

Math 636 - Mathematical Modeling

Stochastic Models

Joseph M. Mahaffy,
(jmahaffy@mail.sdsu.edu)

Department of Mathematics and Statistics
Dynamical Systems Group
Computational Sciences Research Center
San Diego State University
San Diego, CA 92182-7720

<http://jmahaffy.sdsu.edu>

Fall 2017



Outline

- 1 Markov Chain Theory
 - Gerschgorin Circle theorem
 - Princeton Forest Ecosystem
- 2 Gillespie's Method
 - Master Equation
 - Specific Simulation Algorithm
 - Gillespie Algorithm for Lotka Reactions



Markov Chain

Markov Chain

The simplest of *stochastic processes* is one where the current state decides the next state of the system.

Definition (Markov Process)

A *Markov process* is a discrete stochastic process that depends only on the current state, so is independent of its past history.

Definition (Markov Chain)

A *Markov chain* is a model that follows a series of steps using a *Markov process* at each step.



Markov Model

Markov Model for transitions is a *discrete dynamical model* that follows stochastic changes among certain possible states.

- Consider a system that has n possible *states*.
- Assume over a fixed time period there is a certain probability t_{ij} that the system moves from state j into state i .
- These *transition probabilities* form a *transition matrix*, $T = (t_{ij})$, where the columns sum to one.
- Define a probability vector, $\mathbf{x} = (x_1, \dots, x_n)^T$, with nonnegative entries summing to one.
- A *general Markov model* for transitions has the form of a *discrete dynamical system* given by

$$\mathbf{x}_{n+1} = T\mathbf{x}_n.$$



Markov Model

Consider the **Markov Model for transitions** given by:

$$\mathbf{x}_{n+1} = T\mathbf{x}_n.$$

- Since the columns of T sum to one, the **dominant eigenvalue** is $\lambda_1 = 1$.
- The **associated eigenvector** (normalized) provides the **equilibrium distribution**, provided some power of T has all positive entries.
- It is easy to see that $\lambda_1 = 1$ is an eigenvalue by considering looking at $\mathbf{x} = [1, \dots, 1]$, since $\mathbf{x}T = \mathbf{x}$.
- The **Gerschgorin Circle theorem**, which states that all eigenvalues of a matrix, (t_{ij}) , lie inside a circle radius $C_j = \sum_{j \neq i} t_{ij}$ with center at t_{jj} , shows all others have magnitude less than 1.



Gerschgorin Circle theorem

Theorem (Gerschgorin Circle Theorem^a)

Let A be a complex $n \times n$ matrix, with entries a_{ij} . For $i \in \{1, \dots, n\}$ let $R_i = \sum_{j \neq i} |a_{ij}|$ be the sum of the absolute values of the non-diagonal entries in the i^{th} row. Let $D(a_{ii}, R_i)$ be the closed disc centered at a_{ii} with radius R_i . Such a disc is called a Gerschgorin disc. Every eigenvalue of A lies within at least one of the Gerschgorin discs $D(a_{ii}, R_i)$.

Corollary

The eigenvalues of A must also lie within the Gerschgorin discs C_j corresponding to the columns of A .

Since the **transition matrix**, T , has columns summing to $\mathbf{1}$ with one **eigenvalue**, $\lambda_1 = 1$, then all remaining **eigenvalues** have magnitude less than 1.

^ahttps://en.wikipedia.org/wiki/Gerschgorin_circle_theorem, viewed 11/17

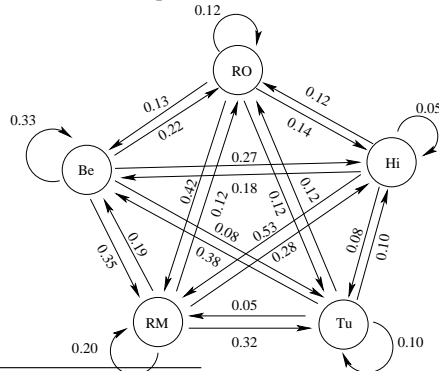


Princeton Forest Ecosystem

1

Princeton Forest Ecosystem: A complex model for the successional dynamics for the Princeton forest ecosystem was created by Horn^{1 2}

Transitional probabilities were found for five dominant species of trees based on which species replaced a resident species of tree that dies.



¹Horn, H. S. (1975). Forest succession, *Scientific American*, **232**, 90-98.

