Specification and Control of Stochastic Biochemical Systems

Eric Klavins
University of Washington
Chemical Reactions are Stochastic

At equilibrium:

- There are as many A molecules as B molecules.
- The number of forward reactions balances the number of reverse reactions.

In bulk, with a few nanomoles of molecules in solution, you do not see the fluctuations.

But if had only a few molecules, you would see things differently.
Outline

• Example Experimental Systems
  • What Stochasticity Can Do
  • Analytical Approaches
    • The Master Equation
    • Moment Dynamics
  • Simulation Based Approaches
    • Simulation Methods
    • Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic
Software Tools
Example: Single Molecule DNA Kinetics Are Observable

Example: Low Copy Numbers in Cells

\[ \emptyset \xrightarrow{\alpha_R} R \xrightarrow{\alpha_P} P \xrightarrow{\beta_P} \emptyset \]

\[ \emptyset \xrightarrow{\beta_R} \emptyset \]

[Graph showing fluorescence/size over time]
Example: Chemical Robotics

\[
P_1 + P_2 \underset{k_1}{\overset{k_{-1}}{\rightarrow}} P_1P_2
\]

\[
P_1P_2 + A \underset{k_2}{\overset{k_{-2}}{\rightarrow}} P_1 + P_2 + A'
\]

\[
A' \underset{u}{\rightarrow} A
\]

\[
y = N_{12}
\]
Outline

- Example Experimental Systems
- What Stochasticity Can Do
  - Analytical Approaches
    - The Master Equation
    - Moment Dynamics
  - Simulation Based Approaches
    - Simulation Methods
    - Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic
Software Tools
What Can You Make?

• Low copy number systems give you integer-valued variables.
  – So you can have states and registers.

• Low copy number systems can flip coins.
  – So you can implement randomized algorithms.
Example:
Computation via Register Machines

states $S_0, S_1, ..., S_n$
registers $R_0, R_1, ..., R_m$

\[ \text{inc}(i, r, j) \]
if the state is $i$, then increase register $r$ and go to state $j$

\[ S_i \xrightarrow{k} S_j + M_r \]

\[ \text{dec}(i, r, j, k) \]
if the state is $i$ and register $r$ is greater than zero, then decrease register $r$ and go to state $j$; otherwise go to state $k$

\[ S_i + M_r \xrightarrow{k} S_j \]
\[ S_i \xrightarrow{\varepsilon} S_k \]

David Soloveichik, Matthew Cook, Erik Winfree and Jehoshua Bruck, Computation with finite stochastic chemical reaction networks, NATURAL COMPUTING. Volume 7, Number 4, 615-633, 2008.
Example: Multiplication

Figure 6. A register machine that multiplies the initial contents of registers 0 and 1. Register 3 holds the final value and register 2 is swap space.

Klavins: Course notes.
Behavior of the Multiplier
Table 1: Averaged Simulation Data

<table>
<thead>
<tr>
<th></th>
<th>M1 = 5</th>
<th>M1 = 2</th>
<th>M1 = 2</th>
<th>M1 = 10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>M2 = 2</td>
<td>M2 = 2</td>
<td>M2 = 5</td>
<td>M2 = 10</td>
</tr>
<tr>
<td>Add</td>
<td>6.83</td>
<td>3.88</td>
<td>6.99</td>
<td>19.39</td>
</tr>
<tr>
<td>Sub</td>
<td>3.04</td>
<td>0.17</td>
<td>0.22</td>
<td>1.09</td>
</tr>
<tr>
<td>Mult</td>
<td>9.02</td>
<td>3.56</td>
<td>8.76</td>
<td>87.9</td>
</tr>
<tr>
<td>Div</td>
<td>2.26</td>
<td>1.08</td>
<td>1.09</td>
<td>4.21</td>
</tr>
<tr>
<td>Less</td>
<td>0.14</td>
<td>0.09</td>
<td>1.28</td>
<td>1.54</td>
</tr>
<tr>
<td>Greater</td>
<td>1.08</td>
<td>0.14</td>
<td>0.24</td>
<td>2.24</td>
</tr>
<tr>
<td>Equals</td>
<td>0.01</td>
<td>0.87</td>
<td>0.26</td>
<td>1.02</td>
</tr>
<tr>
<td>And</td>
<td>0.96</td>
<td>1.02</td>
<td>1.25</td>
<td>1.89</td>
</tr>
<tr>
<td>Inv</td>
<td>0.13</td>
<td>0.17</td>
<td>0.15</td>
<td>0.33</td>
</tr>
<tr>
<td>Or</td>
<td>0.99</td>
<td>0.94</td>
<td>1.1</td>
<td>1.19</td>
</tr>
<tr>
<td>Xor</td>
<td>0.18</td>
<td>0.1</td>
<td>0.04</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Figure 3: Entire BALU State Diagram

Zuyuan Zhang:  
Term project, 2009.
Toward Synthetic Development

Leader Election

Electing a leader in a group of identical processes.

