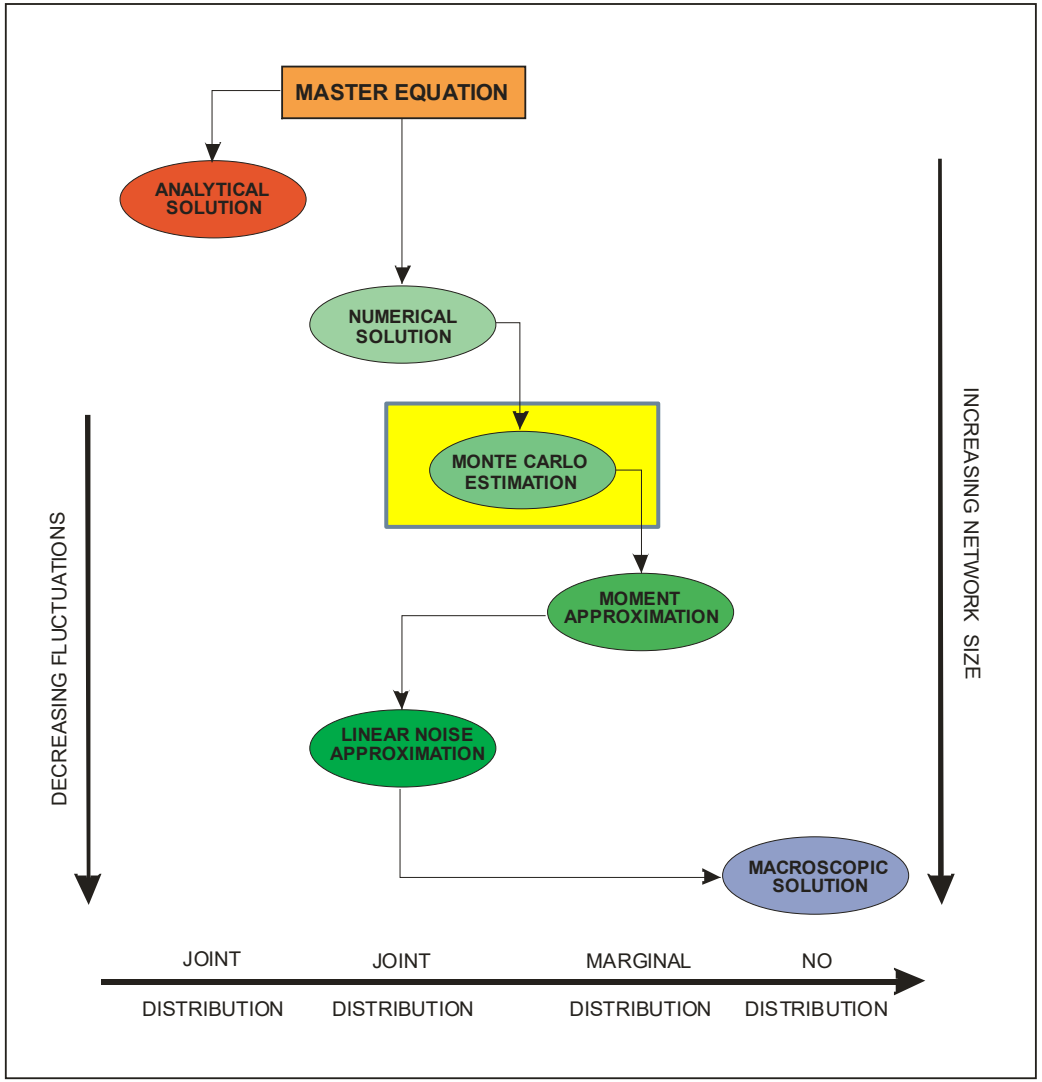
LECTURE #6

JOHN GOUTSIAS
WHITAKER BIOMEDICAL ENGINEERING INSTITUTE
THE JOHNS HOPKINS UNIVERSITY
BALTIMORE, MD 21218

SOLVING THE MASTER EQUATION – PART 3

Available Methods



Monte Carlo Estimation

- We can use Monte Carlo sampling to evaluate the statistical behavior of a Markovian reaction network.
- If we can generate L sample DA trajectories:

