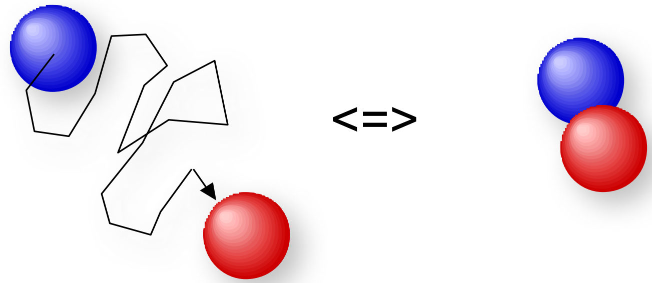
V22

Dynamic Modelling: Rate Equations + Stochastic Propagation

Mass Action Kinetics

Most simple dynamic system: inorganic chemistry

Consider reaction A + B <=> AB



Interesting quantities:
(changes of) densities of A, B, and AB

density = $\frac{\text{number of particles}}{\text{unit volume}}$

$$[A] = \frac{N_A}{V}, \quad \frac{d}{dt}[A](t)$$

1 mol = 1 Mol / Liter = 6.022 x 10^{23} x $(0.1\ \text{m})^{-3}$ = 0.6 nm^{-3}

This means that proteins cannot reach 1 mol concentrations. Why?

Association: probability that A finds and reacts with B
=> changes proportional to densities of A *and* of B

Dissociation: probability for AB to break up
=> changes proportional to density of AB

How to put this into formulas?

Mass Action II

Again: $A + B \Leftrightarrow AB$

Objective: mathematical description for the changes of [A], [B], and [AB]

Consider [A]:

Gain due to dissociation AB => A + B

Loss due to association A + B => AB

$$\frac{d}{dt}[A] = G_A - L_A$$

AB falls apart
=> G_A depends only on [AB]

A has to find B
=> L_A depends on [A] *and* [B]

$$G_A = k_r\,[AB]$$

$$L_A = k_f\,[A]\,[B]$$

phenomenological proportionality constant

$$\frac{d}{dt}[A] = k_r\,[AB] - k_f\,[A]\,[B]$$

Mass Action !!!

A + B <=> AB

For [A]: we just found: $$\frac{d}{dt}[A] = k_r[AB] - k_f[A][B]$$

For [B]: for symmetry reasons $$\frac{d}{dt}[B] = \frac{d}{dt}[A]$$

For [AB]: exchange gain and loss $$\frac{d}{dt}[AB] = -\frac{d}{dt}[A] = k_f[A][B] - k_r[AB]$$

with $[A](t_0)$, $[B](t_0)$, and $[AB](t_0)$ => complete description of the system

time course = initial conditions + dynamics

A Second Example

Slightly more complex: $A + 2B \Leftrightarrow AB_2$

Association: • one A and two B have to come together
• forming one complex AB_2 requires two units of B

$$L_A = k_f [A][B][B] = k_f [A][B]^2 \qquad L_B = 2k_f [A][B]^2$$

Dissociation: one AB_2 decays into one A and two B

$$G_A = k_r [AB_2] \qquad G_B = 2k_r [AB_2]$$

Put everything together

$$\frac{d}{dt}[A] = k_r [AB_2] - k_f [A][B]^2 \qquad \frac{d}{dt}[B] = 2\frac{d}{dt}[A] \qquad \frac{d}{dt}[AB_2] = -\frac{d}{dt}[A]$$

Some Rules of Thumb

$A + 2B \Leftrightarrow AB_2$ "A is produced when AB_2 falls apart or is consumed when AB_2 is built from one A and two B"

Sign matters: Gains with "+", losses with "–"

Logical conditions: "…from A *and* B"

"and" corresponds to "×" "or" corresponds to "+"

Stoichiometries: one factor for each educt (=> $[B]^2$)
prefactors survive

Mass conservation: terms with "–" have to show up with "+", too

$$\frac{d}{dt}[A] = k_r\,[AB_2] - k_f\,[A]\,[B]^2 \qquad \frac{d}{dt}[B] = 2\frac{d}{dt}[A] \qquad \frac{d}{dt}[AB_2] = -\frac{d}{dt}[A]$$

A Worked Example

Lotka-Volterra population model

R1:	A + X => 2X	prey X lives on A
R2:	X + Y => 2Y	predator Y lives on prey X
R3:	Y => B	predator Y dies

stoichiometric matrix S

Rates for the reactions

$$\frac{dR_1}{dt} = k_1 \, A \, X$$

$$\frac{dR_2}{dt} = k_2 \, X \, Y$$

$$\frac{dR_3}{dt} = k_3 \, Y$$

Changes of the metabolites

	R1	R2	R3
A	–1		
X	1	–1	
Y		1	–1
B			1

=> change of X:

$$\frac{dX}{dt} = +k_1 \, A \, X \; - \; k_2 \, X \, Y \; + \; 0$$

Setting up the Equations

With $\vec{v} = \frac{d\vec{R}}{dt} = \begin{pmatrix} dR_1/dt \\ dR_2/dt \\ dR_3/dt \end{pmatrix}$ and $S = \begin{pmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$

we get: $\frac{d}{dt}\vec{X} = \frac{d}{dt}\begin{pmatrix} A \\ X \\ Y \\ B \end{pmatrix} = S\,\frac{d}{dt}\vec{R}$ or $\frac{dX_i}{dt} = \sum_j S_{ij}\frac{dR_j}{dt}$

amounts processed per reaction

speeds of the reactions

Plug in to get:

$$\frac{dA}{dt} = -\frac{dR_1}{dt} = -k_1\,A\,X \qquad \frac{dX}{dt} = +\frac{dR_1}{dt} - \frac{dR_2}{dt} = k_1 AX - k_2 XY$$

$$\frac{dB}{dt} = +\frac{dR_3}{dt} = k_3\,Y \qquad \frac{dY}{dt} = +\frac{dR_2}{dt} - \frac{dR_3}{dt} = k_2 XY - k_3 Y$$

How Does It Look Like?

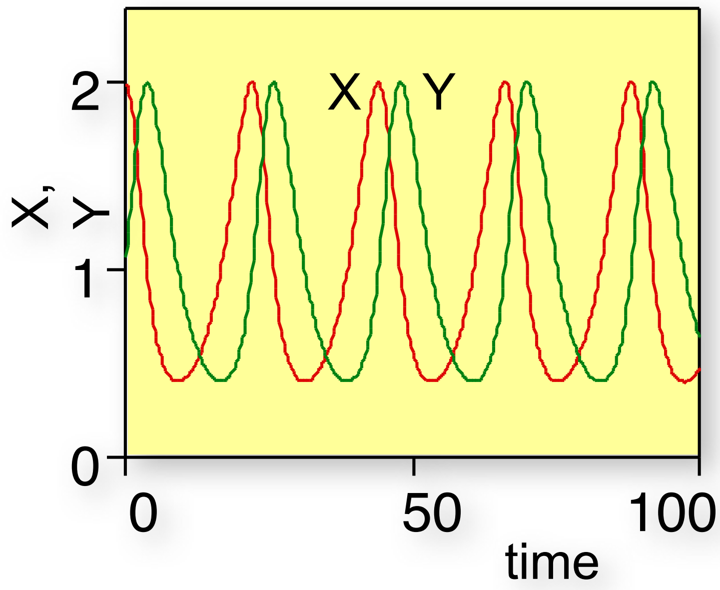
Lotka–Volterra: assume A = const, B ignored

=> cyclic population changes

$$\frac{dX}{dt} = k_1 AX - k_2 XY$$

$$\frac{dY}{dt} = k_2 XY - k_3 Y$$

$k_1 = k_2 = k_3 = 0.3$



Steady State: when the populations do not change anymore

$$\frac{dX}{dt} = \frac{dY}{dt} = 0 \quad => \quad Y = \frac{k_1}{k_2} A \qquad X = \frac{k_3}{k_2}$$

Steady state = fluxes balanced

With $k_1 = k_2 = k_3 = 0.3$ and $A = 1$ => $X = Y = 1$

From rates to differences

Reaction: $$A + B \Longrightarrow AB$$

Rate equation: $$\frac{dA}{dt} = -k \cdot A \cdot B = f\left(A(t), B(t)\right)$$

derivative of A(t) = some function

Taylor expansion for displacement t around $t_0 = 0$:

$$A(t) = A(0) + t \cdot \frac{dA}{dt}(0) + \frac{t^2}{2} \cdot \frac{d^2A}{dt^2}(0) + \ldots = \sum_{k=0} \frac{t^k}{k!} \cdot \frac{d^kA}{dt^k}(0)$$