A simple approach:

\[
\begin{align*}
\text{U} & \quad \text{UL} & \quad \text{UF} \\
\downarrow k_1 & \quad \downarrow k_2 & \quad \downarrow k_2 \\
\text{L} & \quad \text{FL} & \quad \text{FF}
\end{align*}
\]

Better would be to include conflict resolution.
A Leader Election Circuit
A Leader Election Circuit in gro

http://depts.washington.edu/soslab/gro
Another gro Program

http://depts.washington.edu/soslab/gro
Outline

- Example Experimental Systems
- What Stochasticity Can Do
  - Analytical Approaches
    - The Master Equation
    - Moment Dynamics
  - Simulation Based Approaches
    - Simulation Methods
    - Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic
Software Tools
Outline

• Example Experimental Systems
• What Stochasticity Can Do
• Analytical Approaches
  • The Master Equation
  • Moment Dynamics
• Simulation Based Approaches
  • Simulation Methods
  • Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic
Software Tools
Questions

• **Convergence:** In probability, in mean and variance, via a Lyapunov Function.

• **Correctness:** Do individual trajectories do behave as expected?

• **Refinement:** What does it mean for a stochastic process to a refinement or coarse-graining of another stochastic process?
Probability vs. Time

Assumptions:

• The probability of when a given pair of molecules reacts in the next \( dt \) seconds is independent of time.
• A given molecule is equally likely to interact with every other molecule in the system.

Example: Consider a system with one A and one B and the reaction:

\[
A + B \xrightarrow{k} C
\]

\( kdt \) is the probability that the reaction will occur in the next \( dt \) seconds. The two assumptions imply that the time of the reaction is distributed as an exponential random variable with p.d.f. and c.d.f.

\[
f(t) = ke^{-kt}
\]

\[
F(t) = 1 - e^{-\alpha t}
\]

Probability that the reaction has occurred by time \( t \).
The Master Equation

\[ A + B \xrightarrow{k} C \]

Initially 1 A and 1 B:

\[
\begin{align*}
    p_1(t) &= e^{-kt} \\
    p_2(t) &= 1 - e^{-kt}
\end{align*}
\]

Integrate

\[
\begin{align*}
    \dot{p}_1 &= -kp_1 \\
    \dot{p}_2 &= kp_1
\end{align*}
\]

Matrix Form

\[
Q = \begin{pmatrix}
    -k & 0 \\
    k & 0
\end{pmatrix}
\]

\[ \dot{p} = Qp \]
More Molecules

Initially 4 A’s and 3 B’s:

\[
\begin{pmatrix}
\dot{p}_0 \\
\dot{p}_1 \\
\dot{p}_2 \\
\dot{p}_3 \\
\end{pmatrix}
= \begin{pmatrix}
-12k & 0 & 0 & 0 \\
12k & -6k & 0 & 0 \\
0 & 6k & -2k & 0 \\
0 & 0 & 2k & 0 \\
\end{pmatrix}
\begin{pmatrix}
p_0 \\
p_1 \\
p_2 \\
p_3 \\
\end{pmatrix}
\]

Note: easy to solve via

\[x(t) = e^{At}x(0)\]
Another Example

\[ A + B \xrightarrow{k_1/k_2} C \]
\[ C \xrightarrow{k_3/k_4} D \]

Note: It is easy to reason about this network for a given initial state. But what if we want to say something about its behavior for arbitrary initial states? There is a different Markov process for each one!

Also: There may not be a finite number of states, even for a given initial condition.
The Register Machine Example

- It is easy to reason about these networks for a given initial state. But what if we want to say something about their behavior for arbitrary initial states? There is a different Markov process for each one!

- There may not be a finite number of states, even for a given initial condition.

Reactions.m contains code to do this.
Running Example: Controlling Gene Expression

Open Loop

Negative Feedback

Other Control Schemes?

