CPIB SUMMER SCHOOL 2011: INTRODUCTION TO BIOLOGICAL MODELLING

Lecture 3.2
Stochastic Models

Nick Monk

Deterministic vs. Stochastic Models

Deterministic models (differential equations):

- the state is represented by **continuous variables**,
- reactions/interactions are represented as **continuous processes** (production, decay, movement,…).

This is appropriate when the numbers of each species and the frequency of reaction events are “sufficiently large”.

This is not always the case.

e.g. transcription (1 or 2 copies of each gene); mRNA-protein networks in prokaryotes; low-density populations, …

Reaction Rates vs. Events

In deterministic models (differential equations) each process (e.g. transcription, degradation) has a **rate** (which is typically a function of the state).

e.g. Rate of transcription = $\frac{A[TF]}{h+[TF]}$

This gives a measure of how frequently each type of reaction is expected to occur (for a given state).

Alternatively, we can develop models centred on **individual reaction events** (e.g. the production of a single mRNA molecule, or the binding of two protein molecules).

Reaction Rates vs. Events

In a typical biological system, reaction events do not occur at regular intervals.

e.g. in a “well-mixed” solution, molecular motion can be represented as Brownian motion; bimolecular reactions occur when two reactant molecules approach closely enough (and in the correct orientation).

e.g. transcription depends on the assembly of a polymerase complex and then sequential addition of nucleotides.

The timing of these events is essentially **stochastic** rather than deterministic.

Deterministic vs. Stochastic Models

Deterministic	Stochastic
Ordinary differential equations	Continuous time Markov chains
Concentrations of molecules	Numbers of molecules
Future is “predictable” given present knowledge	Includes randomness; every simulation is different
Wide range of techniques available for analysis	Not as many techniques for analysis; often rely on simulations
Good for large number of individuals; qualitative analysis	Better for simulating dynamics with small numbers of molecules
Represents population average	Represents population variability

Stochastic kinetics

- Assume homogeneity:
  - P(molecule in volume δV) is equal for each δV on the timescale of the chemical reactions that change the state.
  - In other words, we assume that the “reaction mixture” (i.e. the inside of the cell) is well-mixed. This may be questionable.
- Then the state of a cell can be represented by a vector $\mathbf{n}$, where

$$\mathbf{n} = [n_1, n_2, \dots n_N],$$

and n_i is the (integer) number of molecules of species i in the cell ($i = 1, 2, \dots, N$). Note that we could equally well use concentrations rather than numbers.

The stoichiometric matrix

- Elementary reaction j changes the state: $n_i \to n_i + \nu_{ij}$, where ν_{ij} is the **stoichiometric matrix**.
- The state vector $\mathbf{n}$ changes by ν_j (the state change vector) when reaction j occurs:
$$\mathbf{n} \to \mathbf{n} + \nu_j, \quad \nu_j = \left(\nu_{1j}, \nu_{2j}, \ldots \nu_{Nj}\right).$$
- Note that the elements of the state change vector are integers (ν_{ij} is just the change in the number of molecules of type i when reaction j occurs).

The stoichiometric matrix

For example, consider transcription of a gene, translation of the mRNA, reversible dimerisation of the resulting protein, and degradation of each species.

State: $\mathbf{n} = [M, P, D]$ = [mRNA, protein, dimer]

Reactions:		*State change vectors:*
1. Transcription:	$\varnothing \to M$	$\nu_1 = [1, 0, 0]$
2. Translation:	$\varnothing \to P$	$\nu_2 = [0, 1, 0]$
3. Dimerisation:	$P + P \to D$	$\nu_3 = [0, -2, 1]$
4. Dissociation:	$D \to P + P$	$\nu_4 = [0, 2, -1]$
5. M degradation:	$M \to \varnothing$	$\nu_5 = [-1, 0, 0]$
6. P degradation:	$P \to \varnothing$	$\nu_6 = [0, -1, 0]$
7. D degradation:	$D \to \varnothing$	$\nu_7 = [0, 0, -1]$

The Master Equation

- Let the **rate** of reaction j be $r_j(\mathbf{n})$.

  In other words, the probability that reaction j occurs in a small time interval Δt is $r_j(\mathbf{n})\,\Delta t$.
- Note that this probability depends on the network state.
- Note also that it is assumed that the reaction occurs instantaneously.
- Let $P(\mathbf{n},t)$ be the probability that that the network is in state $\mathbf{n}$ at time t. The **master equation** describes how this probability changes in time, given a set of elementary reactions.

The Master Equation

- P_{arrive} = Prob. that network arrives in state $\mathbf{n}$ in time $[t, t+\Delta t]$.
- P_{leave} = Prob. that network leaves state $\mathbf{n}$ in time $[t, t+\Delta t]$.
- P_{stay} = Prob. that network stays in state $\mathbf{n}$ in time $[t, t+\Delta t]$.

$$P_{\text{arrive}} = \Delta t \sum_{j=1}^{M} r_j(\mathbf{n}-\nu_j) P(\mathbf{n}-\nu_j, t),$$

$$P_{\text{leave}} = \Delta t \sum_{j=1}^{M} r_j(\mathbf{n}) P(\mathbf{n}, t),$$

$$P_{\text{stay}} = \left[1 - \Delta t \sum_{j=1}^{M} r_j(\mathbf{n})\right] P(\mathbf{n}, t).$$