Truncate this expansion after second term (linear approximation):

$$\begin{aligned} A(t) &\approx A(0) + \quad t \cdot \frac{dA}{dt}(0) \quad + \quad O(t^2) \\ &\approx A(0) + t \cdot f\left(A(0), B(0)\right) + O(t^2) \end{aligned}$$

From rates to differences II

Linear approximation to (true) A(t):

$$
\begin{aligned}
A(t) &\approx A(0) + \quad t \cdot \frac{dA}{dt}(0) \quad + \quad O(t^2) \\
&\approx A(0) + t \cdot f\left(A(0), B(0)\right) + O(t^2)
\end{aligned}
$$

initial condition   increment   error

For $t \to 0$

$$t \cdot \frac{dA}{dt}(0) \;\gg\; \frac{t^2}{2} \cdot \frac{d^2A}{dt^2}(0) \;\gg\; \ldots$$

Use linear approximation for small time step Δt:

$$\boxed{A(t+\Delta t) = A(t) + \Delta t \cdot \frac{dA}{dt}(t)}$$

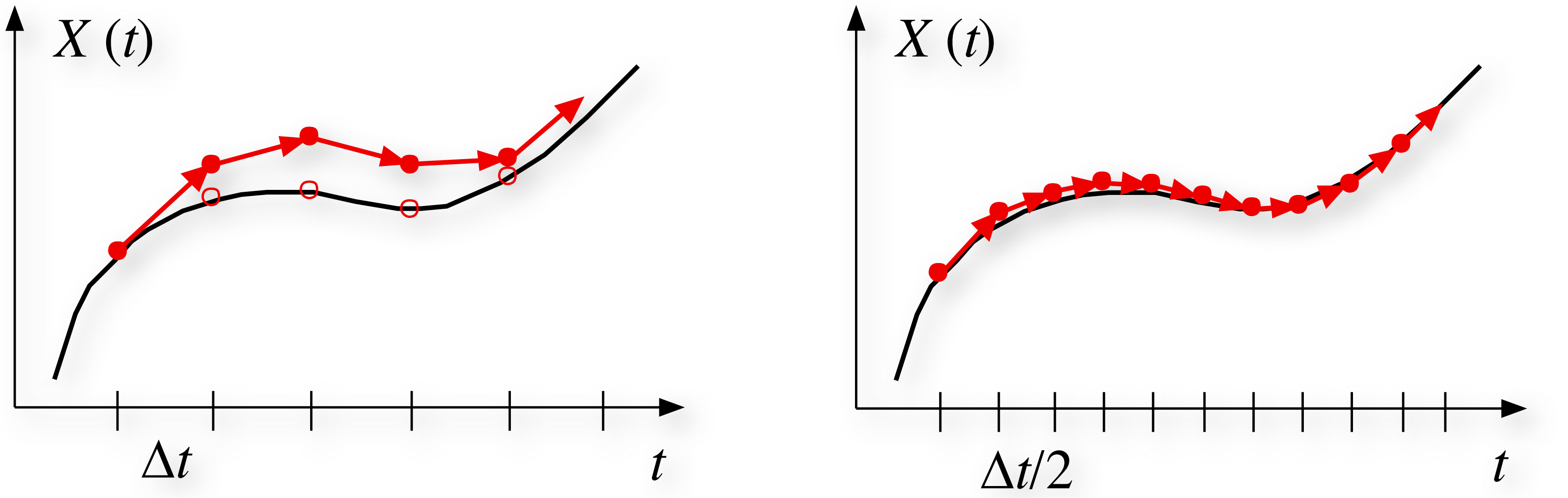
This is the so-called **"forward Euler" algorithm**

"Forward Euler" algorithm

General form: $$\vec{X}_i(t+\Delta t) = \vec{X}_i(t) + \Delta t \cdot \vec{f}(\vec{X}_j(t)) + O(\Delta t^2)$$

relative error: $$\varepsilon = \frac{\Delta t^2/2 \cdot X''}{\Delta t\, X'} \propto \Delta t$$ 1st order algorithm

relative error decreases with 1st power of step size Δt

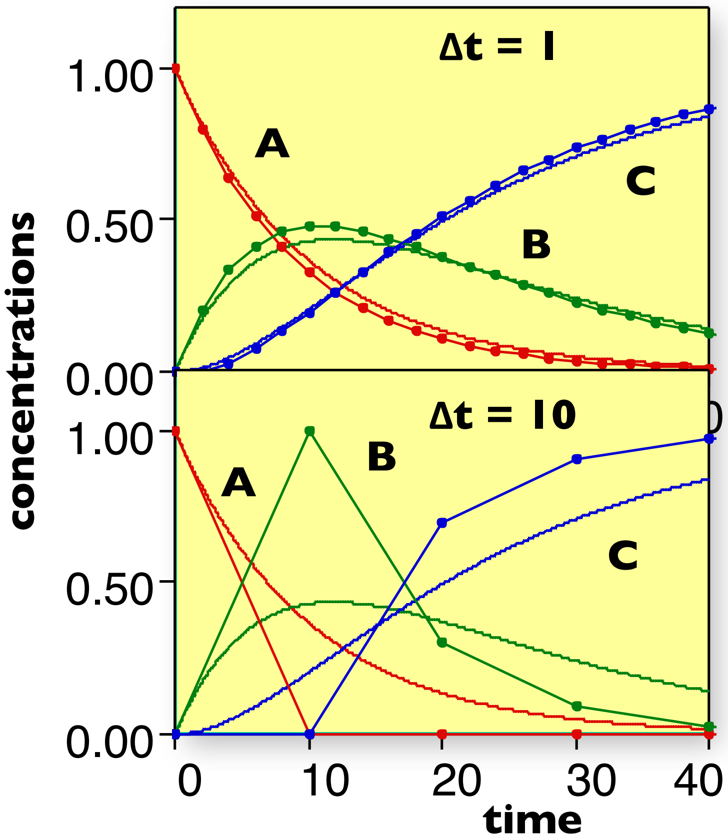


Black: ideal dynamic trajectory, **red:** dynamics integrated by forward Euler algorithm
Right side: integration time steps are half of left side -> smaller error

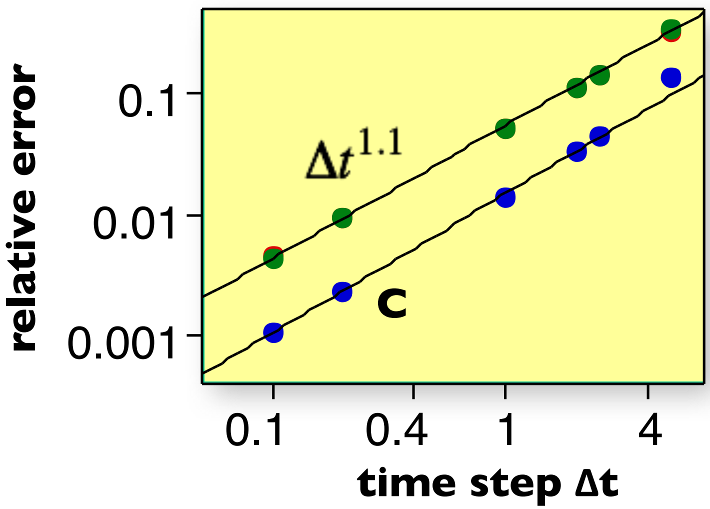
Example: chained reactions

Reaction: $A \Longrightarrow B \Longrightarrow C$ $k_{AB} = 0.1, \quad k_{BC} = 0.07$

Time evolution:



Relative error vs. Δt
at t = 10:



runtime α $(\Delta t)^{-1}$

Example Code: Forward Euler

BspCode_Euler.py

```python
# Initial values
A = 1.0
B = 0.0
C = 0.0

# Rate constants
k1 = 0.1
k2 = 0.07

dt = 0.1
t = 0

# main loop
while(t < 20.0):
    # derivatives
    dR1 = k1 * A
    dR2 = k2 * B

    # add up changes
    A += dt * (-dR1)
    B += dt * (dR1 - dR2)
    C += dt * dR2

    # increment t
    t += dt

    # output
    print t, A, B, C
```

