g(\mathbf{x})| \leq C_g B^{|\mathbf{x}|}, \quad \mathbf{x} \in \mathbb{R}^N \text{ and } C_g, B > 0.$$

We have

$$\left| \mathbb{E} g(\mathbf{X}_{N_T}) - \mathbb{E} g(\mathbf{X}_T) \right| \leq C\tau,$$

where $T = t_{N_T}$, $\tau = \max_n \delta t_n$.

Stationary Distribution

The chemical master equation (FPE) is as before. Denote it as $\partial_t P = \mathcal{L}P$. Here \mathcal{L} is the adjoint operator of the infinitesimal generator

$$\begin{aligned}\mathcal{L}^* u &= \sum_{j=1}^M a_j(\mathbf{x}) u(\mathbf{x} + \boldsymbol{\nu}_j, t) - \sum_{j=1}^M a_j(\mathbf{x}) u(\mathbf{x}, t) \\ &= \sum_{j=1}^M a_j(\mathbf{x}) \left(u(\mathbf{x} + \boldsymbol{\nu}_j, t) - u(\mathbf{x}, t) \right).\end{aligned}$$

For the stationary solution, we ask

$$\mathcal{L}P = 0.$$

Stationary Distribution

- ▶ For reversible reaction, we only consider the equation for x since $x + y = x^T$ ($\nu_1 = -1, \nu_2 = 1$):

$$\begin{aligned} & \left(C_1(x+1)p(x+1) - C_1xp(x) \right) + \\ & \left(C_2(x^T - x + 1)p(x-1) - C_2(x^T - x)p(x) \right) = 0. \end{aligned}$$

Stationary Distribution

- ▶ For reversible reaction, we only consider the equation for x since $x + y = x^T$ ($\nu_1 = -1, \nu_2 = 1$):

$$\begin{aligned} & \left(C_1(x+1)p(x+1) - C_1xp(x) \right) + \\ & \left(C_2(x^T - x + 1)p(x-1) - C_2(x^T - x)p(x) \right) = 0. \end{aligned}$$

- ▶ Define $a_2(x) = C_2(x^T - x)$, $a_1(x) = C_1x$, ($0 \leq x \leq x^T$), we have

$$\left(a_1(x+1)p(x+1) - a_2(x)p(x) \right) - \left(a_1(x)p(x) - a_2(x-1)p(x-1) \right) = 0.$$

Stationary Distribution

- ▶ For reversible reaction, we only consider the equation for x since $x + y = x^T$ ($\nu_1 = -1, \nu_2 = 1$):

$$\begin{aligned} & \left(C_1(x+1)p(x+1) - C_1xp(x) \right) + \\ & \left(C_2(x^T - x + 1)p(x-1) - C_2(x^T - x)p(x) \right) = 0. \end{aligned}$$

- ▶ Define $a_2(x) = C_2(x^T - x)$, $a_1(x) = C_1x$, ($0 \leq x \leq x^T$), we have

$$\left(a_1(x+1)p(x+1) - a_2(x)p(x) \right) - \left(a_1(x)p(x) - a_2(x-1)p(x-1) \right) = 0.$$

- ▶ If $x = 0$, $a_1(x)p(x) - a_2(x-1)p(x-1) = a(0)p(0) = 0$, we have the detailed balance $a_1(x)p(x) = a_2(x-1)p(x-1)$, so

$$\frac{p(x)}{p(x-1)} = \frac{a_2(x-1)}{a_1(x)} \implies \frac{p(x)}{p(0)} = \frac{a_2(x-1)}{a_1(x)} \frac{a_2(x-2)}{a_1(x-1)} \dots \frac{a_2(0)}{a_1(1)}$$

Stationary Distribution

- ▶ We obtain the stationary distribution

$$\begin{aligned} p(x) &= p(0) \left(\frac{C_2}{C_1} \right)^x \frac{x^T!}{x!(x^T - x)!} \\ &= \frac{x^T!}{x!(x^T - x)!} \left(\frac{C_2}{C_1 + C_2} \right)^x \left(\frac{C_1 + C_2}{C_1} \right)^x \end{aligned}$$

Stationary Distribution

- ▶ We obtain the stationary distribution

$$\begin{aligned} p(x) &= p(0) \left(\frac{C_2}{C_1} \right)^x \frac{x^T!}{x!(x^T - x)!} \\ &= \frac{x^T!}{x!(x^T - x)!} \left(\frac{C_2}{C_1 + C_2} \right)^x \left(\frac{C_1 + C_2}{C_1} \right)^x \end{aligned}$$

- ▶ From the normalization we have

$$p(x) \sim B(x^T, q), \quad q = \frac{C_2}{C_1 + C_2}$$

with mean $x^T C_2 / (C_1 + C_2)$, and variance $x^T C_1 C_2 / (C_1 + C_2)$.