Goal: Control mean and variance of $X$.

$r$ can be any function
- a constant
- a function of (the random variable) $X$
- a function of other species yet to be introduced
- etc.
Stationary Distribution of An Infinite System

\[ \emptyset \xrightarrow{k_1} X \xrightarrow{k_2} \emptyset \]

\[
\begin{align*}
\dot{p}_0 &= -k_1 p_0 + k_2 p_1 \\
\dot{p}_1 &= k_1 p_0 - (k_1 + k_2) p_1 + 2k_2 p_2 \\
\dot{p}_2 &= \frac{k_1}{k_1 p_1} - (k_1 + 2k_2) p_2 + 3k_2 p_3 \\
\dot{p}_3 &= \frac{k_1}{k_1 p_2} - (k_1 + 3k_2) p_3 + 4k_2 p_3 \\
\vdots
\end{align*}
\]

First Eqn:

\[ p_1^* = \frac{k_1}{k_2} p_0^* \]

Second Eqn:

\[ 0 = -k_1 p_1^* + 2k_2 p_2^* \]

Sum of 1st n Eqns:

\[ p_n^* = \frac{\alpha^n}{n!} p_0^* \quad \text{where} \quad \alpha = \frac{k_1}{k_2} \]

Using the fact that \( p \) is a probability distribution:

\[ \sum_{n=0}^{\infty} p_n^* = \sum_{n=0}^{\infty} \frac{\alpha^n}{n!} p_0^* = 1 \quad \Rightarrow \quad \sum_{n=0}^{\infty} \frac{\alpha^n}{n!} = \frac{1}{p_0^*} \]

\[ p_n^* = \frac{\alpha^n}{n!} e^{-\alpha} \]

Note: Mean and variance cannot be independently tuned by \( k_2 \). We need a better choice of control.
Solving Infinite Master Equations

• Although easy in simple cases, in general steady state distributions involve finding roots of high order polynomials symbolically.

• Some approaches:
  – Truncate the master equation (tends to work for numerical solutions)
  – Look at moments, instead of the full distribution
  – Simulate
Outline

• Example Experimental Systems
• What Stochasticity Can Do
• Analytical Approaches
  • The Master Equation
  • Moment Dynamics
• Simulation Based Approaches
  • Simulation Methods
• Approximate Abstraction/Refinement

Example: Arithmetic
Software Tools
Running Example: Control of Gene Expression
Moment Dynamics

Say we have $n$ reactions with rates $\lambda_i(x)$ and updates $x \mapsto \phi_i(x)$ for $i = 1$ to $n$.

Then

$$\frac{d}{dt} p(x) = \sum_r p(\phi_r^{-1}(x)) \lambda_r(\phi_r^{-1}(x)) - p(x) \lambda_r(x)$$

is the Master Equation.

Let $\psi(x)$ be a test function with expected value

$$\langle \psi \rangle = \sum_x \psi(x) p(x).$$

Taking the derivative,

$$\frac{d}{dt} \langle \psi \rangle = \sum_x \sum_r \psi(x) p(\phi_r^{-1}(x)) \lambda_r(\phi_r^{-1}(x)) \rangle - \sum_x \sum_r \psi(x) p(x) \lambda_r(x)$$

$$= \sum_r \sum_y \psi(\phi_r(y)) p(y) \lambda_r(y) - \sum_x \sum_r \psi(x) p(x) \lambda_r(x)$$

$$= \sum_r \sum_x \left[ \psi(\phi_r(x)) - \psi(x) \right] p(x) \lambda_r(x)$$

$$= \langle \sum_r \left[ \psi(\phi_r(X)) - \psi(X) \right] p(X) \lambda_r(X) \rangle$$

$$\triangleq \langle L \psi \rangle$$
Example (No Control): $\emptyset \xrightarrow{u} X \xrightarrow{k} \emptyset$

\[
\frac{d}{dt} \langle X \rangle = \langle ((X + 1) - X)u + ((X - 1) - X)kX \rangle
= \langle u \rangle - k\langle X \rangle
= u - k\langle X \rangle
\]

\[
\frac{d}{dt} \langle X^2 \rangle = \langle ((X + 1)^2 - X^2)u + ((X - 1)^2 - X^2)kX \rangle
= \langle 2X + 1u \rangle + \langle -2X + 1kX \rangle
= 2\langle Xu \rangle + \langle u \rangle - 2\langle X^2 \rangle + k\langle X \rangle
= u + (2u + k)\langle X \rangle - 2\langle X^2 \rangle
\]