A => B => C

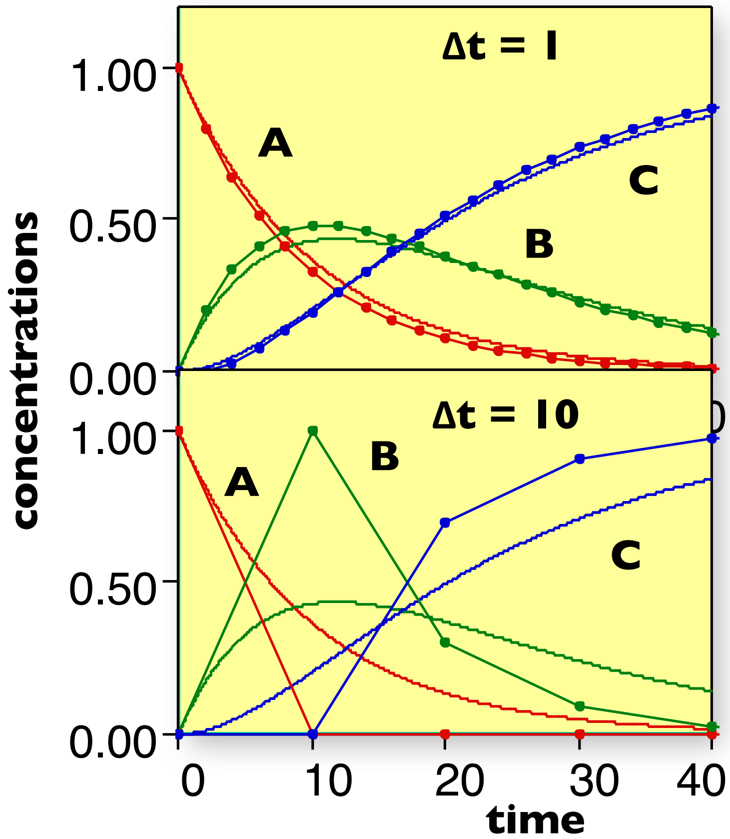
Iterate:

$$A(t+\Delta t) \;=\; A(t) \;+\; \Delta t \cdot \frac{dA}{dt}(t)$$

Important:

first calculate all derivatives,
then update densities!

What is the “correct” time step?



$$A \Longrightarrow B \Longrightarrow C$$

Approximation works for:

$$|\Delta A| = \left|\Delta t \frac{dA}{dt}\right| = |-k_{AB} \cdot A \cdot \Delta t| \ll A$$

=> $\Delta t \ll \dfrac{1}{\max(k)}$

Here: $k_{AB} = 0.1, \quad k_{BC} = 0.07$

=> $\Delta t \ll 0.1^{-1} = 10$

Note 1:
read “«” as “a few percent”

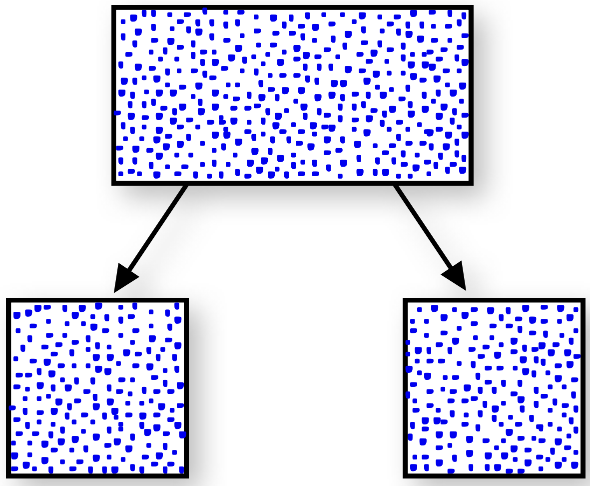
From Test Tubes to Cells

Rate equations <=> description via densities

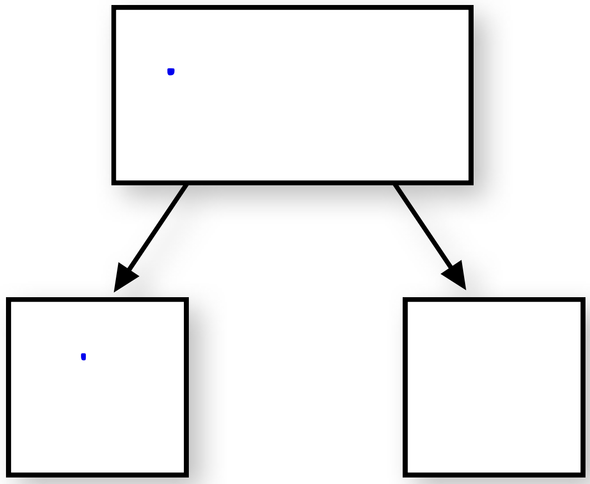
$$\text{density} = \frac{\text{indistinguishable particles}}{\text{volume element}}$$

=> density is a continuum measure,
independent of the volume element

"half of the volume => half of the particles"



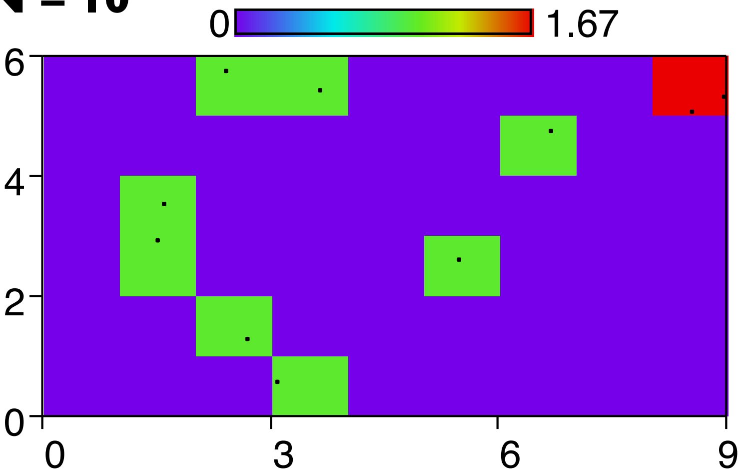
When density gets very low
=> each particle matters



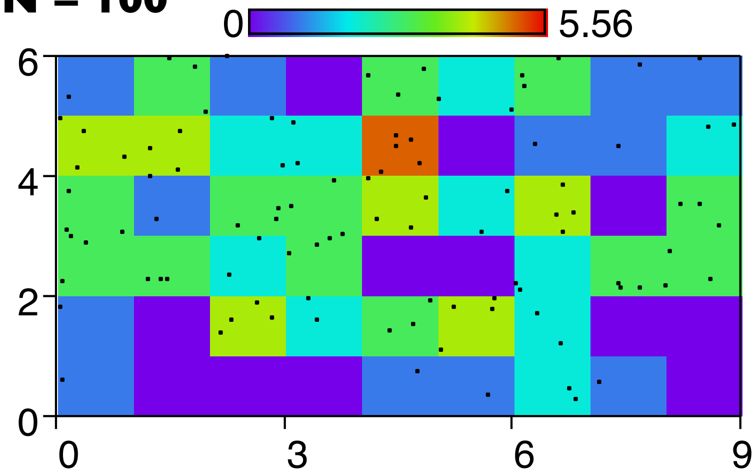
Examples:
~10 Lac repressors per cell, chemotaxis,
transcription from a single gene, …