$$\{\mathbf{z}^{(l)}(t), t > 0\}, \quad l = 1, 2, \ldots, L$$

of the DA process $\mathbf{Z}(t)$, then we can estimate moment dynamics, such as means and covariances

$$\{\mu_Z(m;t) = \mathrm{E}[Z_m(t)],\ t > 0\}$$
$$\{c_Z(m,m';t) = \mathrm{cov}[Z_m(t), Z_{m'}(t)],\ t > 0\}$$

Monte Carlo Estimation

- We can do so, by using the following Monte Carlo estimators:

$$\hat{\mu}_Z(m;t) = \frac{1}{L}\sum_{l=1}^{L} z_m^{(l)}(t)$$

$$\hat{c}_Z(m,m';t) = \frac{1}{L-1}\sum_{l=1}^{L}\left[z_m^{(l)}(t) - \hat{\mu}_Z(m;t)\right]\left[z_{m'}^{(l)}(t) - \hat{\mu}_Z(m';t)\right]$$

- We can also estimate the probability distribution $p_Z(\mathbf{z};t)$ by using

$$\hat{p}_Z(\mathbf{z};t) = \frac{1}{L}\sum_{l=1}^{L}\delta(\mathbf{z}^{(l)}(t) - \mathbf{z})$$

Kronecker delta

Monte Carlo Estimation

- Due to the simple relationship $\mathbf{X}(t) = \mathbf{x}_0 + \mathbf{S}\mathbf{Z}(t)$ between the DA and population processes, we can use similar estimators to approximate the dynamic evolution of the corresponding population statistics.
- Unfortunately, to obtain sufficiently accurate Monte Carlo estimates, we need numerous sample trajectories.
- This is computationally inefficient, especially when estimating high-order moments or probability distributions.
- There is a need for computationally efficient approaches for sampling the ME.

Exact Sampling

- The simplest way to draw samples from the ME is to use the Gillespie algorithm.
- This algorithm is based on answering the following two questions:
  - Given that we are at time t, when will a reaction occur?
  - Which reaction will next occur?
- We must calculate the probability $p_t(\tau, m)$ that, given we are at time t, the next reaction occurs at time $t+\tau$ and that this reaction is the m-th reaction.

Exact Sampling

- Using simple probabilistic arguments, it has been shown by Gillespie that

$$p_t(\tau, m) = \alpha_m(\mathbf{z}(t)) d\tau \exp\left\{-\tau \sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right\}$$

see supplement #3 for details

- This implies

$$p_t(\tau, m) = \frac{\alpha_m(\mathbf{z}(t))}{\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))} \left[\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right] \exp\left\{-\tau \sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right\} d\tau$$

Exact Sampling

- Therefore, we have that

$$p_t(\tau, m) = r_t(m) e_t(\tau) d\tau$$

where

probability density of time of next reaction (exponential)

$$e_t(\tau) = \left[\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t)) \right] \exp\left\{ -\tau \sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t)) \right\}, \quad \tau \geq 0$$

$$r_t(m) = \frac{\alpha_m(\mathbf{z}(t))}{\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))}$$

probability that the next reaction will be the *m*-th reaction

Exact Sampling

- The time of the next reaction and the index of the next reaction are statistically independent random variables with probability density and mass functions $e_t(\tau)$ and $r_t(m)$.

- This leads to the following simulation algorithm.