The Master Equation

$$P(\mathbf{n}, t+\Delta t) = P_{\text{stay}} + P_{\text{arrive}}$$

$$= \left[1 - \Delta t \sum_{j=1}^{M} r_j(\mathbf{n})\right] P(\mathbf{n}, t) + \Delta t \sum_{j=1}^{M} r_j(\mathbf{n}-\nu_j) P(\mathbf{n}-\nu_j, t)$$

$$P(\mathbf{n}, t+\Delta t) - P(\mathbf{n}, t) = \Delta t \sum_{j=1}^{M} \left[r_j(\mathbf{n}-\nu_j) P(\mathbf{n}-\nu_j) - r_j(\mathbf{n}) P(\mathbf{n}) \right].$$

In the limit $\Delta t \to 0$, we get

$$\boxed{\frac{dP(\mathbf{n}, t)}{dt} = \sum_{j=1}^{M} \left[r_j(\mathbf{n}-\nu_j) P(\mathbf{n}-\nu_j) - r_j(\mathbf{n}) P(\mathbf{n}) \right].}$$

Simulating paths of the Master Equation

- The master equation encodes a continuous time, discrete state Markov process.
- The master equation is hard to solve, so often rely on simulating “paths” (i.e. stochastic realisations of the evolution of the state via elementary reactions that occur with probabilities in accordance with the master equation).
- First determine the probabilities of reactions occurring.
- Determine from this the next reaction time (drawn from an exponential distribution).
- Choose the reaction type to implement, according to fractional rates.

Simulating paths of the Master Equation

- Prob that some reaction will occur in time Δt = $\Delta t \sum_{j=1}^{M} r_j(\mathbf{n})$

$$\equiv \Delta t \, r_0 .$$

- Let $p(t)$ = prob that the system has not left state $\mathbf{n}$ at time t given that it was at $\mathbf{n}$ at $t = 0$.
- Then

$$p(t+\Delta t) - p(t) = -r_0 p(t) \Delta t$$

- In the limit: $\frac{dp}{dt} = -r_0 p(t) \Rightarrow p(t) = \exp(-r_0 t).$
- Prob that a reaction has occurred at time t is

$$F(t) = 1 - p(t) = 1 - \exp(-r_0 t).$$

The stochastic simulation algorithm (SSA)

- Prob that a reaction has occurred at time t is

$$F(t) = 1 - p(t) = 1 - \exp(-r_0 t).$$

- Therefore, the system residence time in state $\mathbf{n}$ (i.e. the time to the next reaction) is exponentially distributed with average = $1/r_0$.
- The probability that the first reaction to occur is of type j is its relative contribution to the total rate, i.e. $r_j(\mathbf{n})/r_0$.
- Algorithm (Doob, 1945; Gillespie, 1976,77):
  - Pick the next reaction time from an exponential distribution,
  - Choose a reaction type according to the fractional rates.

The stochastic simulation algorithm (SSA)

- Algorithm (Doob, 1945; Gillespie, 1976,77):
  - Pick the next reaction time from an exponential distribution,
  - Choose a reaction type according to the fractional rates.
- In practice:
  - Draw two random numbers s_1 and s_2 from the uniform distribution in the unit interval.
  - Reaction time = $\frac{1}{r_0(\mathbf{n})} \ln\left(\frac{1}{s_1}\right)$
  - Reaction type j is the smallest integer such that

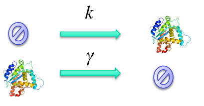
$$\sum_{j'=1}^{j} r_{j'}(\mathbf{n}) > s_2 r_0(\mathbf{n}).$$

The stochastic simulation algorithm (SSA)

1. Specify reaction stoichiometry (state update vectors $\boldsymbol{v}_j$), reaction propensities and simulation time.
2. Specify initial state $\mathbf{n} = \mathbf{n}_0$ and time $t = t_0$.
3. Calculate the sum of all reaction propensities [= $r_0(\mathbf{n})$].
4. Generate two random numbers s_1 and s_2 from the uniform distribution in the unit interval.
    - Time to next reaction $\theta = \frac{1}{r_0(\mathbf{n})} \ln\left(\frac{1}{s_1}\right)$
    - Reaction type j is the smallest integer such that $\sum_{j'=1}^{j} r_{j'}(\mathbf{n}) > s_2 r_0(\mathbf{n})$.
5. Update time $t \rightarrow t + \theta$.
6. Update state $\mathbf{n} \rightarrow \mathbf{n} + \boldsymbol{v}_j$ according to reaction j.
7. Recalculate propensities.
8. Go to 3.