²Horn, H. S. (1975). Markovian properties of forest succession. In M. L. Cody and J. M. Diamond, ed., *Ecology and Evolution of Communities*, 196-211, University Press, Cambridge, MA.



Princeton Forest Ecosystem

2

Princeton Forest Ecosystem: Assume that the ordering of the probability state vector is Red Oak, Hickory, Tulip tree, Red Maple, and Beech (in that order), then the transition matrix is given by:

$$T = \begin{pmatrix} 0.12 & 0.14 & 0.12 & 0.12 & 0.13 \\ 0.12 & 0.05 & 0.08 & 0.28 & 0.27 \\ 0.12 & 0.10 & 0.10 & 0.05 & 0.08 \\ 0.42 & 0.53 & 0.32 & 0.20 & 0.19 \\ 0.22 & 0.18 & 0.38 & 0.35 & 0.33 \end{pmatrix}, \quad \text{with} \quad \begin{pmatrix} \text{Red Oak} \\ \text{Hickory} \\ \text{Tulip} \\ \text{Red Maple} \\ \text{Beech} \end{pmatrix}.$$

The normalized eigenvector associated with $\lambda_1 = 1$ is

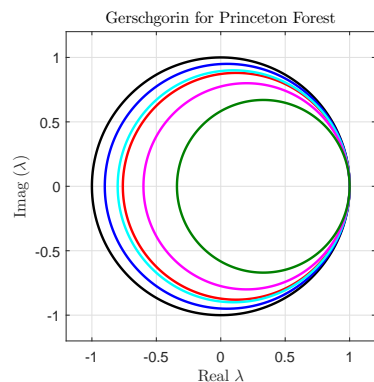
$$\mathbf{x}_e = \begin{pmatrix} 0.1269 \\ 0.1955 \\ 0.0816 \\ 0.2992 \\ 0.2968 \end{pmatrix}.$$

This eigenvector shows that the predicted climax forest community should be approximately 12.69% Red Oak, 19.55% Hickory, 8.16% Tulip tree, 29.92% Red Maple, and 29.68% Beech.



Princeton Forest Ecosystem: Below we graphically view the *Gerschgorin Circle Theorem* for our matrix, showing all *eigenvalues* lie in the unit circle.

$$T = \begin{pmatrix} 0.12 & 0.14 & 0.12 & 0.12 & 0.13 \\ 0.12 & 0.05 & 0.08 & 0.28 & 0.27 \\ 0.12 & 0.10 & 0.10 & 0.05 & 0.08 \\ 0.42 & 0.53 & 0.32 & 0.20 & 0.19 \\ 0.22 & 0.18 & 0.38 & 0.35 & 0.33 \end{pmatrix}.$$



SDSU

Princeton Forest Ecosystem: Consider the action of the *transition matrix*, T , defined above on some initial tree distribution.

- The transition matrix considers what occurs in each “generation,” which would be a succession event.
- Below we simulate what the model predicts would occur starting with all Red Oaks and following **5** generations, *i.e.*, if $\mathbf{x}_0 = [1, 0, 0, 0, 0]^T$

$$\mathbf{x}_i = T\mathbf{x}_{i-1}, \quad i = 1, \dots, 5.$$

Generation	\mathbf{x}_1	\mathbf{x}_2	\mathbf{x}_3	\mathbf{x}_4	\mathbf{x}_5	e.v.
Red Oak	0.12	0.1246	0.1273	0.1268	0.1269	0.1269
Hickory	0.12	0.207	0.1939	0.1956	0.1954	0.1955
Tulip	0.12	0.077	0.0823	0.0816	0.0816	0.0816
Red Maple	0.42	0.2782	0.3018	0.2989	0.2993	0.2992
Beech	0.22	0.3132	0.2947	0.2971	0.2968	0.2968

SDSU

Introduction

- Many chemical reaction systems are very complex.
 - Hard to create detailed ordinary differential equation systems.
 - Problems when the number of reacting molecules is small.
- This is particularly true for biochemical reactions happening inside cells.
- The basic scheme is Gillespie's method ³
 - A stochastic approach to simulating chemical reactions by considering molecules in the reactions as a kind of random walk process.
 - This process is governed by differential-difference equation, called the *master equation*.

³Gillespie, D. T. (1977). Exact stochastic simulation of coupled chemical reactions, *J. Phys. Chem.*, **81**, 2340-2361.