Density Fluctuations

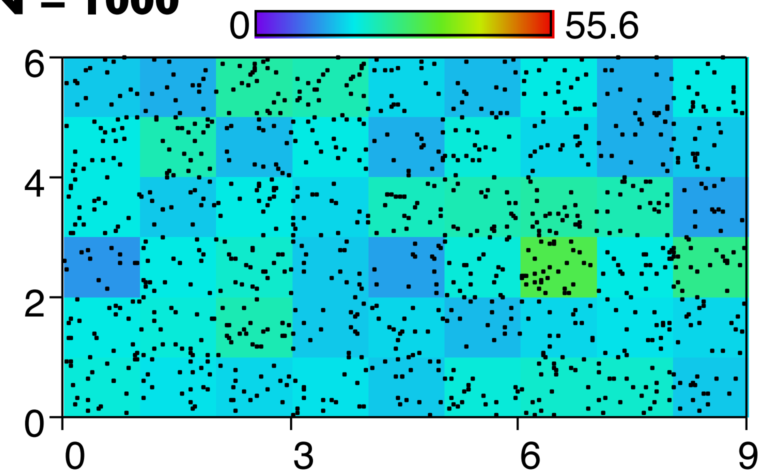
N = 10



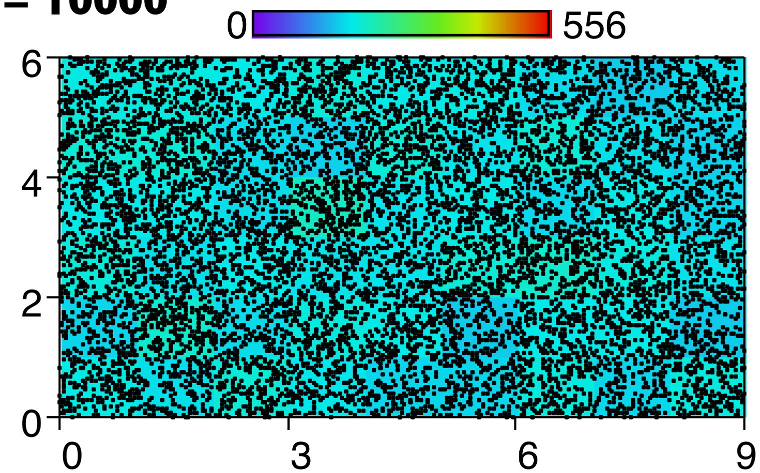
N = 100



N = 1000



N = 10000



Spread: Poisson Distribution

Stochastic probability that *k* events occur follows the Poisson distribution (here: event = "a particle is present"):

$$p_k = \frac{\lambda^k}{k!} e^{-\lambda}$$

k = 0, 1, 2, …
λ > 0 is a parameter

Average: $\langle k \rangle = \sum k \, p_k = \lambda$

Variance: $\sigma^2 = \sum p_k \, (k - \langle k \rangle)^2 = \lambda$

$$\sigma = \sqrt{\lambda}$$

Relative spread (error): $\frac{\Delta k}{k} = \frac{\sigma}{\langle k \rangle} = \frac{1}{\sqrt{\lambda}}$

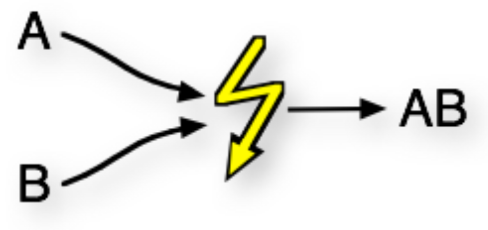
Avg. number of particles per unit volume	100	1000	1 Mol
relative uncertainty	10%	3%	1e-12

=> Fluctuations are negligible for "chemical" test tube situations

Reactions in the Particle View

Consider association:

A + B => AB



Continuous rate equation: $$\frac{d[AB]}{dt} = k\,[A]\,[B]$$

Number of new AB in volume V during Δt:

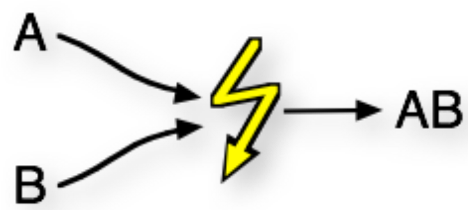
$$\begin{aligned} \Delta N_{AB} &= \frac{d[AB]}{dt}\,V\,\Delta t \\ &= k_{AB}\,\frac{N_A}{V}\,\frac{N_B}{V}\,V\,\Delta t \\ &= \frac{k_{AB}\,\Delta t}{V}\,N_A\,N_B \\ &= P_{AB}\,N_A\,N_B \end{aligned}$$

Density "picture" | Particle "picture"
reaction rate k_{AB} => **reaction probability** P_{AB}

Units!

Consider:

A + B => AB



Change in the number of AB:

$$\Delta N_{AB} = P_{AB}\, N_A\, N_B$$

Association probability:

$$P_{AB} = \frac{k_{AB}\,\Delta t}{V}$$

Units: Continuous case

$$\frac{dAB}{dt} = k_{AB}\, A\, B$$

$$\left[\frac{dAB}{dt}\right] = \frac{\text{Mol}}{l\, s} \qquad [A] = [B] = \frac{\text{Mol}}{l} \qquad \Leftrightarrow \qquad [k_{AB}] = \frac{l}{\text{Mol}\, s}$$

Stochastic case

$$[N_{AB}] = [N_A] = [N_B] = 1 \qquad \Leftrightarrow \qquad [P_{AB}] = 1$$

Direct Implementation

A + B => AB

```python
# continuous association of A and B

# parameter
tEnd = 5.0
dt = 0.01
volume = 100.0

# rate and probability
kAB = 1.0
prob = kAB * dt / volume

# initial conditions: particle numbers
A = 1000
B = 1000
AB = 0

# convert to densities
A = A/volume
B = B/volume
AB = AB/volume

# main loop
t = 0.0
print t, "\t", A, "\t", B, "\t", AB

while(t<tEnd):
    dAB = dt * kAB * A * B

    AB += dAB
    A -= dAB
    B -= dAB

    # increment time and output
    t += dt
    print t, "\t", A, "\t", B, "\t", AB
```

```python
# Stochastic association of A + B => AB

import random

# parameter
tEnd = 5.0
dt = 0.01
volume = 100.0

# rate and probability
kAB = 1.0
prob = kAB * dt / volume

# initial conditions
A = 1000
B = 1000
AB = 0

# main loop
t = 0.0
print t, "\t", A/volume, "\t", B/volume, "\t", AB/volume

while(t<tEnd):
    dAB = 0
    # check for every pair A, B
    for ia in xrange(A):
        for ib in xrange(B):
            r = random.random()
            if (r < prob):
                dAB+=1
    AB += dAB
    A -= dAB
    B -= dAB

    # increment time and output
    t += dt
    print t, "\t", A/volume, "\t", B/volume, "\t", AB/volume
```

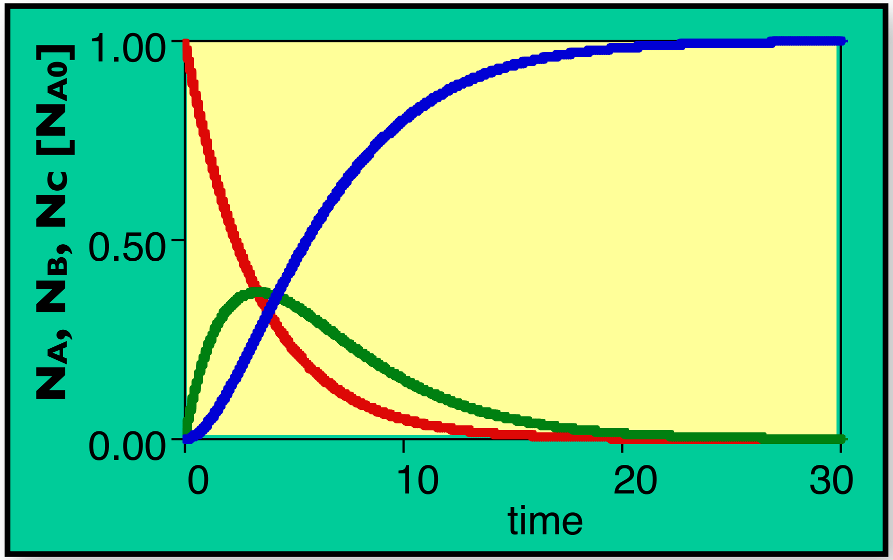
Note: both versions are didactic implementations

Example: Chained Reactions

A => B => C

Rates: $$\frac{dA}{dt} = -k_1 A \qquad \frac{dB}{dt} = k_1 A - k_2 B \qquad \frac{dC}{dt} = k_2 B$$

Time course from continuous rate equations (benchmark):

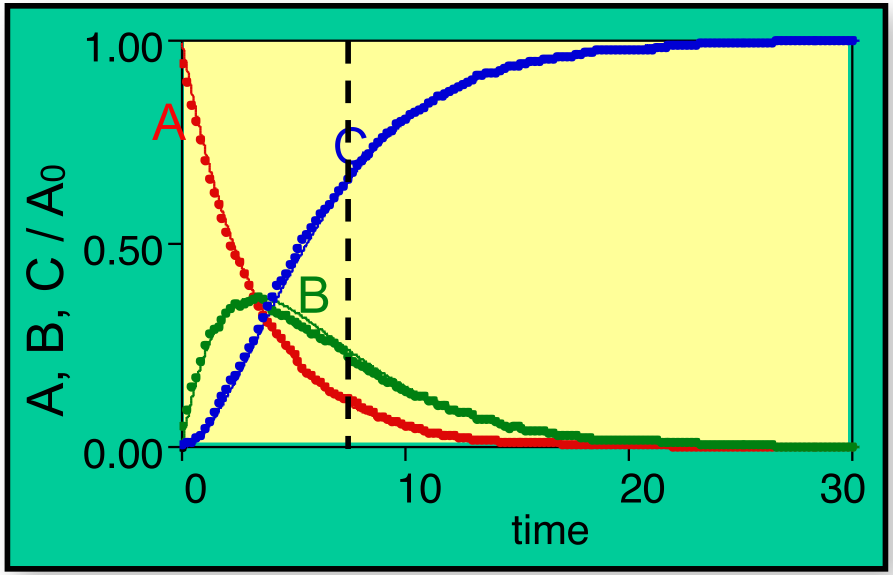


$k_1 = k_2 = 0.3$ (units?)