Simple Example

- A constitutively expressed protein
- Two types of reaction:

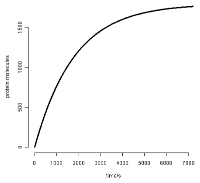


Protein synthesis (constant rate)

Protein degradation

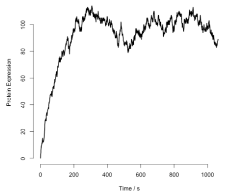
ODE Model:

$$\frac{dP}{dt} = k - \gamma P$$

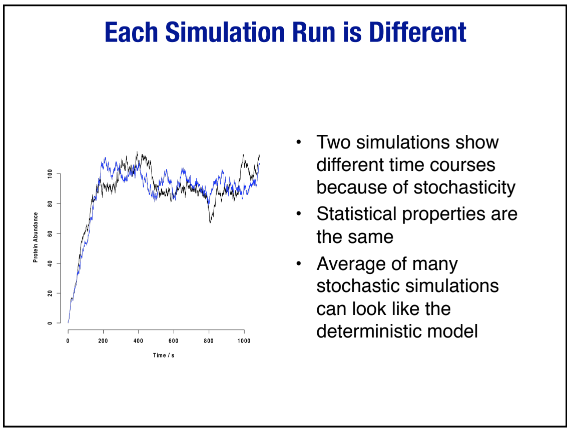


Parameters:
- $k = 1\ \mathrm{Ms}^{-1}$
- $\gamma = 1/1800\ \mathrm{s}^{-1}$
- $P_0 = 0$

Stochasic Simulation

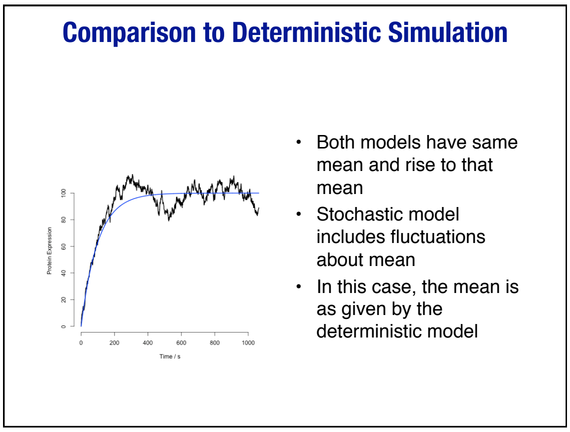


- Parameters:
    - $k = 1\ s^{-1}$
    - $\gamma = 0.01\ s^{-1}$
    - $P_0 = 0$
- Random fluctuations in numbers of molecules



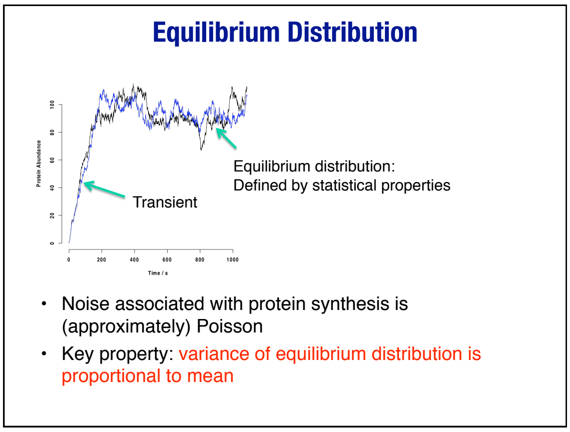
Each Simulation Run is Different

- Two simulations show different time courses because of stochasticity
- Statistical properties are the same
- Average of many stochastic simulations can look like the deterministic model



Comparison to Deterministic Simulation

- Both models have same mean and rise to that mean
- Stochastic model includes fluctuations about mean
- In this case, the mean is as given by the deterministic model

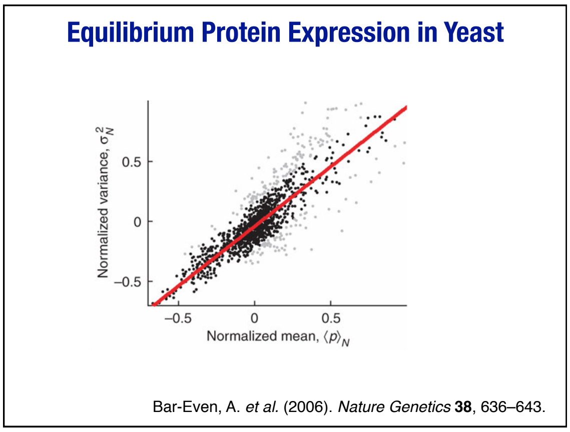


Equilibrium Distribution

Equilibrium distribution:
Defined by statistical properties

Transient

- Noise associated with protein synthesis is (approximately) Poisson
- Key property: variance of equilibrium distribution is proportional to mean



Equilibrium Protein Expression in Yeast

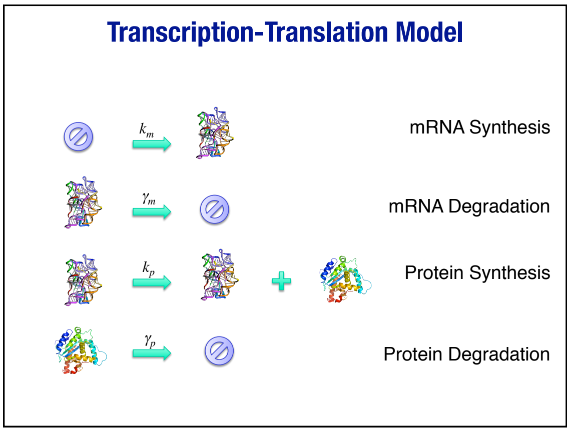
Bar-Even, A. *et al.* (2006). *Nature Genetics* **38**, 636–643.

How Important is Poisson Noise?

Percentage error in mean (coefficient of variability) proportional to 1/sqrt(mean).