The Gillespie Algorithm

1. Initialize by setting $\mathbf{z}(0) = \mathbf{0}$ and $t = 0$.
2. Calculate the propensity functions $\alpha_m(\mathbf{z}(t)), m = 1, 2, ..., M$.
3. Choose the time τ of occurrence of the next reaction by drawing a sample from the exponential distribution $e_t(\tau)$.
4. Choose the index m of the next reaction by drawing a sample from the discrete distribution $r_t(m)$.
5. Update the DAs at time $t + \tau$ by setting $\mathbf{z}(t + \tau) = \mathbf{z}(t) + \mathbf{e}_m$, where $\mathbf{e}_m$ is the m-th column of the $M \times M$ identity matrix and set $\mathbf{z}(t') = \mathbf{z}(t)$, for every $t \le t' < t + \tau$.
6. If $t + \tau < t_{\max}$, advance the time ahead by setting $t + \tau \to t$ and return to **STEP 2**. Otherwise set $\mathbf{x}(t) = \mathbf{x}_0 + \mathbf{S}\mathbf{z}(t)$, for $0 \le t \le t_{\max}$ and **STOP**.

The Gillespie Algorithm

- The exponential distribution

$$e_t(\tau) = \left[\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right] \exp\left\{-\tau \sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right\}$$

  implies that the mean time of the next reaction is given by

$$\left[\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right]^{-1}$$

- Moreover, the variance is given by

$$\left[\sum_{m'=1}^{M} \alpha_{m'}(\mathbf{z}(t))\right]^{-2}$$

- When at least one reaction is characterized by a large propensity value, the Gillespie algorithm is forced to move in small steps on the average.
- This can result in large computational burden.

The Gillespie Algorithm

- The Gillespie algorithm is computationally demanding, especially when applied to large and highly reactive systems.
- It requires faithful simulation of each reaction.
- A typical realization of the DA process may require numerous samples to be drawn from the probability distributions $e_t(\tau)$ and $r_t(m)$.
- Improvements, by means of "leaping", are available that significantly reduce computations, but the methods are still inefficient when used in conjunction with Monte Carlo estimation.

Gaussian Leaping

- In general, the DA process satisfies the following equation:

$$Z_m(t) = P_m\left[\int_0^t \alpha_m(\mathbf{Z}(t'))dt'\right], \quad t > 0$$

where $P_m, m = 1,2,\ldots,M$, are statistically independent Poisson random variables with unit rates.

$$P[\lambda] \sim \Pr[P = k] = \frac{\lambda^k}{k!}e^{-\lambda}, k = 0,1,\ldots.$$

Poisson random variable with rate λ.

- Because of the Markovian nature of the process, we also have that

$$Z_m(t+\tau) = Z_m(t) + P_m\left[\int_t^{t+\tau} \alpha_m(\mathbf{Z}(t'))dt'\right]$$

Gaussian Leaping

$$Z_m(t+\tau) = Z_m(t) + P_m\left[\int_t^{t+\tau} \alpha_m(\mathbf{Z}(t'))dt'\right]$$

- If the occurrence of reactions within the time interval $[t, t+\tau)$ does not appreciably affect the propensity functions, then (leap condition)

$$Z_m(t+\tau) \simeq Z_m(t) + P_m\left[\alpha_m(\mathbf{Z}(t))\tau\right]$$

- If the expected occurrence of each reaction during $[t, t+\tau)$ is much larger than one (firing condition), then the Poisson distribution can be approximated by a Gaussian distribution.

Gaussian Leaping

- In this case, we can approximate the DA process $\mathbf{Z}(t)$ with another process $\hat{\mathbf{Z}}(t)$ that satisfies

$$\hat{Z}_m((j+1)\tau) = \hat{Z}_m(j\tau) + \alpha_m(\hat{\mathbf{Z}}(j\tau))\tau + \sqrt{\alpha_m(\hat{\mathbf{Z}}(j\tau))\tau}\, G_m^{(j)}$$

where $\{G_m^{(j)}, j = 0,1,\ldots, m = 1,2,\ldots,M\}$ are mutually independent standard Gaussian random variables that are statistically independent of $\hat{\mathbf{Z}}$.