SDSU

Chemical Reactions – ODEs: The traditional models for chemical kinetics use systems of ODEs of the form:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2, \dots, x_n), \\ \dot{x}_2 &= f_2(x_1, x_2, \dots, x_n), \\ &\vdots \\ \dot{x}_n &= f_n(x_1, x_2, \dots, x_n). \end{aligned}$$

- Often highly nonlinear systems determined by structures and rate constants for M chemical reactions.
- The models are continuous and deterministic.
- Biological situations commonly have small numbers of specific molecules and significant fluctuations.
- ODEs may not accurately follow the “average” molecular populations.
- This may be particularly significant for certain threshold switches.

SDSU

Stochastic Formulation of Chemical Kinetics

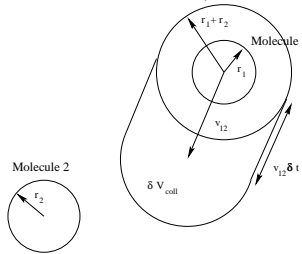
Stochastic Chemical Kinetics: Assume idealized spherical molecular species, S_1 and S_2 , in thermal, but not necessarily chemical equilibria.

A collision occurs when the center to center distance decreases to $r_{12} = r_1 + r_2$.

Calculate the rate of collisions in a fixed volume by estimating the number of S_2 molecules whose centers lie inside

$$\delta V_{coll} = \pi r_{12}^2 + v_{12} \delta t.$$

(If $\delta t \rightarrow 0$, then this becomes an ODE model.)



SDSU

Stochastic Formulation of Chemical Kinetics

Stochastic Markov Process: Assume that the molecules are distributed randomly and uniformly in volume, V .

- This implies that the probability that the center of an arbitrary S_2 molecule is inside δV_{coll} at time t is the ratio $\frac{\delta V_{coll}}{V}$.
- Average this ratio over velocity distributions of S_1 and S_2 .
- The average probability that a particular 1-2 pair will collide in a small time interval δt is given by:

$$\frac{\overline{\delta V_{coll}}}{V} = V^{-1} \pi r_{12}^2 \bar{v}_{12} \delta t.$$

- $\bar{v}_{12} = \sqrt{8kT/\pi r_{12}}$ is the **Maxwellian velocity distribution**.
- If there are X_1 molecules of S_1 and X_2 molecules of S_2 , then the probability of any 1-2 collisions is

$$X_1 X_2 V^{-1} \pi r_{12}^2 \bar{v}_{12} \delta t.$$

- These collisions are a **stochastic Markov process**.

SDSU

 Stochastic Reaction Constant c_μ

Stochastic Reaction Constant c_μ : Apply the above **stochastic Markov process** to reactive collisions, then

$X_1 X_2 c_1 dt =$ probability that an R_1 reaction will occur inside the volume, V , in the time interval $(t, t + dt)$.

- More generally, suppose that V contains a spatially homogeneous mixture of X_i molecules of species S_i , ($i = 1, \dots, N$).
- These N species interact through M specified chemical reaction channels, R_μ , ($\mu = 1, \dots, M$).
- Assume there exists M constants, c_μ , ($\mu = 1, \dots, M$), depending on physical properties of the molecules and the temperature, where $c_\mu dt =$ average probability that a particular combination of R_μ reactant molecules will react in the time interval $(t, t + dt)$.
- This equation is the fundamental hypothesis of the stochastic formulation of chemical kinetics and is valid for “well-mixed” systems.

SDSU

Stochastic Reaction and ODE Model

Stochastic Reaction and ODE Model: The stochastic reaction model is readily connected to the ODE model.

- The **mean stochastic approach** is closely related to the rate constants, k_i , in deterministic equations:

$$k_i = \frac{V c_i \langle X_i X_{i+1} \rangle}{\langle X_i \rangle \langle X_{i+1} \rangle},$$

where $\langle X \rangle =$ average ensemble and $\langle XY \rangle \simeq \langle X \rangle \langle Y \rangle$.

- It follows that $k_i \simeq V c_i$.
- The V remains in this formulation, whereas the **ODE models** use concentrations.
- There are a number of differences between this formulation and the ODE models, especially due to the discrete nature and other properties, but the models are considered closely related.