Stochasticity is particularly important in systems with low numbers of reactants

Number of molecules	**Molar (bacterial cell)**	**Processes**	**c.v.**
1	1nM	Genetic	100%
10	10nM	↑	32%
100	100nM	↕	10%
1000	1μM	↓	3%
10^6	1mM	Metabolic	0.1%



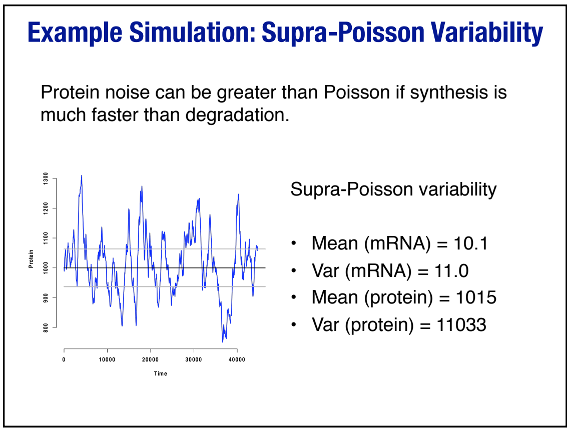
Transcription-Translation Model

mRNA Synthesis

mRNA Degradation

Protein Synthesis

Protein Degradation



Example Simulation: Supra-Poisson Variability

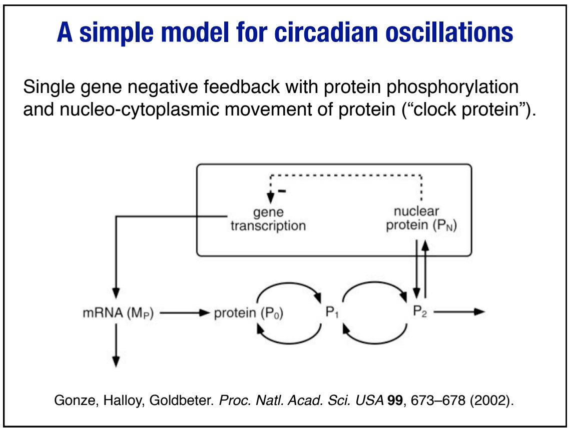
Protein noise can be greater than Poisson if synthesis is much faster than degradation.

Supra-Poisson variability

- Mean (mRNA) = 10.1
- Var (mRNA) = 11.0
- Mean (protein) = 1015
- Var (protein) = 11033

Examples of Behaviour

- As system volume gets large, mean of stochastic model can behave like deterministic model
- But individual realizations can be quite different!
  - Oscillations in stochastic model not seen in deterministic model
  - Mean of stochastic system different from deterministic model
  - Stochastic switching between (quasi) steady states

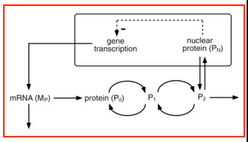


A simple model for circadian oscillations

Single gene negative feedback with protein phosphorylation and nucleo-cytoplasmic movement of protein (“clock protein”).

Gonze, Halloy, Goldbeter. *Proc. Natl. Acad. Sci. USA* **99**, 673–678 (2002).

A simple model for circadian oscillations



$$\frac{dM_P}{dt} = v_s \frac{K_I^n}{K_I^n + P_N^n} - v_m \frac{M_P}{K_m + M_P}$$

$$\frac{dP_0}{dt} = k_s M_P - v_1 \frac{P_0}{K_1 + P_0} + v_2 \frac{P_1}{K_2 + P_1}$$

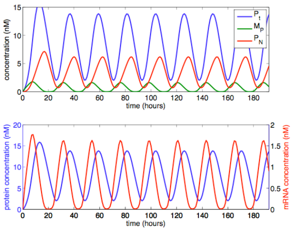
$$\frac{dP_1}{dt} = v_1 \frac{P_0}{K_1 + P_0} - v_2 \frac{P_1}{K_2 + P_1} - v_3 \frac{P_1}{K_3 + P_1} + v_4 \frac{P_2}{K_4 + P_2}$$

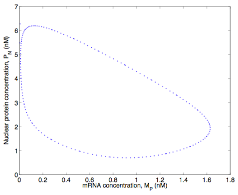
$$\frac{dP_2}{dt} = v_3 \frac{P_1}{K_3 + P_1} - v_4 \frac{P_2}{K_4 + P_2} - v_d \frac{P_2}{K_d + P_2} - k_1 P_2 + k_2 P_N$$

$$\frac{dP_N}{dt} = k_1 P_2 - k_2 P_N$$

Note the nonlinear functional forms for degradation of mRNA and P_2.

Numerical simulation of ODE model





Elementary reactions: transcription

Need to represent binding of transcription factor P_N to multiple sites on the DNA. These are the elementary reactions that we need for the SSA.

Consider 4 binding sites. Transcription is blocked only if all 4 sites are bound by P_N.