\[
\left( \begin{array}{c}
\dot{\mu}_1 \\
\dot{\mu}_2
\end{array} \right) = \left( \begin{array}{cc}
-k & 0 \\
2u + k & -2k
\end{array} \right) \left( \begin{array}{c}
\mu_1 \\
\mu_2
\end{array} \right) + \left( \begin{array}{c}
1 \\
1
\end{array} \right) u
\]

\[
\mu_1 \rightarrow \frac{u}{k}
\]

\[
\mu_2 \rightarrow \frac{u^2}{\sigma^2}
\]

\[
\sqrt{\mu_2 - \mu_1^2} = \sigma \rightarrow \sqrt{\frac{u}{k}}
\]

Note: Mean and variance can not be independently tuned by $u$. We need a better choice of control.
Example (Feedback):

\[ \varnothing \xrightarrow{r-kX} X \xrightarrow{\beta} \varnothing \]

\( r - kX \) is impossible to implement (a rate can’t be negative).

But,

a) We are interested in the local behavior of the stationary distribution for smallish fluctuations and \( r - kX \) is the constant and linear part of whatever \( f(X) \) we do implement.

b) If \( u \) is non-linear, the moments don’t close:

\[
\mu = \begin{pmatrix}
\mu_1 \\
\mu_2 \\
\mu_3 \\
\vdots
\end{pmatrix} = \begin{pmatrix}
X_1 \\
X_2 \\
X_1^2 \\
X_2^2 \\
X_1X_2 \\
X_1^3 \\
\vdots
\end{pmatrix}
\]

If rates are not unimolecular or constant, then some moments of order \( n \) will depend on higher order moments.

: This is also when the master difficult.

: Moment equations may still help.

: Various approximations exist (e.g. cumulant truncation).
Example: Feedback

Define test functions

\[ \mu_x = \langle X \rangle \]
\[ \sigma_X^2 = \langle X^2 \rangle - \langle X \rangle^2 \]

Use \( \frac{d}{dt} \langle \psi \rangle = \langle L \psi \rangle \) to get

\[ \frac{d}{dt} \mu_X = r - (k + \beta)\mu_X \]
\[ \frac{d}{dt} \sigma_X^2 = r + (\beta - k)\mu_X - 2(\beta + k)\sigma_X^2 \]

The stationary distribution satisfies:

\[ \mu^* = \frac{r\beta}{(k + \beta)^2} \quad \text{and} \quad \kappa^* = \frac{r}{k + \beta} \]

Tunable, but
- mean sensitive to degradation rate
- variance coupled to \( r \).
Aside: Mathematica Code

Reactions.m includes:

- Mass action kinetics
- Markov Processes and Master Equations
- Gillespie Simulations
- Moment and Cumulant Dynamics Analysis
Idea: Proportional-Integral Control

\[ u = \gamma Z - kX \]

\[ \dot{Z} = r - X \]

This is now a continuous integrator.
At steady state, \( z = X \).

- \( Z_{\text{off}} \) in large supply
- Reverse rate saturates

\[ h(X, Z) = \frac{\nu Z^m}{(K_Z + Z^m)(K_X + X^n)} \approx h(\mu_X^*, \mu_Z^*) \]

\[ + \left. \frac{\partial h(X, Z)}{\partial X} \right|_{\mu = \mu^*} (X - \mu_X^*) \]

\[ + \left. \frac{\partial h(X, Z)}{\partial Z} \right|_{\mu = \mu^*} (Z - \mu_Z^*) \]

\[ = \alpha + \gamma Z - kX. \]
Mixed Continuous / Discrete

Suppose we have a concurrent continuous process

\[ \dot{z} = f(X_1, \ldots, X_n, z). \]

Then the $L$ can be extended to

\[ L\psi = \frac{\partial \psi}{\partial z} f(X_1, \ldots, X_n, z) + \sum_i (\psi_{new} - \psi) k_i \]

\[ \frac{d}{dt} \langle \psi \rangle = \langle L\psi \rangle \quad \text{still works!} \]
Proportional-Integral Control

\[ u \xrightarrow{\varnothing} X \xrightarrow{k} \varnothing \]

\[ \dot{z} = X - r \]

\[ u = h[-k_P(X - r) - k_I\dot{z}] \]

Stable (eigenvalues in left half plane) = convergence in mean and variance.