- As $\tau \to 0^+$, this equation converges to the Langevin equation

$$d\hat{Z}_m(t) = \alpha_m(\hat{\mathbf{Z}}(t))dt + \sqrt{\alpha_m(\hat{\mathbf{Z}}(t))}\, dW_m(t)$$

increments of independent standard Brownian motions

Gaussian Leaping

- Equation

$$\hat{Z}_m((j+1)\tau) = \hat{Z}_m(j\tau) + \alpha_m(\hat{\mathbf{Z}}(j\tau))\tau + \sqrt{\alpha_m(\hat{\mathbf{Z}}(j\tau))\tau}\, G_m^{(j)}$$

  provides a numerical method for solving the Langevin equation, known as the Euler-Maruyama method.

- The approximation method based on this equation is known as Gaussian leaping method.

- Gaussian leaping requires that we use small enough τ so we can obtain a sufficiently good approximation to the Langevin equations.

https://en.wikipedia.org/wiki/Euler-Maruyama_method

Gaussian Leaping

- Gaussian leaping may result in crude approximations of the DA and population processes.
- The main culprit is the difficulty in determining an appropriate time step τ so that the two required leaping and firing conditions are simultaneously satisfied.
- Gaussian approximation are usually not be appropriate.
- The method may produce reaction occurrences within a time interval $[j\tau,(j+1)\tau)$ that may result in negative species populations, which may not be appropriate in certain types of reaction networks.
- Extreme caution should be exercised when replacing the ME with the Langevin equations.

Poisson Leaping

$$Z_m(t+\tau) = Z_m(t) + P_m\left[\int_t^{t+\tau} \alpha_m(\mathbf{Z}(t'))dt'\right]$$

- If the occurrence of reactions within the time interval $[t, t+\tau)$ does not appreciably affect the propensity functions, then (leap condition)

$$Z_m(t+\tau) \approx Z_m(t) + P_m\left[\alpha_m(\mathbf{Z}(t))\tau\right]$$

- If the expected occurrence of each reaction during $[t, t+\tau)$ is much larger than one (firing condition), then the Poisson distribution can be approximated by a Gaussian distribution.

Poisson Leaping

- In this case, we can approximate the DA process $\mathbf{Z}(t)$ with another process $\hat{\mathbf{Z}}(t)$ that satisfies

$$\hat{Z}_m((j+1)\tau) = \hat{Z}_m(j\tau) + P_m\left[\alpha_m(\hat{\mathbf{Z}}(j\tau))\tau\right]$$

- The resulting method is usually referred to as Poisson leaping method.
- This approximation relaxes the second condition required by Gaussian leaping.

Poisson Leaping

- We expect to obtain accurate samples of the DA process, if we choose a time step τ that sufficiently satisfies the leap condition.
- The problem here is to determine the maximum possible value of τ such that the leap condition is approximately satisfied.
- If τ is not chosen appropriately, the resulting method may produce negative populations (same problem as in Gaussian leaping).

Poisson Leaping

- To avoid negative populations, it has been suggested to approximate the Poisson distribution by a binomial distribution.

$$\Pr[P=k]=\frac{\lambda^k}{k!}e^{-\lambda}, k=0,1,.... \qquad \Pr[B=k]=\binom{n}{k}p^k(1-p)^{n-k}, k=0,1,....,n$$

- The main rationale behind this choice is that the maximum number of occurrences produced by the binomial distribution is always bounded and easily controlled by one of the two parameters (parameter *n*) used to specify the distribution.