Reaction number	Reaction step	Probability of reaction
1	$G + P_N \xrightarrow{a_1} GP_N$	$w_1 = a_1 \times G \times P_N / \Omega$
2	$GP_N \xrightarrow{d_1} G + P_N$	$w_2 = d_1 \times GP_N$
3	$GP_N + P_N \xrightarrow{a_2} GP_{N2}$	$w_3 = a_2 \times GP_N \times P_N / \Omega$
4	$GP_{N2} \xrightarrow{d_2} GP_N + P_N$	$w_4 = d_2 \times GP_{N2}$
5	$GP_{N2} + P_N \xrightarrow{a_3} GP_{N3}$	$w_5 = a_3 \times GP_{N2} \times P_N / \Omega$
6	$GP_{N3} \xrightarrow{d_3} GP_{N2} + P_{N3}$	$w_6 = d_3 \times GP_{N3}$
7	$GP_{N3} + P_N \xrightarrow{a_4} GP_{N4}$	$w_7 = a_4 \times GP_{N3} \times P_N / \Omega$
8	$GP_{N4} \xrightarrow{d_4} GP_{N3} + P_N$	$w_8 = d_4 \times GP_{N4}$
9	$[G, GP_N, GP_{N2}, GP_{N3}] \xrightarrow{v_s} M_P$	$w_9 = v_s \times (G + GP_N + GP_{N2} + GP_{N3})$

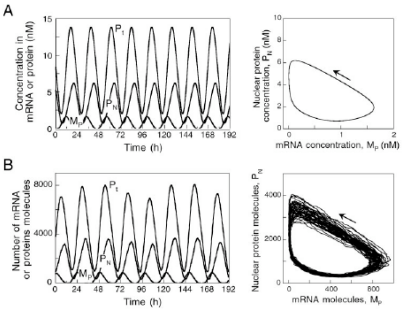
Elementary reactions: kinetics

10	$M_P + E_m \xrightarrow{k_{m1}} C_m$	$w_{10} = k_{m1} \times M_P \times E_m / \Omega$
11	$C_m \xrightarrow{k_{m2}} M_P + E_m$	$w_{11} = k_{m2} \times C_m$
12	$C_m \xrightarrow{k_{m3}} E_m$	$w_{12} = k_{m3} \times C_m$
13	$M_P \xrightarrow{k_s} M_P + P_0$	$w_{13} = k_s \times M_P$
14	$P_0 + E_1 \xrightarrow{k_{11}} C_1$	$w_{14} = k_{11} \times P_0 \times E_1 / \Omega$
15	$C_1 \xrightarrow{k_{12}} P_0 + E_1$	$w_{15} = k_{12} \times C_1$
16	$C_1 \xrightarrow{k_{13}} P_1 + E_1$	$w_{16} = k_{13} \times C_1$
17	$P_1 + E_2 \xrightarrow{k_{21}} C_2$	$w_{17} = k_{21} \times P_1 \times E_2 / \Omega$
18	$C_2 \xrightarrow{k_{22}} P_1 + E_2$	$w_{18} = k_{22} \times C_2$
19	$C_2 \xrightarrow{k_{23}} P_0 + E_2$	$w_{19} = k_{23} \times C_2$
20	$P_1 + E_3 \xrightarrow{k_{31}} C_3$	$w_{20} = k_{31} \times P_1 \times E_3 / \Omega$
21	$C_3 \xrightarrow{k_{32}} P_1 + E_3$	$w_{21} = k_{32} \times C_3$
22	$C_3 \xrightarrow{k_{33}} P_2 + E_3$	$w_{22} = k_{33} \times C_3$
23	$P_2 + E_4 \xrightarrow{k_{41}} C_4$	$w_{23} = k_{41} \times P_2 \times E_4 / \Omega$
24	$C_4 \xrightarrow{k_{42}} P_2 + E_4$	$w_{24} = k_{42} \times C_4$
25	$C_4 \xrightarrow{k_{43}} P_1 + E_4$	$w_{25} = k_{43} \times C_4$
26	$P_2 + E_d \xrightarrow{k_{d1}} C_d$	$w_{26} = k_{d1} \times P_2 \times E_d / \Omega$
27	$C_d \xrightarrow{k_{d2}} P_2 + E_d$	$w_{27} = k_{d2} \times C_d$
28	$C_d \xrightarrow{k_{d3}} E_d$	$w_{28} = k_{d3} \times C_d$
29	$P_2 \xrightarrow{k_1} P_N$	$w_{29} = k_1 \times P_2$
30	$P_N \xrightarrow{k_2} P_2$	$w_{30} = k_2 \times P_N$

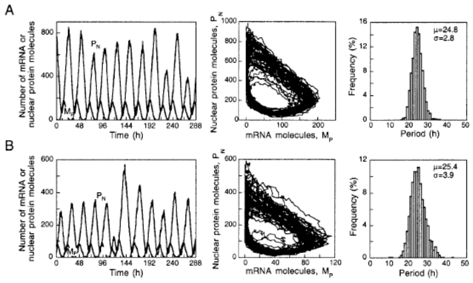
Varying the total number of molecules

- The probabilities are derived from the rates in the deterministic version of the model.
- The total number of molecules in the system can be varied using the parameter Ω.
- Large Ω corresponds to large numbers of molecules.
- Bimolecular reaction rates depend on Ω (they are concentration dependent — cf. ODE models).
- Check that the stochastic scheme gives solutions that resemble the deterministic simulations when Ω is large.
- Study the effect of reducing Ω.