SDSU

Master Equation

Grand Probability function is given by:

$$\frac{d}{dt}P(X_1, \dots, X_N; t) = \sum_{\mu=1}^M [B_{\mu} - a_{\mu}P(X_1, \dots, X_N; t)].$$

- Its derivation is very similar to the derivation of the birth only process.
- This equation is harder to use than deterministic equations.

The discrete time version of this **grand probability function** with time step dt is given by:

$$P(X_1, \dots, X_N; t + dt) = P(X_1, \dots, X_N; t) \left[1 - \sum_{\mu=1}^M a_{\mu}dt \right] + \sum_{\mu=1}^M B_{\mu}dt.$$

- The **first term** on the right is the **probability of staying in the current state** in $[t, t + dt]$.
- The **second term** is the **probability that a reaction from another state** comes into the state $\{X_1, \dots, X_N\}$ in $[t, t + dt]$.

SDSU

Master Equation

Discrete Grand Probability function is given by:

$$P(X_1, \dots, X_N; t + dt) = P(X_1, \dots, X_N; t) \left[1 - \sum_{\mu=1}^M a_{\mu}dt \right] + \sum_{\mu=1}^M B_{\mu}dt.$$

- The quantity $a_{\mu}dt = c_{\mu}dt \times$ (number of distinct R_{μ} molecular combinations in the state (X_1, \dots, X_N))
- This equals the probability that an R_{μ} reaction occurs in V during $(t, t + dt)$ given that the system is in the state (X_1, \dots, X_N) at time t .
- The terms $B_{\mu}dt$ represent the probabilities that the system is one R_{μ} reaction removed from the state (X_1, \dots, X_N) .

SDSU

Stochastic Simulation Algorithm

Stochastic Simulation Algorithm is designed to move from the **probabilistic Master equation** to a simulation the **stochastic time evolution of the chemical reactions**.

Assume the system is in state (X_1, \dots, X_N) at time t , then to simulate the model there are two questions to answer:

- 1 When will the next reaction occur?
- 2 What kind of reaction is it?

We introduce the **probability function**:

$$P(\tau, \mu) \equiv \text{probability that given state } (X_1, \dots, X_N) \text{ at time } t, \text{ the next reaction in } V \text{ occurs in the infinitesimal time interval } (t + \tau, t + \tau + d\tau) \text{ and this reaction is an } R_{\mu} \text{ reaction.}$$

This is the **reaction probability density function** on the space of the **continuous variable** τ ($0 \leq \tau < \infty$) and the **discrete variable** μ ($\mu = 1, 2, \dots, M$).

These variables are needed for answering the two questions posed above.

SDSU

Stochastic Simulation Algorithm

The **algorithm** needs the analytical expressions for $P(\tau, \mu)$.

Define the following for each reaction, R_{μ}

$$h_{\mu} \equiv \text{number of distinct } R_{\mu} \text{ molecular reactant combinations available in the state } (X_1, \dots, X_N) \text{ } (\mu = 1, \dots, M).$$

- As examples:
 - $S_1 + S_2 \rightarrow \text{anything}$ gives $h_{\mu} = X_1 X_2$.
 - $2S_1 \rightarrow \text{anything}$ gives $h_{\mu} = \frac{X_1(X_1-1)}{2}$.

In general, h_{μ} is a combinatorial function of X_1, \dots, X_N , so

$$a_{\mu}dt \equiv h_{\mu}c_{\mu}dt = \text{probability that an } R_{\mu} \text{ reaction will occur in } V \text{ in } (t, t + dt) \text{ given that the system is in the state } (X_1, \dots, X_N) \text{ at time } t \text{ } (\mu = 1, \dots, M).$$

SDSU

Stochastic Simulation Algorithm

Write the *probability density function* as the product of $P_0(\tau)$.

- It is the probability that given that the system is in the state (X_1, \dots, X_N) at time t .
- No reaction occurs in the time interval $(t, t + \tau)$.
- The subsequent probability that an R_μ reaction occurs in the interval $(t + \tau, t + \tau + d\tau)$:

$$P(\tau, \mu)d\tau = P_0(\tau) \cdot a_\mu d\tau.$$

The expression

$$P_0(\tau) = \exp\left(-\sum_{\mu=1}^M a_\mu \tau\right),$$

is the *exponential waiting time* for a reaction to occur.



