IV Master in Biophysics
Universidad Autónoma de Madrid
Oct 26 – Nov 8/2006

Juan F. Poyatos

Stochastic dynamics

Evolutionary Systems Biology Lab

JUAN F. POYATOS

SPANISH NATIONAL CANCER CENTRE

http://bioinfo.cnio.es/~jpoyatos/

Outlook

- WHAT is stochastic dynamics?

- WHY cellular stochastic dynamics?

- HOW do we deal with stochastic dynamics?

THEORY + EXPERIMENTS

day I


0010100
001011001
001110011011
111100
01001
01011
100001
100
1
100G0AT
ATT01GCA0A
A0T01
ATTCG
TATGGATGG
TTA

Stochastic motion

- motion generated by random forces, e.g., forces randomly applied in time

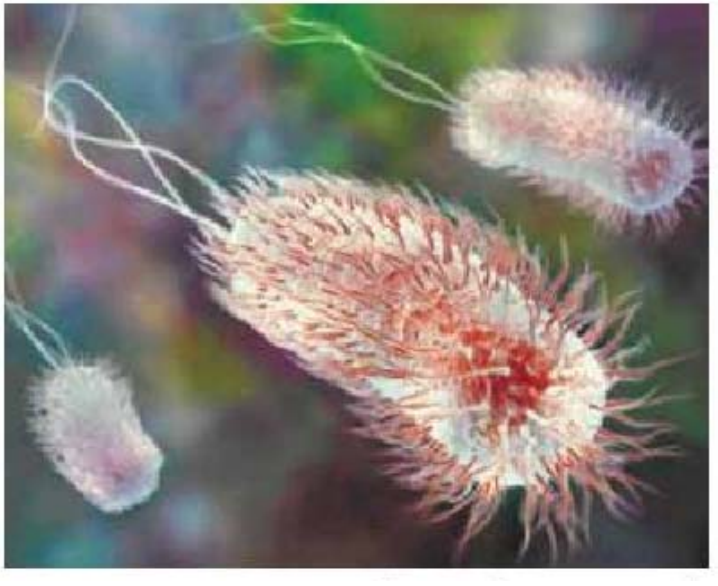
- to describe a stochastic system we need probabilities

- chemical systems are intrinsically stochastic (noisy), specially when a small pool of reactants is involved

is gene expression noisy?

- Many molecules that take part in gene expression (including DNA and important regulatory molecules such as the enzyme polymerase) act at extremely low intracellular concentrations (low copy numbers)

- Gene expression as a series of biochemical reactions experiences "surprising" things when one takes the discreteness of molecule number seriously



Escherichia Coli (E.coli) numbers

2μm long
1μm diameter

$V = \pi r^2 l = \pi/2\ 10^{-15}$ liters
[RNA Polymerase] ~ 100nM = 100 molecules
(1nM ~ 1 molecule)

Biochemical noise

-consider a simple gene expression system (unregulated gene)

$$\emptyset \xrightarrow{k} P, \quad \text{production}$$

$$P \xrightarrow{\delta} \emptyset, \quad \text{first order degradation}$$

a common approach is to describe these reactions by means of differential reaction-rate equations

$$\frac{d[P]}{dt} = k - \delta[P]$$

This approach assumes that the time evolution of such reaction is both continuous and deterministic