Stochastic simulation (Ω = 500)

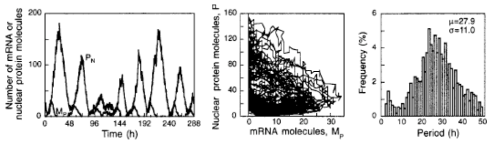


Stochastic simulation (Ω = 100, 50)



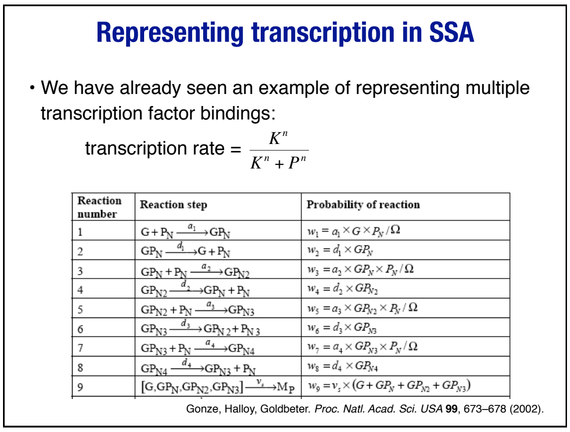
Stochastic simulation (Ω = 10)

- For small numbers of molecules, the limit cycle oscillations become very irregular.
- However, the period histogram shows that some periodicity is preserved.



Representing transcription in SSA

- The SSA assumes that all reactions are elementary (i.e. instantaneous mass-action kinetics).
- Regulated transcription is not an elementary reaction; how can we represent it?
  1. Model multiple transcription factor binding events as elementary reactions (but can we parameterise this?)
  2. Relax the requirement for only elementary reactions and use a nonlinear propensity function.



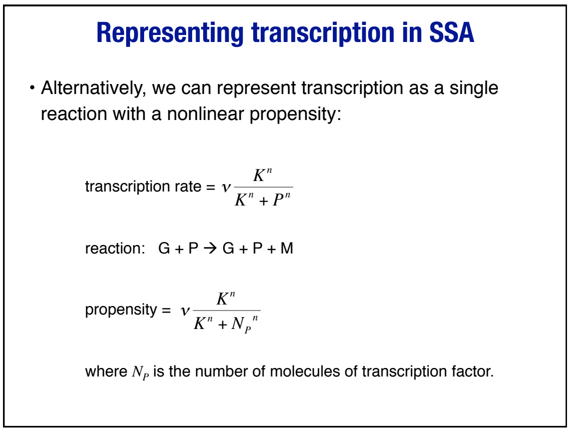
Representing transcription in SSA

- We have already seen an example of representing multiple transcription factor bindings:

$$\text{transcription rate} = \frac{K^n}{K^n + P^n}$$

Reaction number	Reaction step	Probability of reaction
1	$G + P_N \xrightarrow{a_1} GP_N$	$w_1 = a_1 \times G \times P_N / \Omega$
2	$GP_N \xrightarrow{d_1} G + P_N$	$w_2 = d_1 \times GP_N$
3	$GP_N + P_N \xrightarrow{a_2} GP_{N2}$	$w_3 = a_2 \times GP_N \times P_N / \Omega$
4	$GP_{N2} \xrightarrow{d_2} GP_N + P_N$	$w_4 = d_2 \times GP_{N2}$
5	$GP_{N2} + P_N \xrightarrow{a_3} GP_{N3}$	$w_5 = a_3 \times GP_{N2} \times P_N / \Omega$
6	$GP_{N3} \xrightarrow{d_3} GP_{N2} + P_{N3}$	$w_6 = d_3 \times GP_{N3}$
7	$GP_{N3} + P_N \xrightarrow{a_4} GP_{N4}$	$w_7 = a_4 \times GP_{N3} \times P_N / \Omega$
8	$GP_{N4} \xrightarrow{d_4} GP_{N3} + P_N$	$w_8 = d_4 \times GP_{N4}$
9	$[G, GP_N, GP_{N2}, GP_{N3}] \xrightarrow{v_s} M_P$	$w_9 = v_s \times (G + GP_N + GP_{N2} + GP_{N3})$

Gonze, Halloy, Goldbeter. *Proc. Natl. Acad. Sci. USA* **99**, 673–678 (2002).



Representing transcription in SSA

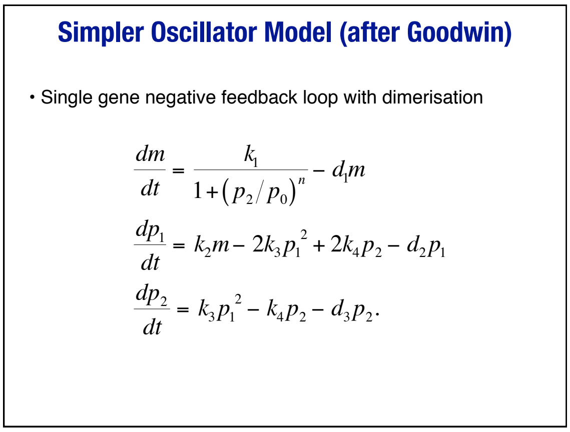
- Alternatively, we can represent transcription as a single reaction with a nonlinear propensity:

$$\text{transcription rate} = v\frac{K^n}{K^n + P^n}$$

reaction: G + P → G + P + M

$$\text{propensity} = v\frac{K^n}{K^n + N_P{}^n}$$

where N_P is the number of molecules of transcription factor.