Stochastic Simulation Algorithm

It follows that the reaction probability density function satisfies:

$$P(\tau, \mu) = \begin{cases} a_\mu \exp(-a_0 \tau) & \text{if } 0 \leq \tau < \infty \text{ and } \mu = 1, \dots, M \\ 0 & \text{otherwise,} \end{cases}$$

where $a_\mu = h_\mu c_\mu$ ($\mu = 1, \dots, M$) and

$$a_0 \equiv \sum_{\nu=1}^M a_\nu \equiv \sum_{\nu=1}^M h_\nu c_\nu.$$

This probability is key to the *Stochastic Simulation Algorithm*.



Stochastic Simulation Algorithm

The *stochastic simulation algorithm* is a *Monte Carlo simulation* that uses **two random numbers** at each step of the process.

The **random numbers**, r_1 and r_2 , are selected from the unit interval:

- 1 Give the *waiting time*, τ , for a reaction to happen.
- 2 Define *specifically which reaction*, μ , occurs.

Chose these variables as follows:

$$\tau = \frac{1}{a_0} \ln\left(\frac{1}{r_1}\right),$$

$$\mu = \text{integer satisfying } \sum_{i=1}^{\mu-1} a_i \leq r_2 a_0 \leq \sum_{i=1}^{\mu} a_i.$$



Specific Simulation Algorithm

Simulation Algorithm for time evolution of a chemically reacting system.

Step 0 (Initialization): Input M reaction constants c_1, \dots, c_M and N initial molecular populations numbers X_1, \dots, X_N . Set $t = 0$ and reaction number $n = 0$. Initialize the random number generator.

Step 1: Calculate and store the M quantities $a_1 = h_1 c_1, \dots, a_M = h_M c_M$ for the current populations, where h_i is a function of X_1, \dots, X_N . Calculate and store $a_0 = \sum_{\mu=1}^M a_\mu$.

Step 2: Generate random numbers r_1 and r_2 . Compute

$$\tau = \frac{1}{a_0} \ln\left(\frac{1}{r_1}\right),$$

$$\mu = \text{integer satisfying } \sum_{i=1}^{\mu-1} a_i \leq r_2 a_0 \leq \sum_{i=1}^{\mu} a_i.$$

Step 3: Increase t by τ (add waiting time) and adjust molecular populations based on the reaction R_μ . (For example, if $S_1 + S_2 \rightarrow 2S_1$, then X_1 increases by one and X_2 decreases by one.) Increase the reaction counter by one, $n \rightarrow n + 1$.

Repeat **Steps 1–3** until the reaction reaches the time desired.

Run the simulation multiple times computing *averages* and *standard deviations*.



Discussion of the Simulation

Discussion of the Simulation

Advantages

- This method is exact and *mathematically rigorous*, designed to simulate stochastic events in the *spatially homogeneous master equation*.
- Not approximations of continuous changes with finite time steps, so allows sudden molecular changes.
- *Easily coded* independent of how complicated and coupled the chemical equations.
- Minimal computer memory required because of the *Markov process*.
- Can easily obtain *averages* and *variation* to collect statistics on the reactions.

Disadvantages

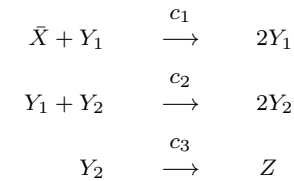
- Uses lots of *computer time*, so need high speed processors.
- Only a limited number of molecules and reactions are possible from a practical standpoint.
- Need *high quality random number generators* because of the huge number of random numbers being used.
- Statistical averages are computationally expensive.



Gillespie Algorithm for Lotka Reactions

Gillespie Algorithm⁴ for Lotka Reactions: The Lotka chemical reactions (developed by Lotka in 1920) that result in the famous *Lotka-Volterra predator-prey model*.

The chemical reactions are written:



The ODE or *classic Lotka-Volterra model* satisfies:

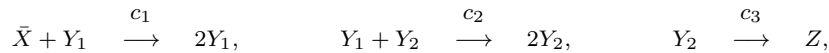
$$\begin{aligned} \frac{dY_1}{dt} &= c_1XY_1 - c_2Y_1Y_2, \\ \frac{dY_2}{dt} &= c_2Y_1Y_2 - c_3Y_2, \end{aligned}$$

⁴Gillespie, D. T. (1977). Exact stochastic simulation of coupled chemical reactions, *J. Phys. Chem.*, **81**, 2340-2361.