continuous? molecule number changes in discrete ways

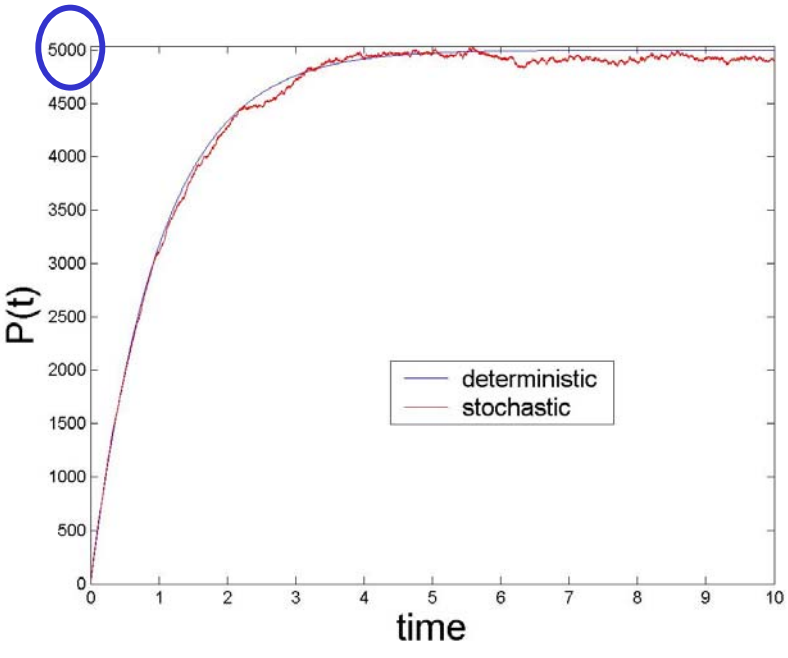
deterministic? impossible to predict the motion of (classical) molecules due to the ignorance of positions and velocities of all components of the system

however in many cases of course the time evolution of a chemically reacting system can, to a very acceptable degree of accuracy, be treated as a continuous, deterministic process

MATLAB code 1

```
% .. code1.m
% .. simple gene expression deterministic equations

clear all
k = 10;
delta = 1;

tspan = [0 10];
P0 = 0;
options = [];
[t P] = ode23(@code1equations,tspan,P0,options,k,delta);
```

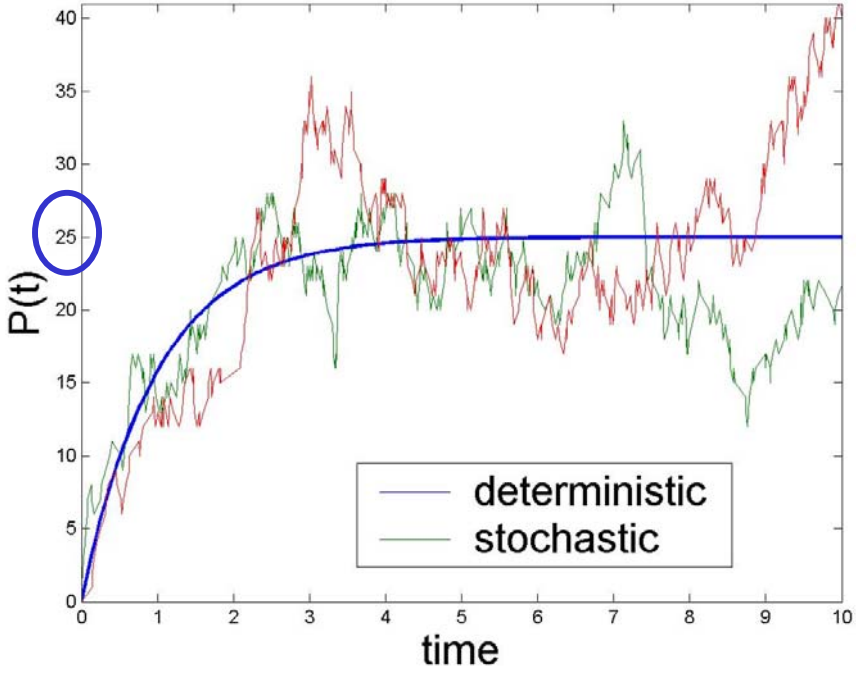
```
% .. code1equations.m
% .. rate equations for code1

function dPdt = code1equations(t,P,k,delta)

dPdt = [k - delta*P(1)];
```



large number of molecules
deterministic approximation works

small number of molecules
deterministic approximation fails

large protein fluctuations



Stochastic description of chemical reactions

Recall:

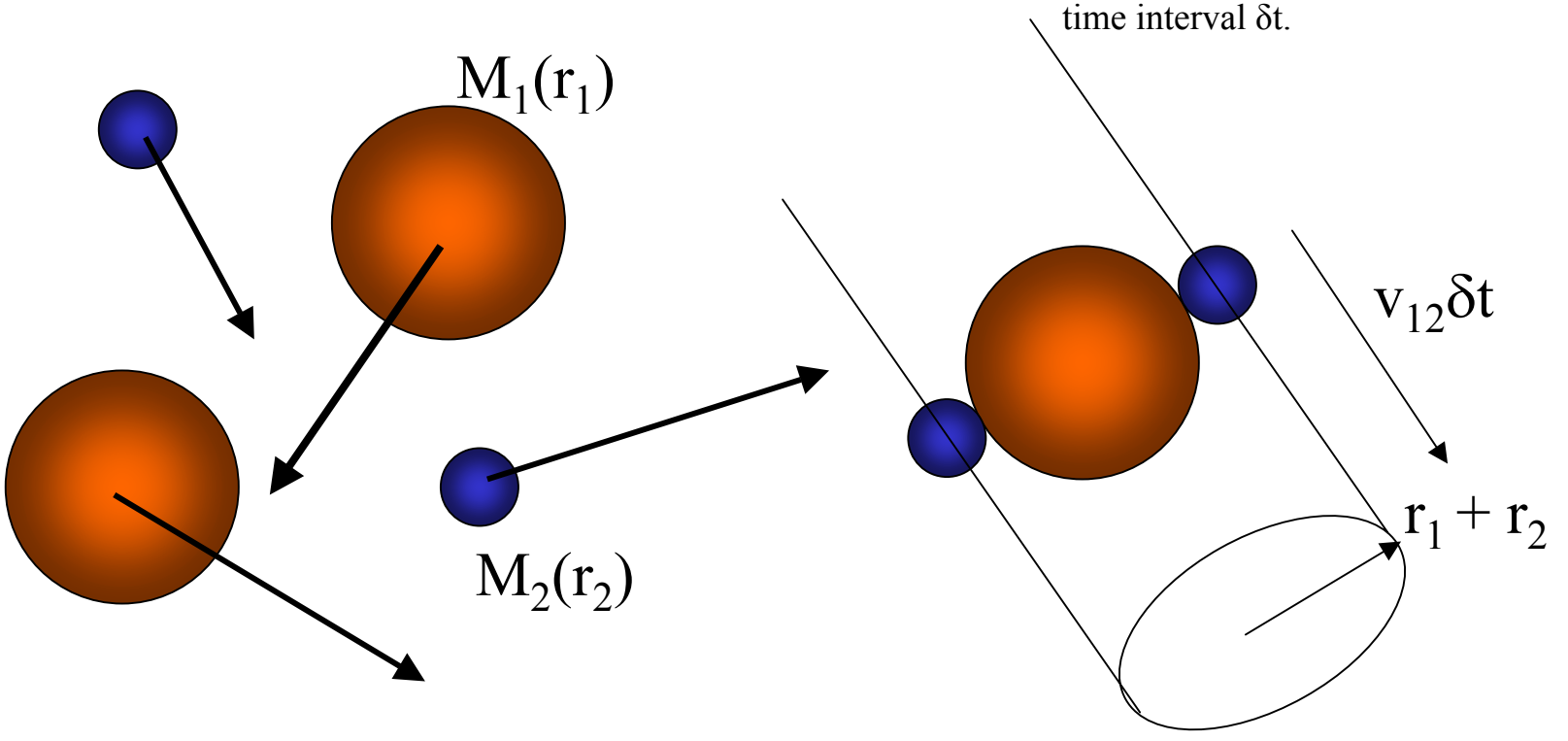
For a stochastic system it is not possible to determine exactly the state of the system at later times given its state at the current time.
We must thus deal with probabilities.

Basis of the stochastic formulation: a chemical reaction occurs when molecules collide in an appropiate way

- Molecular collisions: random microscopic events

Stochastic description of chemical reactions

V_{coll} – Collision volume. The molecules M_2 which are within collision volume will be hit by a particular molecule M_1 in the next time interval δt.