Simpler Oscillator Model (after Goodwin)

- Single gene negative feedback loop with dimerisation

$$\frac{dm}{dt} = \frac{k_1}{1 + (p_2/p_0)^n} - d_1 m$$

$$\frac{dp_1}{dt} = k_2 m - 2k_3 p_1^2 + 2k_4 p_2 - d_2 p_1$$

$$\frac{dp_2}{dt} = k_3 p_1^2 - k_4 p_2 - d_3 p_2.$$

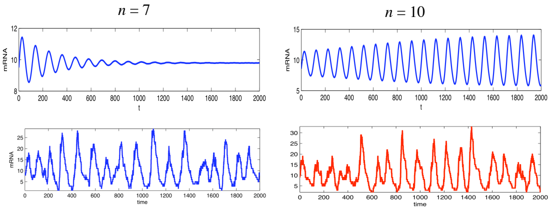
Simpler Oscillator Model (after Goodwin)

	Reaction	Rate
1. transcription	$\varnothing \to M$	$\frac{k_1}{1+(p_2/p_0)^n}$
2. M degradation	$M \to \varnothing$	$d_1 m$
3. translation	$\varnothing \to P_1$	$k_2 m$
4. P_1 dimerisation	$P_1 + P_1 \to P_2$	$k_3 p_1(p_1-1)$
5. P_1 degradation	$P_1 \to \varnothing$	$d_2 p_1$
6. P_2 dissociation	$P_2 \to P_1 + P_1$	$k_4 p_2$
7. P_2 degradation	$P_2 \to \varnothing$	$d_3 p_2$

Stoichiometric matrix: $$v = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & -1 & 2 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & -1 \end{pmatrix}$$

Simulations: Stochastic Resonance

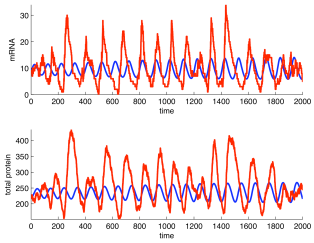
$$k_1 = 2, k_2 = 0.5, k_3 = 0.0001, k_4 = 0.001, d_1 = 0.03, d_2 = d_3 = 0.02, p_0 = 50$$



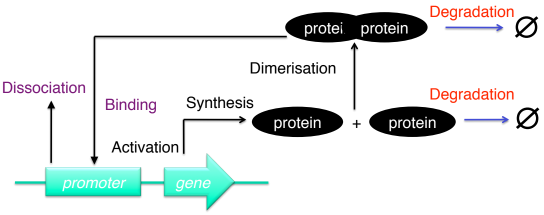
Stochastic model can show sustained oscillations even when the ODE model has damped oscillations

Simulations

Even when the ODE model has sustained oscillations ($n = 10$) the stochastic oscillations can have a significantly greater amplitude



Simple Genetic Switch Model



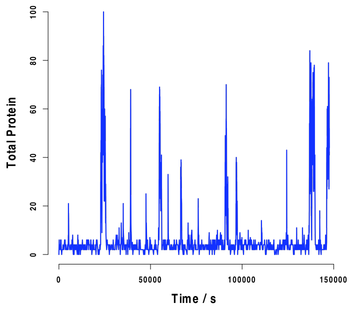
Elementary Reactions

Process	Reaction	Parameter
Basal synthesis	DNA → DNA + Prot	k_0
Dimerization	Prot + Prot → Dimer	k_{dim}
Degradation	Prot → ∅	μ
Degradation	Dimer → ∅	μ
Binding	DNA + Dimer → Complex	k_{ass}
Dissociation	Complex → DNA + Dimer	k_{dis}
Activated synthesis	Complex → Complex + Prot	k_1

Realistic Parameters (1 μm cell)

Parameter	Value	Comment
k_0	0.03 s^{-1}	Molecules per cell
k_{dim}	0.0001 s^{-1}	Diffusion limited
μ	0.01 s^{-1}	Rapidly turned over protein (monomer and dimer)
k_{ass}	0.01 s^{-1}	Diffusion limited: faster because of 1D diffusion
k_{dis}	0.08 s^{-1}	Binding site affinity is 8nM
k_1	1 s^{-1}	Transcription, translation and mRNA stability

Stochastic Simulation



ODE Model

$$\frac{dP_1}{dt} = k_0 D + k_1 C - k_{\dim} P_1^2 - \mu P_1$$

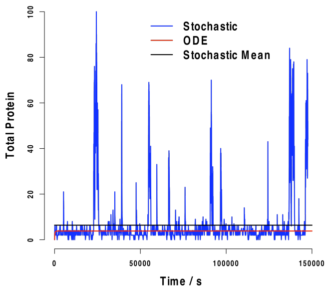
$$\frac{dP_2}{dt} = \frac{1}{2} k_{\dim} P_1^2 - \mu P_2 - k_{ass} D P_2 + k_{dis} C$$