Deterministic Leaping

- If, in addition to the two conditions required by Gaussian leaping, we have that

$$\alpha_m(\mathbf{Z}(j\tau))\tau \gg \sqrt{\alpha_m(\mathbf{Z}(j\tau))\tau}, \text{ for every } j = 0,1,\ldots, m = 1,2,\ldots,M$$

  then the DA process will satisfy

$$z_m((j+1)\tau) \simeq z_m(j\tau) + \alpha_m(\mathbf{z}(j\tau))\tau, \text{ for } j = 0,1,\ldots, m = 1,2,\ldots,M$$

- In this case, we can compute the DA process in a simple iterative fashion.
- However, we can only produce a deterministic (fluctuation-free) DA process.
- Used only when stochastic fluctuations are negligible.

Maximum Entropy Approximation

- Estimating the probability distributions $p_Z(\mathbf{z};t)$ and $p_X(\mathbf{x};t)$ by sampling the ME can be computationally demanding and, in most cases, intractable.
- Depending on available data, the size of the reaction network, and available computational resources, it may only be possible to measure or accurately estimate the first few moments

$$\mu_X^{(k)}(n;t) = \mathrm{E}[X_n^k(t)], k = 1,2,...,K$$

of the population process.

Maximum Entropy Approximation

- By invoking the principle of maximum entropy (MaxEnt), we may be able to approximately derive an analytical form for the marginal probability distribution

$$p_X(x_n;t) = \Pr[X_n(t) = x_n]$$

https://en.wikipedia.org/wiki/Principle_of_maximum_entropy

Maximum Entropy Approximation

Principle of maximum entropy:

An appropriate approximation of the true-but-unknown distribution of $X_n(t)$ is the probability distribution $p_X(x_n;t)$ that maximizes the Shannon entropy

$$S(p_X;t) = -\sum_{x_n} p_X(x_n;t) \ln p_X(x_n;t)$$

subject to known information about $X_n(t)$ - i.e., knowledge of the support of $p_X(x_n;t)$ and of the moments $\mu_X^{(k)}(n;t)$.

https://en.wikipedia.org/wiki/Entropy_%28information_theory%29

Maximum Entropy Approximation

- **Solution of MaxEnt:**

$$\hat{p}_X(x_n;t) = \frac{1}{\zeta(t)} \exp\left\{-\sum_{k=1}^{K} \lambda_k(t) x_n^k\right\}, \quad x_n \geq 0, t > 0$$

$$\zeta(t) = \sum_{u_n} \exp\left\{-\sum_{k=1}^{K} \lambda_k(t) u_n^k\right\}$$

where the parameters $\lambda_k(t)$ must be chosen so that

$$\sum_{x_n} x_n^k \hat{p}_X(x_n;t) = \hat{\mu}_X^{(k)}(n;t), \text{ for } k = 1,2,...,K$$

estimated by Monte Carlo sampling or computed from available experimental measurements

Maximum Entropy Approximation

- **Special Case #1: Only the first moment is known or estimated**

$$\hat{p}_X(x_n;t) = \left[\frac{1}{1+\mu_X^{(1)}(n;t)}\right]\left[\frac{\mu_X^{(1)}(n;t)}{1+\mu_X^{(1)}(n;t)}\right]^{x_n}, \quad x_n \geq 0, t > 0$$

geometric distribution see supplement #4 for details

- **Special Case #2: Only the first two moments are known or estimated**

$$\hat{p}_X(x_n;t) = \left(\sum_{u\geq 0}\exp\left\{-\lambda_1(t)u - \lambda_2(t)u^2\right\}\right)^{-1}\exp\left\{-\lambda_1(t)x_n - \lambda_2(t)x_n^2\right\},$$
$$x_n \geq 0, t > 0$$

quadratic Gibbs distribution

Remarks

- Determining the MaxEnt distribution for the case of a random vector becomes increasingly difficult as the dimensionality increases.
- MaxEnt may produce probability distributions that assign nonzero probability to stoichiometrically impossible population states.
- This problem may be addressed by using MaxEnt to approximate the probability distribution $p_Z(\mathbf{z};t)$ of the DA process when possible.