$$V_{coll} = v_{12}\delta t\ \pi\ (r_1 + r_2)^2$$

$$P(\text{a given } M_1 \text{ and } M_2 \text{ collide}) = \frac{\overline{v_{12}}\,\delta t\,\pi\,(r_1 + r_2)^2}{V}$$

$$P(\text{a } M_1 \text{ and } M_2 \text{ molecule collide}) = n_1\,n_2\,\frac{\overline{v_{12}}\,\delta t\,\pi\,(r_1 + r_2)^2}{V}$$

and finally

diffusion-limited R close to one always

$$P(\text{a } M_1 \text{ and } M_2 \text{ react}) = n_1\,n_2\,\boxed{\frac{\overline{v_{12}}\,R\,\pi\,(r_1 + r_2)^2}{V}}\,\delta t = n_1\,n_2\,r\,\delta t$$

probability that a given M_1 and M_2 react in unit time (**r**)

units of inverse time

this is the fundamental hypothesis from which we derive both the Master Equation and the Stochastic Simulation approaches.

The Master Equation

The stochastic framework considers the discrete number of molecules whose state changes probabilistically

Recall our previous simple gene expression model

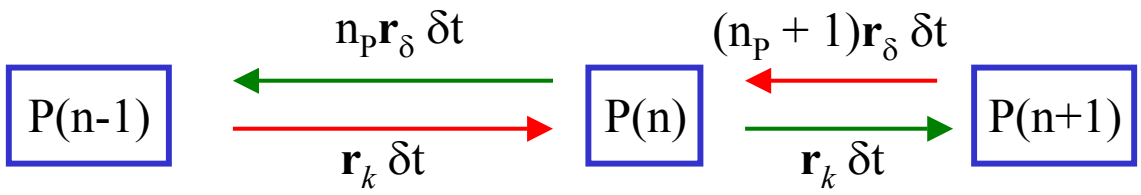
$$\emptyset \xrightarrow{k} P, \qquad \text{P}(k \text{ reaction}) = \mathbf{r}_k \, \delta t$$

$$P \xrightarrow{\delta} \emptyset, \qquad \text{P}(\delta \text{ reaction}) = n_P \, \mathbf{r}_\delta \, \delta t$$

$$\frac{d[P]}{dt} = k - \delta[P]$$

Thus, we go from reaction rates to reaction probabilities per unit time

How does the probability of having, say, n *P* molecules, p(n), change with time?



$$
\begin{aligned}
p(n, t+\delta t) &= p(n,t) \\
\rightarrow &+ p(n-1,t) r_k \delta t \\
\leftarrow &+ p(n+1,t)(n_P+1) r_\delta \delta t \\
\leftarrow &- p(n,t) n_P r_\delta \delta t \\
\rightarrow &- p(n,t) r_k \delta t .
\end{aligned}
$$

and thus we get in the limit $\delta t \rightarrow 0$

$$
\frac{dp(n)}{dt} = -p(n)(r_k + n_P r_\delta) + p(n-1) r_k + p(n+1)(n_P+1) r_\delta
$$

Some comments:

- All moments of the distribution p(n) can be derived from it
- It is a linear equation in p(n).
- Solving the master equation can be done for simple systems, however only normally at steady state.
- In connection with experiments, p(n) would represent the fraction of cells having n copies of some given protein

Equation of the mean; emergence of deterministic law

$$\frac{d\langle n\rangle}{dt} = \sum_n n\frac{dp_n}{dt}$$

$$= \sum_n n[-p_n(r_k + nr_\delta) + p_{n-1}r_k + p_{n+1}(n+1)r_\delta]$$

$$= -r_k\langle n\rangle - r_\delta\sum_n n^2 p_n + r_k\sum_n p_{n-1}n + r_\delta\sum_n n(n+1)p_{n+1}$$

$$= r_k - r_\delta\langle n\rangle$$

Considering that $[P] = \frac{\langle n \rangle}{V}$

We can rewrite the deterministic equation as

$$\frac{d\langle n \rangle}{dt} = Vk - \delta V[P] = Vk - \delta\langle n \rangle.$$

And thus

$$\boxed{\begin{aligned} r_k &= Vk \\ r_\delta &= \delta \end{aligned}}$$

pseudofirst-order reaction

first order reaction

Steady State

$$\frac{dp_n}{dt} = 0 = -p_n(r_k + nr_\delta) + p_{n-1}r_k + p_{n+1}(n+1)r_\delta$$

and

$$-p_n r_k + p_{n+1} r_\delta (n+1) = -p_{n-1} r_k + p_n r_\delta n$$

then

$-p_n r_k + p_{n+1} r_\delta (n+1)$ is constant (independent of n).