$$\frac{dC}{dt} = k_{ass} D P_2 - k_{dis} C$$

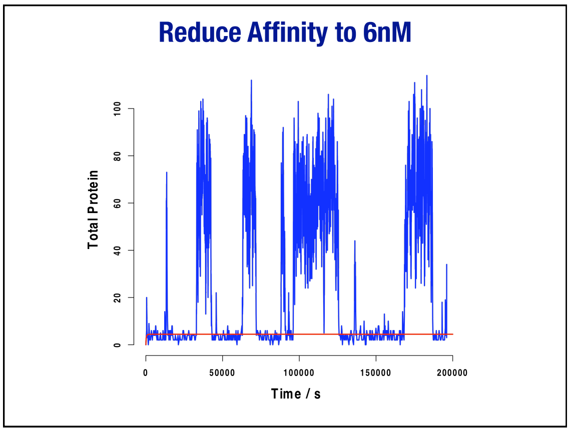
$$C + D = D_0$$

Stochastic vs. ODE Simulations

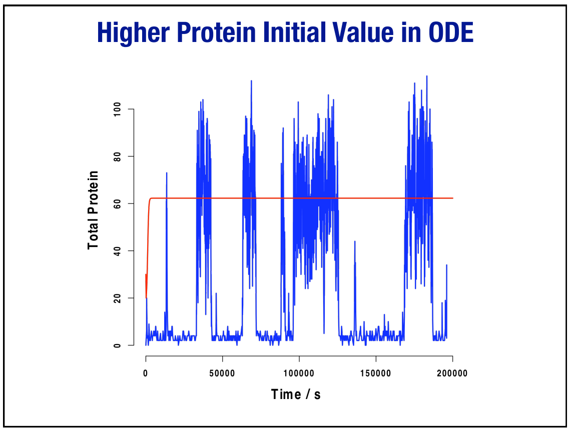
ODE steady state not necessarily equal to stochastic mean



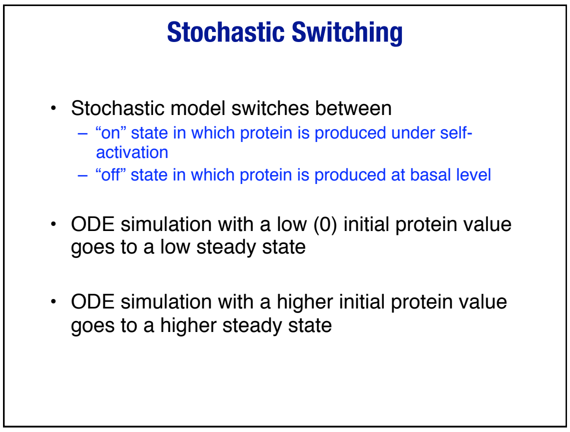
- ODE steady state = 3.81
- Stochastic mean = 6.65



Reduce Affinity to 6nM

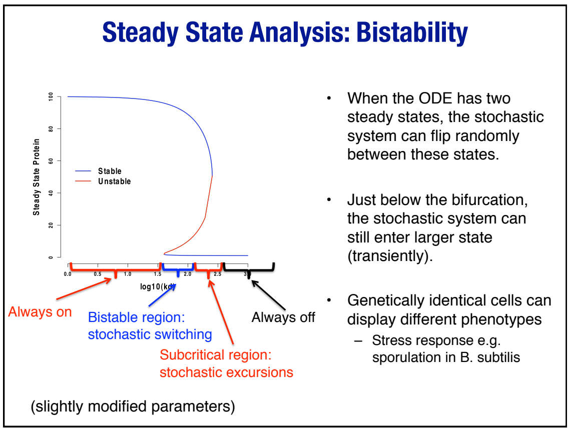


Higher Protein Initial Value in ODE



Stochastic Switching

- Stochastic model switches between
  - “on” state in which protein is produced under self-activation
  - “off” state in which protein is produced at basal level
- ODE simulation with a low (0) initial protein value goes to a low steady state
- ODE simulation with a higher initial protein value goes to a higher steady state



Steady State Analysis: Bistability

Always on

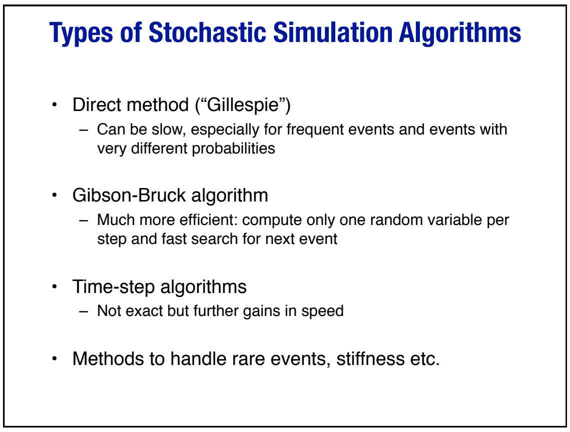
Bistable region: stochastic switching

Always off

Subcritical region: stochastic excursions

(slightly modified parameters)

- When the ODE has two steady states, the stochastic system can flip randomly between these states.
- Just below the bifurcation, the stochastic system can still enter larger state (transiently).
- Genetically identical cells can display different phenotypes
  - Stress response e.g. sporulation in B. subtilis



Types of Stochastic Simulation Algorithms

- Direct method (“Gillespie”)
  - Can be slow, especially for frequent events and events with very different probabilities
- Gibson-Bruck algorithm
  - Much more efficient: compute only one random variable per step and fast search for next event
- Time-step algorithms
  - Not exact but further gains in speed
- Methods to handle rare events, stiffness etc.