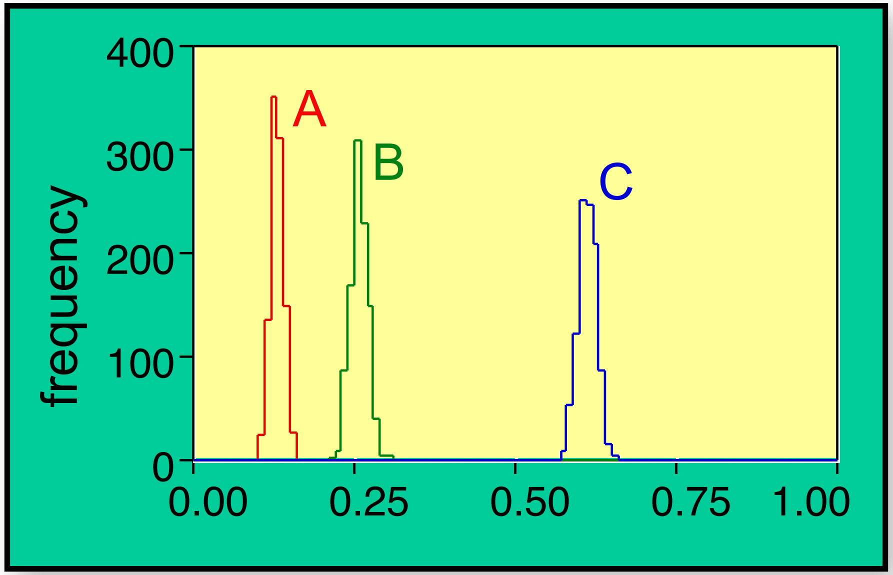
Stochastic Implementation

A => B => C $\qquad$ A_0 = 1000 particles initially

t = 7



$k_1 = k_2 = 0.3$

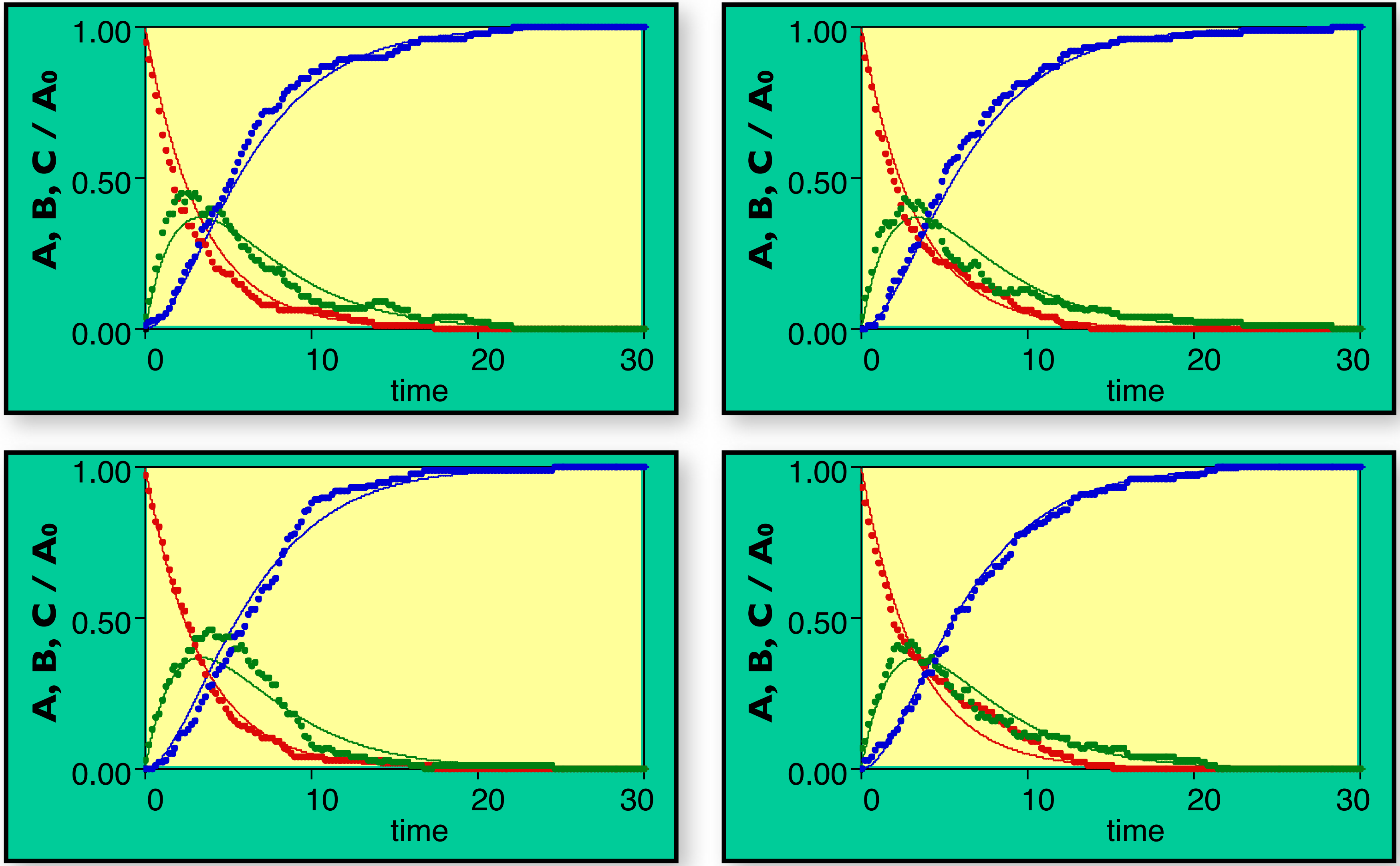


Values at t = 7 (1000 runs)