further, considering that $\langle n \rangle_{ss} = \frac{r_k}{r_\delta}$ this constant is zero

thus $p_n = \frac{\langle n \rangle_{ss}}{n} p_{n-1} = \ldots = \frac{\langle n \rangle_{ss}^n}{n!} p_0.$

since $\sum_n p_n = 1$ we get $\boxed{p_n = \frac{\langle n \rangle_{ss}^n}{n!} e^{\langle n \rangle_{ss}}}$

the steady state distribution is the Poisson Distribution

Poisson distribution

mean $\langle n \rangle = \langle n \rangle_{ss}$

variance $\sigma^2 = \langle n \rangle_{ss}$

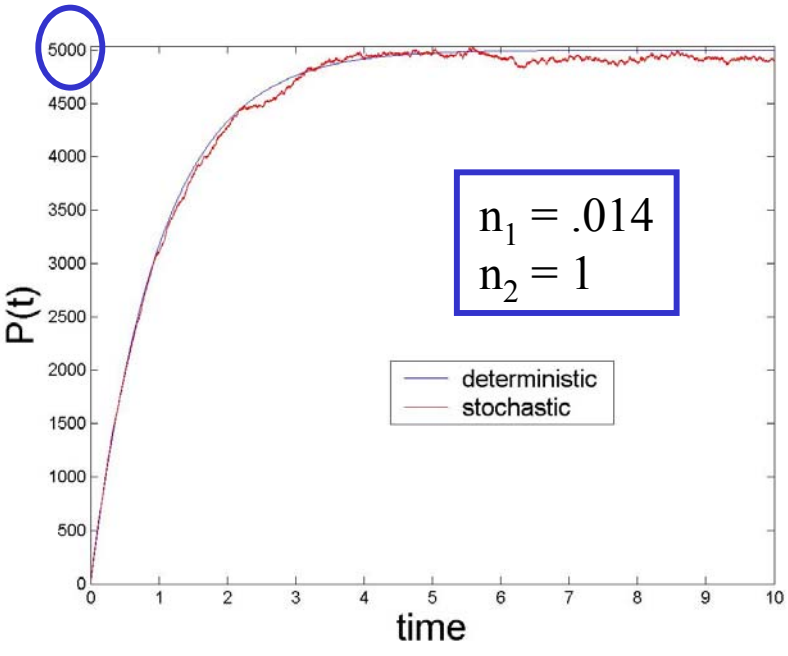
Macroscopic statistics

What is noise then?