The mean value of X converges to r (insensitively).

\[ \langle \dot{Z} \rangle = \langle X \rangle - r = 0 \Rightarrow \langle X \rangle^* = r \]

The steady state standard deviation is tunable via \( k_P \)

\[ \sqrt{\langle X^2 \rangle - \langle X \rangle^2} = \sigma \rightarrow \sqrt{\frac{kr + kpr^2}{k + k_P} - r^2} \]
Simulations

\[ \mathcal{O} \xrightarrow{u} X \xrightarrow{k} \emptyset \]

\[ \dot{z} = X - r \]

\[ u = h[-k_P(X - r) - k_I \dot{z}] \]

\[ \langle X \rangle \quad k_P = 0.1 \quad \langle X \rangle \quad k_P = 2.0 \]
A Class of Network Structures

PI Control in the literature
- Alon’s chapter on chemotaxis
- Napp, Burden and Klavins: Control of Stochastic Robotics
- Two component systems in general
A Class of Network Structures

A Stochastic Hybrid System … with closed moment dynamics.

Gene expression: (Discrete) \[ \emptyset \xrightarrow{u_i(X,Z)} X_i \xrightarrow{\beta_i} \emptyset \]

Regulation: \[ u_i(X, Z) = \gamma_i Z_i - \sum_{j=1}^{n} k_{ij} X_j \]

Integration: (Continuous) \[ \dot{Z}_i = r_i - X_i \]
Moments

Group the means and moments into vectors and matrices

\[
\mu = \begin{pmatrix} \mu_X \\ \mu_Z \end{pmatrix} \triangleq \begin{pmatrix} \langle X \rangle \\ \langle Z \rangle \end{pmatrix} \quad \text{and} \quad M \triangleq \begin{pmatrix} \langle XX^T \rangle & \langle XZ^T \rangle \\ \langle ZX^T \rangle & \langle ZZ^T \rangle \end{pmatrix}
\]

\[
\kappa = \begin{pmatrix} \kappa_{XX^T} & \kappa_{XZ^T} \\ \kappa_{ZX^T} & \kappa_{ZZ^T} \end{pmatrix} \triangleq M - \mu \mu^T
\]

Group the parameters

\[
P \triangleq \text{diag}(\beta_1, \ldots, \beta_n) \quad \text{Degradation}
\]

\[
\Gamma \triangleq \text{diag}(\gamma_1, \ldots, \gamma_n) \quad \text{Integrator gain (tunable)}
\]

\[
K \triangleq \{k_{ij}\} \quad \text{Network (tunable)}
\]

\[
r \triangleq \begin{pmatrix} r_1 \\ \vdots \\ r_n \end{pmatrix}^T \quad \text{Reference inputs}
\]
Moment Dynamics

Use the extended generator to get mean dynamics

\[
\frac{d}{dt} \begin{pmatrix} \mu_X \\ \mu_Z \end{pmatrix} = \begin{pmatrix} -P - K & \Gamma \\ -I & 0 \end{pmatrix} \begin{pmatrix} \mu_X \\ \mu_Z \end{pmatrix} + \begin{pmatrix} 0 \\ I \end{pmatrix} r
\]

\[\dot{\mu} = A\mu + Br\]

And the second moment dynamics

\[
\dot{M} = AM + MA^T + C(\mu)
\]

\[C(\mu) = \begin{pmatrix}
\text{diag}(\Gamma \mu_Z + (P - K)\mu_X) & \mu_X r^T \\
\mu_Z r^T + r\mu_Z & \mu_Z r^T + r\mu_Z
\end{pmatrix}
\]
Properties

\[ \dot{\mu} = A\mu + Br \]
\[ \dot{M} = AM + M A^T + C(\mu) \]

**Theorem 1:** The network converges in mean and variance if and only if \( A \) is Hurwitz.

**Theorem 2:** The unique steady state mean \( \mu_x^* \) is \( r \) and is insensitive to \( K, \Gamma, \) and \( P \).

**Theorem 3:** The steady state covariance matrix in \( X \) can be placed arbitrarily. That is, if \( W \) is positive definite, then \( K \) and \( \Gamma \) can be found so that in steady state \( \kappa_{XX}^* = W \).
Example: One Gene

c) Proportional/integral control

\[ \begin{align*}
\emptyset & \xrightarrow{\gamma Z - kX} \beta X & \emptyset \\
\dot{Z} &= r - X
\end{align*} \]

\[ \mu^* = r \quad \text{and} \quad \kappa^* = \frac{r \beta}{k + \beta} \]

Mean and variance independently tunable
Example: Two Genes

Specification

\[ \mu^* = r = \begin{pmatrix} 10 \\ 20 \end{pmatrix} \]

\[ \kappa XX^T = \begin{pmatrix} 1 & 2 \\ 2 & 10 \end{pmatrix} \]
Example: Excitable Oscillator

Ensemble dynamics: Damped oscillator (-a_i+b_i j, -a_i-b_i j).