=> Stochastic version exhibits fluctuations

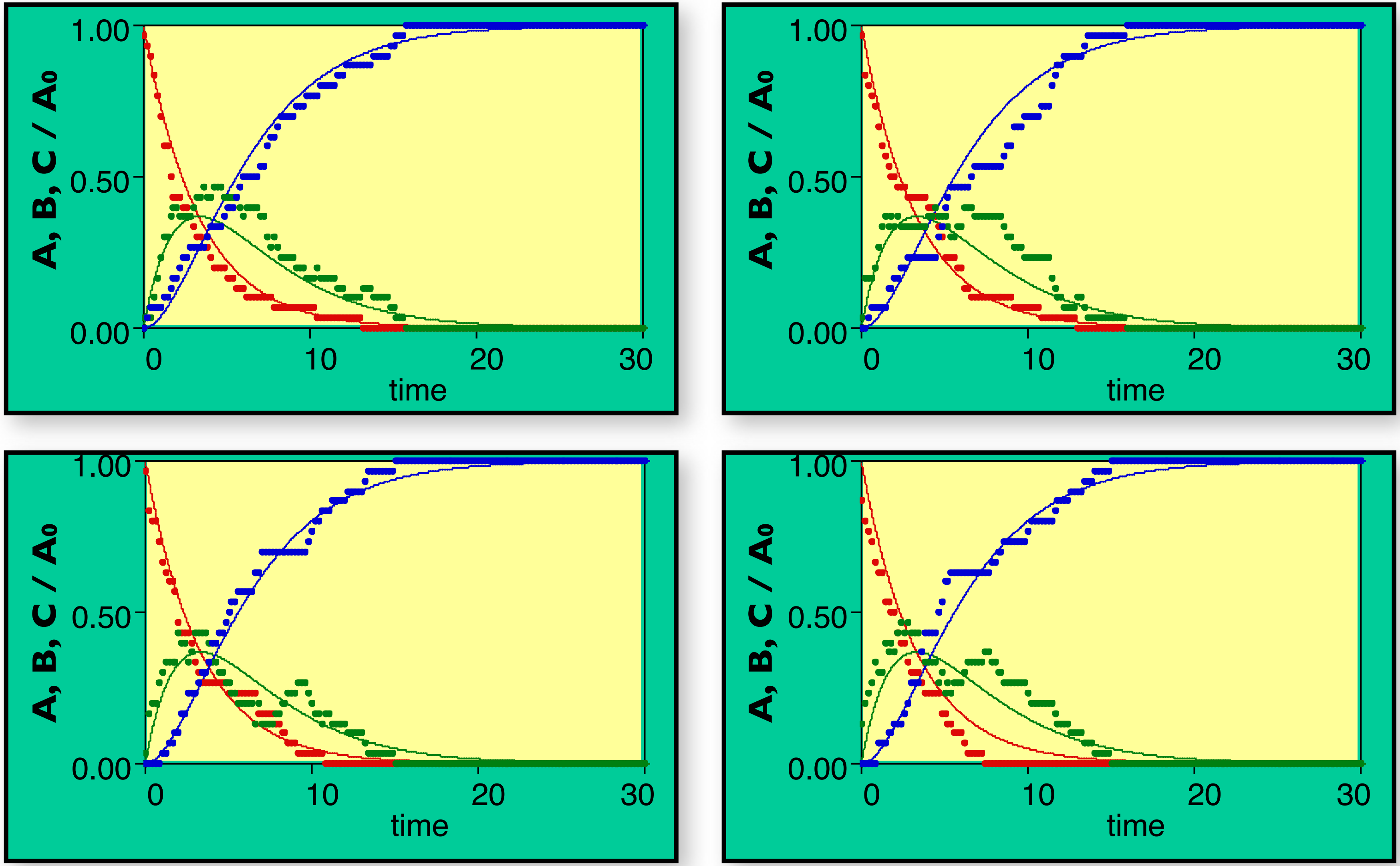
Less Particles => Larger Fluctuations

$A_0 = 100$ shown are 4 different runs



Even Less Particles

$A_0 = 30$



Spread vs. Particle Number

Poisson:
relative fluctuations $\propto 1/\sqrt{N}$

Repeat calculation 1000 times and record values at t = 7.

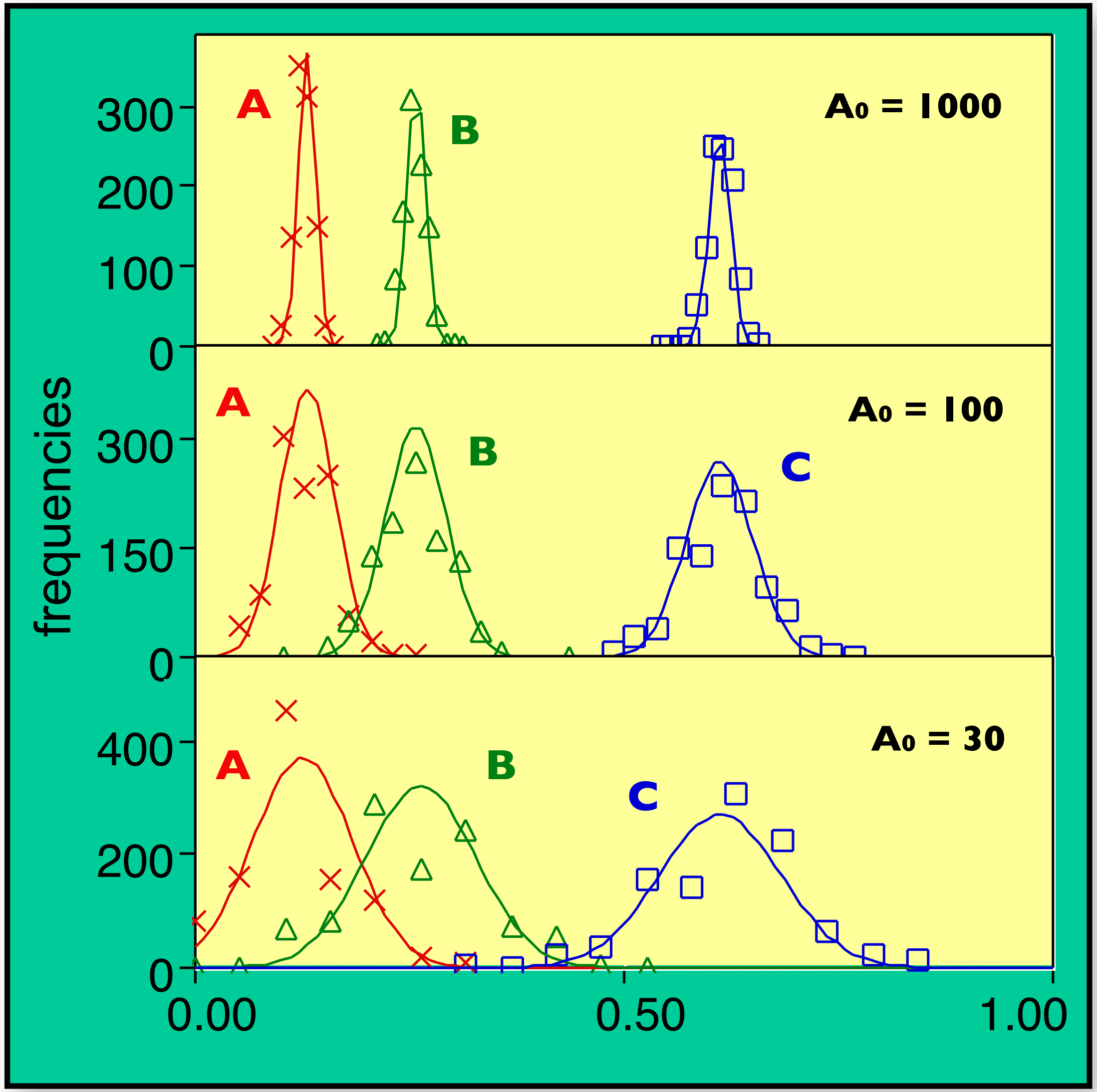
Fit distributions with Gaussian (Normal distribution)

$$g(x) = \exp\left[-\frac{(x - <x>)^2}{w/\sqrt{A_0}}\right]$$

$<A> = 0.13$, $w_A = 0.45$

$<B> = 0.26$, $w_B = 0.55$

$<C> = 0.61$, $w_C = 0.45$



Stochastic Propagation

Naive implementation:

```
For every timestep:
      events = 0
      For every possible pair of A,
      B:
          get random number r ∈ [0, 1)
          if r ≤ PAB:
              events++
      AB += events
      A, B -= events
```

Features of this implementation

+ very simple
+ direct implementation of the underlying process

– costly runtime $O(N^2)$
– first order approximation
– one trajectory at a time

=> how to do better???

Determine complete probability distribution
=> **Master equation**

More efficient propagation
=> **Gillespie algorithm**

A Fast Algorithm

Exact Stochastic Simulation of Coupled Chemical Reactions

Daniel T. Gillespie*

Research Department, Naval Weapons Center, China Lake, California 93555 *(Received May 12, 1977)*

Publication costs assisted by the Naval Weapons Center

There are two formalisms for mathematically describing the time behavior of a spatially homogeneous chemical system: The *deterministic approach* regards the time evolution as a continuous, wholly predictable process which is governed by a set of coupled, ordinary differential equations (the "reaction-rate equations"); the *stochastic approach* regards the time evolution as a kind of random-walk process which is governed by a single differential-difference equation (the "master equation"). Fairly simple kinetic theory arguments show that the stochastic formulation of chemical kinetics has a firmer physical basis than the deterministic formulation, but unfortunately the stochastic master equation is often mathematically intractable. There is, however, a way to make exact numerical calculations within the framework of the stochastic formulation without having to deal with the master equation directly. It is a relatively simple digital computer algorithm which uses a rigorously derived Monte Carlo procedure to *numerically simulate* the time evolution of the given chemical system. Like the master equation, this "stochastic simulation algorithm" correctly accounts for the inherent fluctuations and correlations that are necessarily ignored in the deterministic formulation. In addition, unlike most procedures for numerically solving the deterministic reaction-rate equations, this algorithm never approximates infinitesimal time increments dt by finite time steps Δt. The feasibility and utility of the simulation algorithm are demonstrated by applying it to several well-known model chemical systems, including the Lotka model, the Brusselator, and the Oregonator.