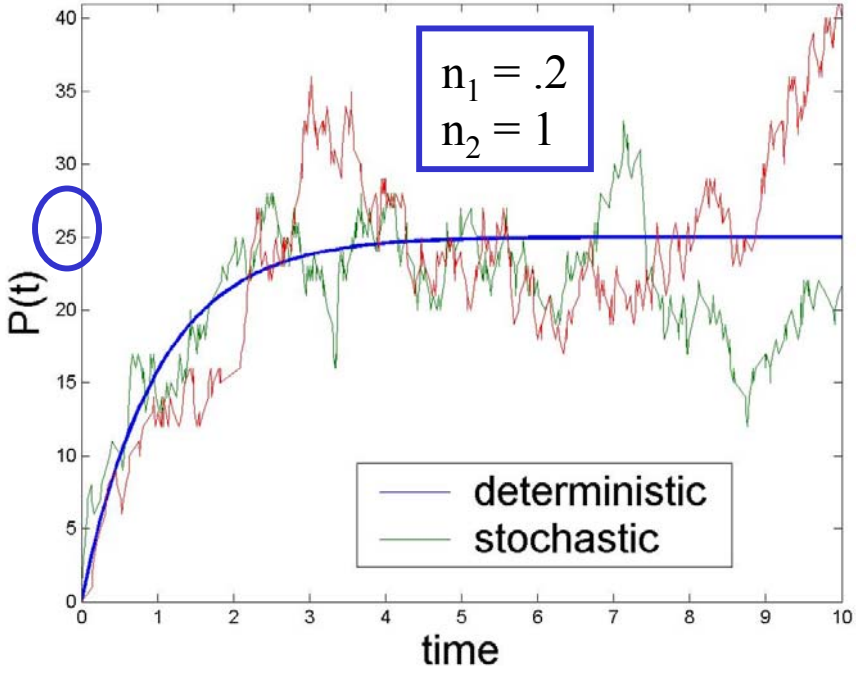
standard deviation

definition-1 (coe. variat.) = $n_1 = \frac{\sigma}{\langle n \rangle}$ (= $1/\sqrt{\langle n \rangle}$. Poisson distribution, noise increases as the number of molecules decreases)

definition-2 (Fano factor) = $n_2 = \frac{\sigma^2}{\langle n \rangle}$ (= 1, Poisson distribution mean = variance)



large number of molecules
deterministic approximation works



small number of molecules
deterministic approximation fails

large protein fluctuations

Master equations and gene expression

- Genes are generally regulated by complex nonlinear functions. Analitical studies become difficult.

- Two types of approximation methods

    1) Numerical Simulation → Gillespie's algorithm

    2) Perturbation Methods → Langevin equations, ...

Simulating Stochastic Reactions

Two key questions:

When will the next reaction occur?
What kind of reaction will it be?

$P(\tau,\mu)d\tau$ = probability that, given the state $(X_1,...X_N)$ at time t, the next reaction in V occurs in the infinitesimal time interval $(t + \tau, t + \tau + d\tau)$ and it will be a R_μ reaction.

propensity function, e.g., $n_1 n_2$ **r**

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

here,

$P_0(\tau)$: the probability that no reaction happens in the time interval $(t, t + \tau)$

$a_\mu d\tau$: the probability that reaction R_μ will happen in the time interval $(t + \tau, t + \tau + d\tau)$

The function $P_0(t)$: (no reaction)

$P_0(t+dt) = P_0(t)(1 - a_0\, dt)$ $\quad a_0 = \sum_{j=1,M} a_j$

$(P_0(t+dt) - P_0(t))/dt = -a_0\, P_0(t)$

$d\, P_0 /dt = -a_0\, P_0(t)$

$P_0(t) = \exp(-a_0 t)$

The reaction probability density function:

$P(\tau,\mu)d\tau = P_0(\tau)\, a_\mu d\tau = a_\mu \exp(-a_0\tau)\, d\tau$ $\quad \mu = 1,...,M \quad \tau \in (0,+\infty)$

It is possible to write $P(\tau,\mu)$ as a product of $P(\tau)$ and $P(\mu)$:

$P(\tau,\mu)d\tau = a_\mu \exp(-a_0\tau)\, d\tau = (a_\mu / a_0)\, a_0 \exp(-a_0\tau)\, d\tau$

$P(\mu) = (a_\mu / a_0)$

$P(\tau) = a_0 \exp(-a_0\tau)\, d\tau$

Therefore, we may determine the waiting time for the next reaction by generating two random numbers following distributions P(τ) and P(μ).

Note that the algorithm is a rigurous consequence of the Fundamental Hypothesis

Gillespie's algorithm

Step 0

Input the desired values for the stochastic rate constants $c_1,...,c_M$. Set the initial molecular population numbers $X_1,...,X_N$ and set the time variable t to 0. Initialize the unit-interval random number generator (note UiRN ↔ distributions $P(\tau)$ and $P(\mu)$).

Step 1

For the current state $X_1,...,X_N$ calculate and store M values of propensity functions $a_1 = h_1c_1,...,a_M = h_Mc_M$. Accumulate and store the sum of propensity functions $a_0=\sum_{j=1,M}a_j$

Step 2.

Generate two random numbers $r_1, r_2 \in (0,1)$ using UiRN. Calculate $\tau = (1/a_0)\ln(1/r_1)$ and take μ to be that integer for which $(a_1 + a_2 + ,..., + a_{\mu-1}) < r_2a_0 \leq (a_\mu +, ..., + a_M)$

Step 3.

Update the state of the system by executing one elementary reaction R_μ and increase time of the simulation t by τ.