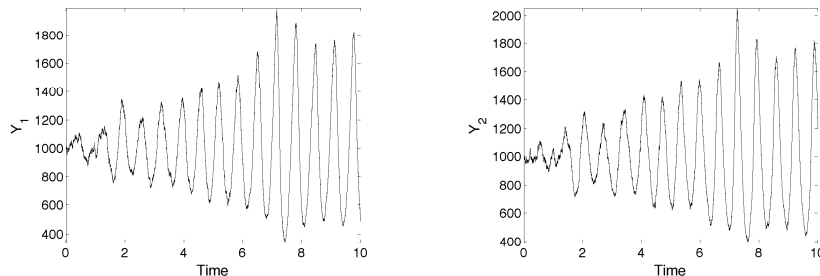
Gillespie Algorithm for Lotka Reactions

MatLab Code for the Gillespie Algorithm simulates the Lotka chemical reactions:



with the parameters, $c_1X = 10$, $c_2 = 0.01$, and $c_3 = 10$.

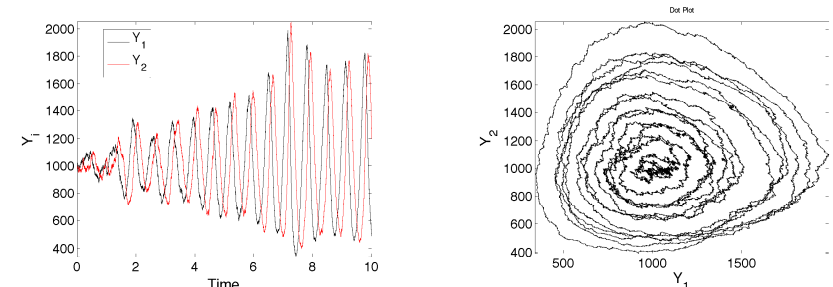
The nonzero equilibrium is $Y_{1e} = Y_{2e} = 1000$, and these values are used as starting values for the simulation.



Gillespie Algorithm for Lotka Reactions

Gillespie Algorithm Simulations for $t = 10$ sec: These simulations require hundreds of thousands of time steps because of the small size of the time steps from the algorithm.

Below is the composite of Y_1 and Y_2 along with a *phase portrait*.

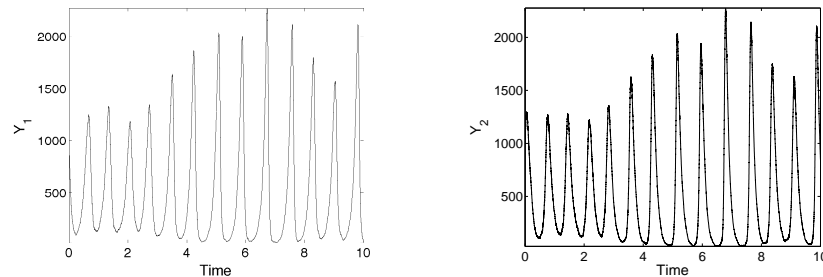


Gillespie Algorithm for Lotka Reactions

Gillespie Algorithm Simulation with $c_2 = 0.02$ in the *Lotka Reactions*.

The interaction of the two chemical species is increased, which affects the simulation by rapidly increasing the amplitude of oscillation and changing the equilibrium value.

Below is the *time series* for Y_1 and Y_2 .



SDSU

Gillespie Algorithm for Lotka Reactions

Below is a code created by Desmond J. Higham to perform the *Gillespie Algorithm for Lotka Reactions*

```

1  % LV_gill.M
2  % Simple implementation of the Stochastic ...
   Simulation Algorithm
3  % (or Gillespies algorithm) for the ...
   Lotka-Volterra system.
4  %
5  % rand('state',100) % Can fix rand # pattern
6  % stoichiometric matrix for rxs
7  V = [1 -1 0; 0 1 -1];
8  %%% Parameters and Initial Conditions
9  X = zeros(2,1);
10 X(1) = 1000; % initial molecules of X1
11 X(2) = 1000; % initial molecules of X2
12 Y1(1) = X(1); % store # of molecules
13 Y2(1) = X(2);
    
```