Stochastic dynamics: Sloppy oscillations with specific means.
Example #2

\[
\emptyset \xrightarrow{k_1} X \xleftrightarrow{u} Y \xrightarrow{k_3} \emptyset
\]

\[
k_4 \downarrow \xrightarrow{k_2} Y \xrightarrow{k_3} \emptyset
\]

Full state feedback controller with an integrator

\[
\dot{z} = Y - r
\]

\[
u = h\left[-k_{P_x}X - k_{P_y}Y - z\right]
\]

\[
k_{P_x} < 0, \quad k_{P_y} > 0
\]
Example #2

\[ \emptyset \xrightarrow{k_1} X \xleftarrow{u} Y \xrightarrow{k_3} \emptyset \]

\[ k_{P_x} = -0.01, \ k_{P_y} = 0.01 \]

\[ k_{P_x} = -1, \ k_{P_y} = 1 \]
Open Moments

\[ \emptyset \xrightarrow{k_1} X \xrightarrow{u} Y \xrightarrow{k_3} \emptyset \]
\[ \quad \quad \downarrow k_2 \quad \quad k_4 \quad \quad \emptyset \]

Steady State Moment Equations Give

\[ \langle \dot{Z} \rangle = \langle Y \rangle - r = 0 \neq \langle Y \rangle^* = r \]

Example Second Moment

\[ \frac{d}{dt} \langle YZ \rangle = -r \langle Y \rangle + \langle Y^2 \rangle - (k_2 + k_3) \langle Z \rangle - k_{px} \langle X^2 Z \rangle - k_{py} \langle XY Z \rangle - k_r \langle XZ^2 \rangle \]

Idea: Approximate Higher Order Moments? \( \Rightarrow \) No general results.
Proving Convergence

\[ \langle \dot{Z} \rangle = \langle X \rangle - r = 0 \Rightarrow \langle X \rangle^* = r \]  

WHEN THE SYSTEM IS ERGODIC

If the moments are closed, you can check for a stable steady state or just reason about the mean and variance.

If not, some other argument must be used.
Lyapunov Criterion for Markov Processes

**Theorem (Meyn):** If for some compact region $C$ and positive constant $\varepsilon$, there exists a positive radially unbounded function $V(q,x)$ such that

$$L V(q,x) \leq -\varepsilon \quad \forall (q,x) \notin C$$

then the process is ergodic.

I.e., the expected value of $V$ decreases outside of $C$.

Can only reasonably expect to achieve $r$ in $[y_{\text{min}}, y_{\text{min}}]$. 

Integral Control Works for any SCRN

**Theorem (Napp and Klavins):** Suppose steady state distributions $p_{min}$ and $p_{max}$ for constant inputs $u_{min}$ and $u_{max}$ respectively and suppose that

1. $\dot{z} = Y - r$;

2. $u = h[-k_i z]$ with $k_i > 0$

3. $p_{min} \leq r \leq p_{max}$.

Then $\langle Y \rangle \to r$ with finite variance.

$$\tilde{V}(q, x) = \begin{cases} 
-x + c^-(q), & x \leq x_{min} \\
 f_{smooth}(q, x), & x_{min} < x \leq x_{max} \\
x + c^+(q), & x < x_{max}
\end{cases}$$

Napp&Klavins, 2010
Outline

- Example Experimental Systems
- What Stochasticity Can Do
- Analytical Approaches
  - The Master Equation
  - Moment Dynamics
- Simulation Based Approaches
  - Simulation Methods
  - Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic

Software Tools
Outline

• Example Experimental Systems

• What Stochasticity Can Do

• Analytical Approaches
  • The Master Equation
  • Moment Dynamics

• Simulation Based Approaches
  • Simulation Methods

• Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic
Software Tools
Data from Simulations and Experiments

What really happens in the cell

A simple model

What about:
- A complex model?
- A refinement?
- An implementation?
- ...