$t < Tmax$

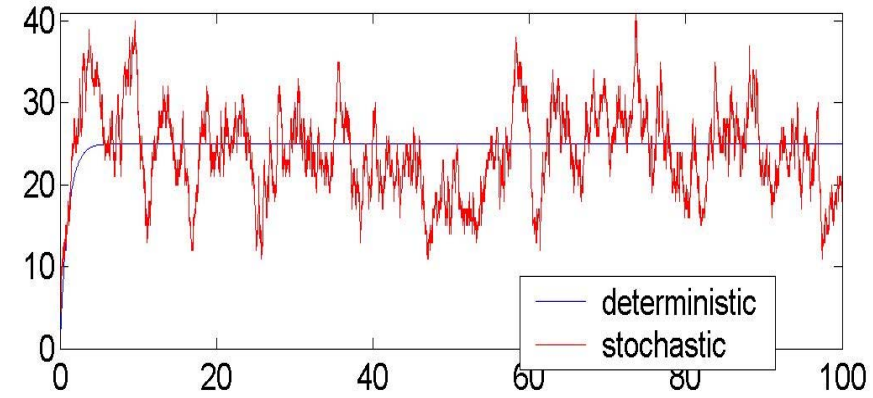
Finish

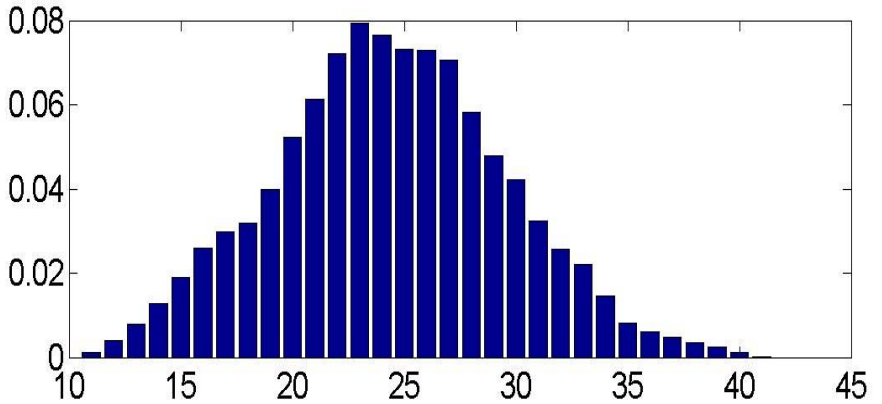
MATLAB code 2

```
% .. code1stoch.m
% .. simple gene expression stochastic and deterministic

clear all
k = 25;
delta = 1;

% .. stochastic eqs. Gillespie's algorithm
P = 0;
Pstochastic = P;
tmax = 10;
t = 0;
tspan = t;
```

```
while t < tmax

    % .. a's
    a = [k, delta*P(1)];
    a0 = sum(a);
    % .. determine time of next reaction
    r1 = rand;
    tau = -log(r1)/a0;
    t = t + tau;
    % .. determine nature of next reaction
    r2 = rand;
    acumsum = cumsum(a)/a0;
    chosen_reaction = min(find(r2 <= acumsum));

    if chosen_reaction == 1;
        P(1) = P(1) + 1;
    else
        P(1) = P(1) - 1;
    end

    tspan = [tspan,t];
    Pstochastic = [Pstochastic;P];

end
```

```
% .. deterministic eqs.
P0 = 0;
options = [];
[t P] = ode23(@code1equations,tspan,P0,options,k,delta);

% .. plot
subplot(211)
plot(t,P,t,Pstochastic,'r')
legend('deterministic','stochastic')
axis([0 tmax 0 max(Pstochastic)]);

% .. histogram, example of matlab use
subplot(212)
vv = Pstochastic(find(t>3));
his = min(vv):max(vv);
histovv = length(his);
cc = 0;
for n = his
    cc = cc + 1;
    histovv(cc) = length(find(vv == n));
end
histovv = histovv/sum(histovv);
bar(his,histovv)
meanhist = sum(his.*histovv)
varihist = sum(his.*his.*histovv) - meanhist*meanhis_
fano = varihist/meanhist
```






```
meanhist = 24.3909
varihist = 26.7338
fano = 1.0961
```