SDSU

Gillespie Algorithm for Lotka Reactions

MatLab code for the *Gillespie Algorithm for Lotka Reactions* continued.

```

14 % set chem rx coefficients
15 c(1) = 10; c(2) = 0.01; c(3) = 10;
16 t = 0; % initial time
17 T(1) = t;
18 tfinal = 10; % final time
19 i = 1;
20 while t < tfinal
21 % rx combination functions
22 a(1) = c(1)*X(1);
23 a(2) = c(2)*X(1)*X(2);
24 a(3) = c(3)*X(2);
25 asum = sum(a); % total a
    
```

SDSU

Gillespie Algorithm for Lotka Reactions

MatLab code for the *Gillespie Algorithm for Lotka Reactions* continued.

```

26 % generate rand # and find rx occurring
27 j = min(find(rand<cumsum(a/asum)));
28 % 2nd rand # for time until rx
29 tau = log(1/rand)/asum;
30 X = X + V(:,j); % Stochastic matrix adjusts X
31 t = t + tau;
32 i = i + 1;
33 T(i) = t;
34 Y1(i) = X(1);
35 Y2(i) = X(2);
36 end
    
```

SDSU

Gillespie Algorithm for Lotka Reactions

MatLab code for the *Gillespie Algorithm for Lotka Reactions* continued.

```

38 figure(101)
39 plot(T,Y1,'b-',T,Y2,'r-');grid;
40 xlim([0,10]);
41 fontlabs = 'Times New Roman'; % Font type used in ...
    labels
42 xlabel('$t$', 'FontSize',14, 'FontName',fontlabs,...
43     'interpreter','latex');
44 ylabel('Molecules', 'FontSize',14, 'FontName',fontlabs);
45 set(gca, 'FontSize',12);
46
47 figure(102)
48 plot(Y1,Y2,'b-');grid;
49 xlabel('$Y_1$', 'FontSize',14, 'FontName',fontlabs,...
50     'interpreter','latex');
51 ylabel('$Y_2$', 'FontSize',14, 'FontName',fontlabs,...
52     'interpreter','latex');
```

SDSU

ODE for Lotka Reactions

The ODE or *classic Lotka-Volterra model* satisfies:

$$\begin{aligned}\frac{dY_1}{dt} &= c_1XY_1 - c_2Y_1Y_2, \\ \frac{dY_2}{dt} &= c_2Y_1Y_2 - c_3Y_2,\end{aligned}$$

MatLab code for solving the *ODE Lotka Reactions* uses the function below.

```

1 function yp = lv_ode(t,y)
2 % LV model for ODE
3 yp = [10*y(1) - 0.01*y(1)*y(2);
4       -10*y(2) + 0.01*y(1)*y(2)];
5 end
```

SDSU

ODE for Lotka Reactions

MatLab script for solving the *ODE Lotka Reactions* satisfies.

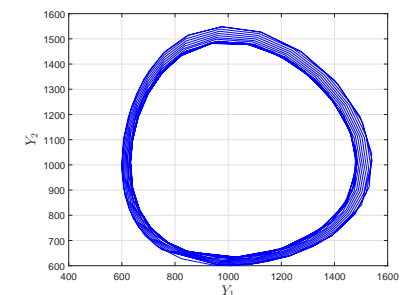
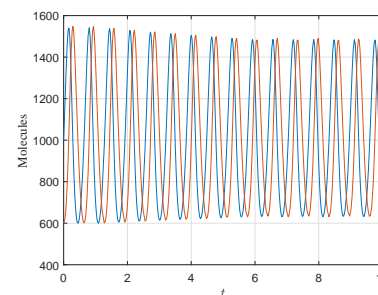
```

1 % Simulate LV Model
2 % Requires Stiff Solver ode15s
3 [t1,y]=ode15s(@lv_ode,[0,10],[1000,600]);
4
5 figure(101)
6 plot(t1,y);grid;
7 xlim([0,10]);
8 fontlabs = 'Times New Roman';
9 xlabel('$t$', 'FontSize',14, 'FontName',fontlabs,...
10     'interpreter','latex');
11 ylabel('Molecules', 'FontSize',14, 'FontName',fontlabs);
12 set(gca, 'FontSize',12);
13 print -depsc lv_ode_ts.eps
```

SDSU

ODE for Lotka Reactions

The solution for the *ODE Lotka Reactions* should be a perfect *periodic solution*, but numerical differences in the coefficients and large numbers of molecular species make this a *stiff ODE system*.



SDSU