D. Gillespie, J. Phys. Chem. 81 (1977) 2340–2361

Gillespie – Step 0

Consider decay reaction: A => Ø (this model describes e.g. the radioactive decay)

Probability for one reaction in (t, $t+\Delta t$) with $A(t)$ molecules = $A(t)\ k\ \Delta t$

Naive Algorithm:

```
A = A0
For every timestep:
    get random number r ε
    [0, 1)
    if r ≤ A*k*dt:
       A = A-1
```

It works, but: $A*k*dt << 1$ for reasons of (good) accuracy

=> many many steps where nothings happens

=> Use adaptive stepsize method?

Gillespie – Step 1

Idea: Figure out **when** the next reaction will take place!

(In between the discrete events nothing happens anyway … :-)

Suppose there are A(t) molecules in the system at time t

f(A(t), s) = probability that with A(t) molecules the next reaction takes place in interval (t+s, t+s+ds) with ds => 0

g(A(t), s) = probability that with A(t) molecules no reaction occurs in (t, t+s)

Then:
$$f(A(t),s)\,ds = g(A(t),s)\,A(t+s)\,k\,ds$$

No reaction during (t, t+s):

$$f(A(t),s)\,ds = g(A(t),s)\,\underbrace{A(t)\,k\,ds}_{\textbf{probability for reaction in (t+s, t+s+ds)}}$$

Probability for (No Reaction)

Now we need g(A(t), s)

Extend g(A(t), s) a bit:

$$g(A(t), s+ds) = g(A(t), s)\,[1 - A(t+s)\,k\,ds]$$

Replace again *A(t+s)* by *A(t)* and rearrange:

$$\lim_{ds \to 0} \frac{g(A(t), s+ds) - g(A(t), s)}{ds} = \frac{dg(A(t), s)}{ds} = -A(t)k\, g((A(t), s)$$

With g(A, 0) = 1 ("no reaction during no time")

=> Distribution of waiting times between discrete reaction events:

$$g(A(t), s) = \exp[-A(t)\,k\,s]$$

Life time = average waiting time: $s_0 = \frac{1}{kA(t)}$

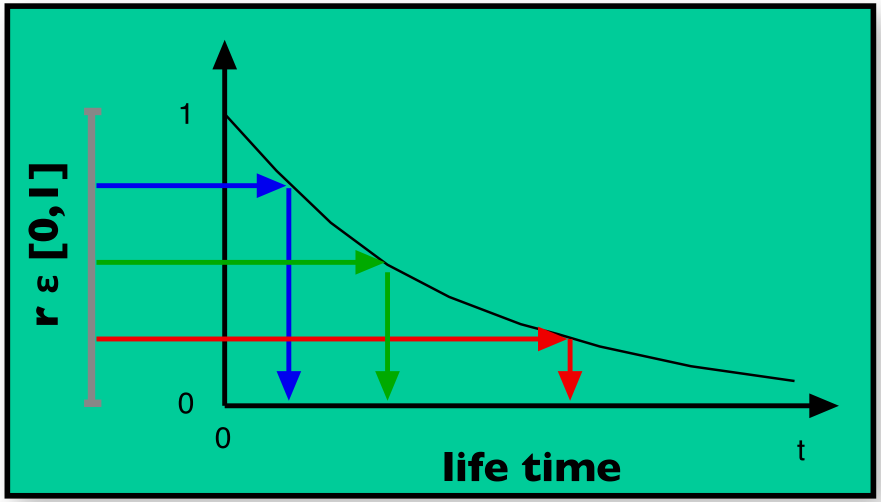
Exponentially Distributed Random Numbers

Exponential probability distribution:

$$g(A(t),s) = \exp[-A(t)\,k\,s]$$

Solve $r = \exp[-A(t)\,k\,s]$ for s:

$$s = \frac{1}{kA(t)} \ln\left[\frac{1}{r}\right] = \frac{1}{\alpha_0} \ln\left[\frac{1}{r}\right]$$



Simple Gillespie algorithm for the decay reaction A => Ø :

```
A = A0
While(A > 0):
     get random number r ε [0, 1)
     t = t + s(r)
     A = A - 1
```

Gillespie vs. Naive Algorithm

Naive:

"What is the probability that an event will occur during the next Δt?"

=> small fixed timesteps

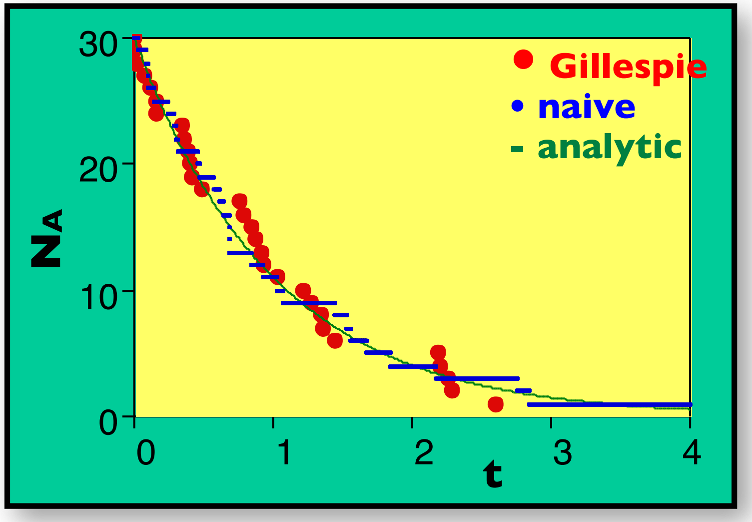
=> 1st order approximation

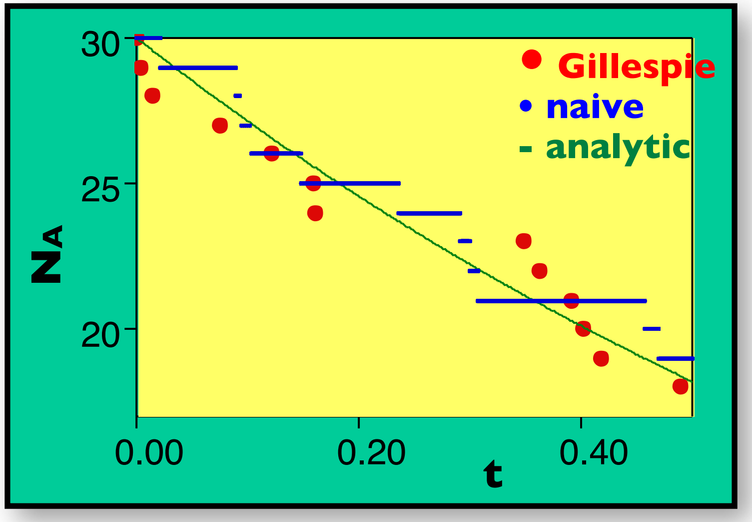
Gillespie:

"How long will it take until the next event?"