Plan: Reason about systems based on the data they produce.
Simulation Approaches

The Stochastic Simulation Algorithm (Gillespie’s SSA)

Next reaction

\[ P[t, j] = k_{i,j}e^{-K_i t} \]

\[ p_{i,j} = \int_0^\infty k_{i,j}e^{-K_i t} dt = \frac{k_{i,j}}{K_i} \]

Time of the next reaction

\[ \int_0^t K_i e^{-K_i \tau} d\tau = 1 - e^{-K_i t} = r \in [0, 1] \]

\[ \sum_{j=1}^N k_{i,j}e^{-K_i t} = K_i e^{-K_i t} \]

1. Choose an initial condition \( v \) equal to some vector of the copy numbers of the species in the reaction network.
2. Set \( t = 0 \).
3. For each reaction \( \rho \) applicable in \( v \), determine the rate \( k_\rho \).
4. Choose the next reaction via Equation 8.6.
5. Choose the \( \Delta t \) via Equation 8.7 and set \( t \) to \( t + \Delta t \).
Simulation Approaches

Plain Old Euler Integration

- With mixed discrete / continuous systems, the SSA doesn’t directly work.
- And there is diminishing return for systems with many reactions.

Choose a timestep $\delta$ such that $\delta \lambda_{\text{max}} < 1$ for the largest rate $\lambda_{\text{max}}$ in your system.

- Enabled reactions $1, \ldots, N(t)$ with rates $\lambda_1, \ldots, \lambda_{N(t)}$.
- $\mu_i = \sum_{j < i} \lambda_i$.
- Choose $r \in [0, 1]$.
- Fire the reaction $i$ such that $\mu_i$ is the largest and exists. Otherwise, do nothing.
- $x(t + \delta) = x(t) + \delta f(x, q)$.
- $t \to t + \delta$. 
gro Simulations

\[ P = \{ \]
\[ \text{guard}_1: \text{command}_1 \]
\[ \ldots \]
\[ \text{guard}_n: \text{command}_n \]
\[ \} \]

- Growth is continuous.
- Signaling is continuous (finite element sim).
- Physics via Chipmunk (which takes \( dt \) as an argument at each step).
- Guards may have \( \text{rand}(0,1) < 0.25 \) evaluated at each iteration.
Outline

- Example Experimental Systems
- What Stochasticity Can Do
- Analytical Approaches
  - The Master Equation
  - Moment Dynamics
- Simulation Based Approaches
  - Simulation Methods
- Approximate Abstraction/Refinement

Running Example: Control of Gene Expression

Example: Arithmetic

Software Tools
In computer science, non-determinism has nice definitions for abstraction, refinement, implementation, simulation, etc.

For stochastic processes, what does it mean for one process to be an abstraction of another? A refinement? A coarse graining?

Approximate Bisimulation: Turns bisimulation into a metric on processes. Distance zero means bismilar. Distance epsilon means close.

\[ W(P_1, P_3) > W(P_2, P_3) \]
Comparing Stochastic Behaviors

Let $f$ be a metric on the space of trajectories $\Omega$.

For any two probability distributions $P_1$ and $P_2$ on $\Omega$, the Wasserstein Metric is defined by

$$W(P_1, P_2) = \inf_{Q \in J(P_1, P_2)} \int_{\Omega \times \Omega} f(\omega, \eta) dQ(\omega, \eta).$$

When $\Omega$ is finite, finding $W$ amounts to solving the linear program

Minimize $\sum_i \sum_j f(\omega_i, \eta_j) Q_{i,j}$

Subject to $\sum_j Q_{i,j} = \frac{1}{n}$

$\sum_i Q_{i,j} = \frac{1}{n}$

$Q_{i,j} \geq 0$

Hard to compute!

Example: Abstracting Gene Expression

[k_1]


Deleted Slides

• Unpublished data and examples deleted.
Send klavins@uw.edu the email address associated with your Google account so that you may download gro!

- Mac OS X 10.5.8 and up
- Windows 7 + Cygwin
Outline

• Example Experimental Systems
• What Stochasticity Can Do
• Analytical Approaches
  • The Master Equation
  • Moment Dynamics
• Simulation Based Approaches
  • Simulation Methods
• Approximate Abstraction/Refinement

• Done!