=> variable timesteps

=> exact





Gillespie – Complete

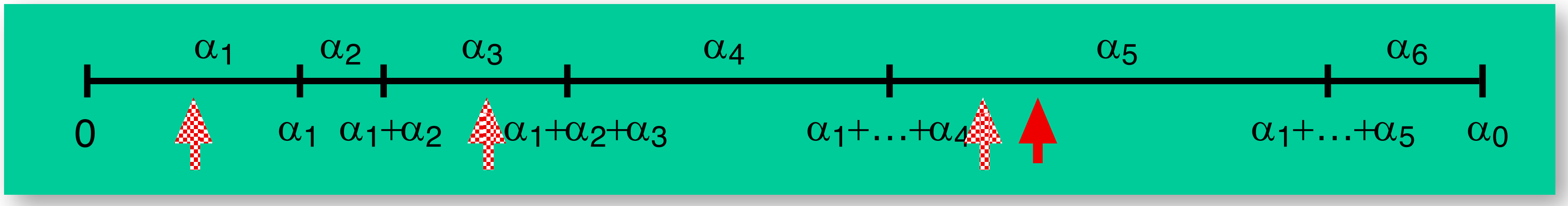
For an arbitrary number of reactions (events):

(i) determine probabilities for the individual reactions: α_i $i = 1, \ldots, N$

total probability $\alpha_0 = \Sigma\, \alpha_i$

(ii) get time s until next event in any of the reactions: $$s = \frac{1}{\alpha_0} \ln\left[\frac{1}{r_1}\right]$$

(iii) Choose the next reaction j from: $$\sum_{i=1}^{j-1} \alpha_i \leq \alpha_0 r_2 < \sum_{i=1}^{j} \alpha_i$$



(iv) update time and particle numbers

An Example with Two Species

Reactions: $A + A \xrightarrow{k_1} \varnothing$ $A + B \xrightarrow{k_2} \varnothing$ $\varnothing \xrightarrow{k_3} A$ $\varnothing \xrightarrow{k_4} B$

Continuous rate equations: $$\frac{dA}{dt} = k_3 - 2A^2 k_1 - ABk_2$$

Stationary state:

with $k_1 = 10^{-3}\ s^{-1}$ $k_2 = 10^{-2}\ s^{-1}$ $k_3 = 1.2\ s^{-1}$ $k_4 = 1\ s^{-1}$

=> $A_{ss} = 10, B_{ss} = 10$

Gillespie Algorithm

(a4) Generate two random numbers r_1, r_2 uniformly distributed in $(0,1)$.

(b4) Compute the propensity functions of each reaction by $\alpha_1 = A(t)(A(t)-1)k_1$, $\alpha_2 = A(t)B(t)k_2$, $\alpha_3 = k_3$ and $\alpha_4 = k_4$. Compute $\alpha_0 = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4$.

(c4) Compute the time when the next chemical reaction takes place as $t+\tau$ where

$$\tau = \frac{1}{\alpha_0} \ln \left[\frac{1}{r_1}\right]. \tag{2.29}$$

(d4) Compute the number of molecules at time $t+\tau$ by

$$A(t+\tau) = \begin{cases} A(t)-2 & \text{if } 0 \le r_2 < \alpha_1/\alpha_0; \\ A(t)-1 & \text{if } \alpha_1/\alpha_0 \le r_2 < (\alpha_1+\alpha_2)/\alpha_0; \\ A(t)+1 & \text{if } (\alpha_1+\alpha_2)/\alpha_0 \le r_2 < (\alpha_1+\alpha_2+\alpha_3)/\alpha_0; \\ A(t) & \text{if } (\alpha_1+\alpha_2+\alpha_3)/\alpha_0 \le r_2 < 1; \end{cases} \tag{2.30}$$

$$B(t+\tau) = \begin{cases} B(t) & \text{if } 0 \le r_2 < \alpha_1/\alpha_0; \\ B(t)-1 & \text{if } \alpha_1/\alpha_0 \le r_2 < (\alpha_1+\alpha_2)/\alpha_0; \\ B(t) & \text{if } (\alpha_1+\alpha_2)/\alpha_0 \le r_2 < (\alpha_1+\alpha_2+\alpha_3)/\alpha_0; \\ B(t)+1 & \text{if } (\alpha_1+\alpha_2+\alpha_3)/\alpha_0 \le r_2 < 1; \end{cases} \tag{2.31}$$

Then continue with step (a4) for time $t+\tau$.

Stochastic Simulation

$A + A \xrightarrow{k_1} \varnothing$ $A + B \xrightarrow{k_2} \varnothing$ $\varnothing \xrightarrow{k_3} A$ $\varnothing \xrightarrow{k_4} B$

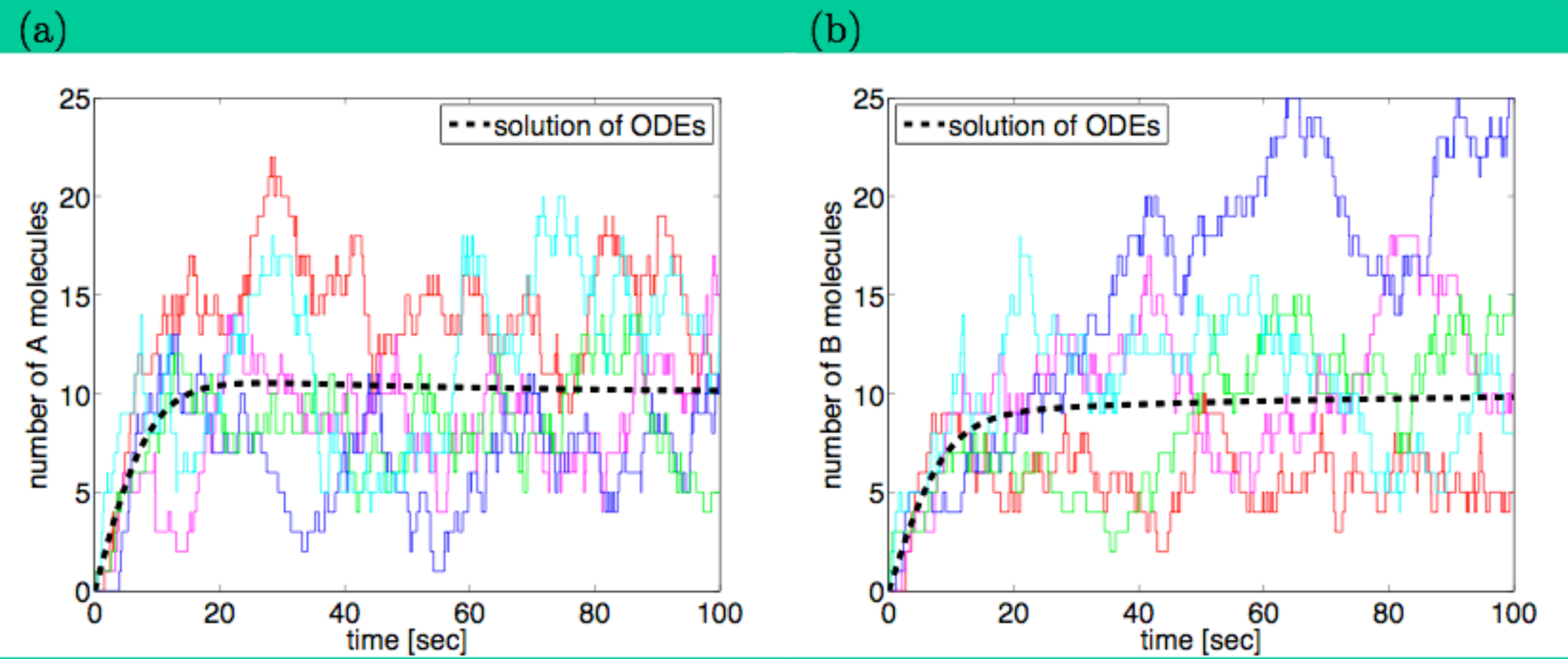


FIG. 2.3. *Five realizations of SSA (a4)–(d4). Number of molecules of chemical species A (left panel) and B (right panel) are plotted as functions of time as solid lines. Different colours correspond to different realizations. The solution of (2.33)–(2.34) is given by the dashed line. We use* $A(0) = 0$, $B(0) = 0$, $k_1 = 10^{-3}\ sec^{-1}$, $k_2 = 10^{-2}\ sec^{-1}$, $k_3 = 1.2\ sec^{-1}$ *and* $k_4 = 1\ sec^{-1}$.

Distribution of Stationary States

$A + A \xrightarrow{k_1} \varnothing$ $\quad A + B \xrightarrow{k_2} \varnothing$ $\quad \varnothing \xrightarrow{k_3} A$ $\quad \varnothing \xrightarrow{k_4} B$

$k_1 = 10^{-3}\ s^{-1}$ $\quad$ **$k_2 = 10^{-2}\ s^{-1}$** $\quad$ **$k_3 = 1.2\ s^{-1}$** $\quad$ **$k_4 = 1\ s^{-1}$**

Continuous model:
$A_{ss} = 10$, $\quad B_{ss} = 10$

<=>

From long–time Gillespie runs:
$\langle A\rangle = 9.6$, $\quad \langle B\rangle = 12.2$

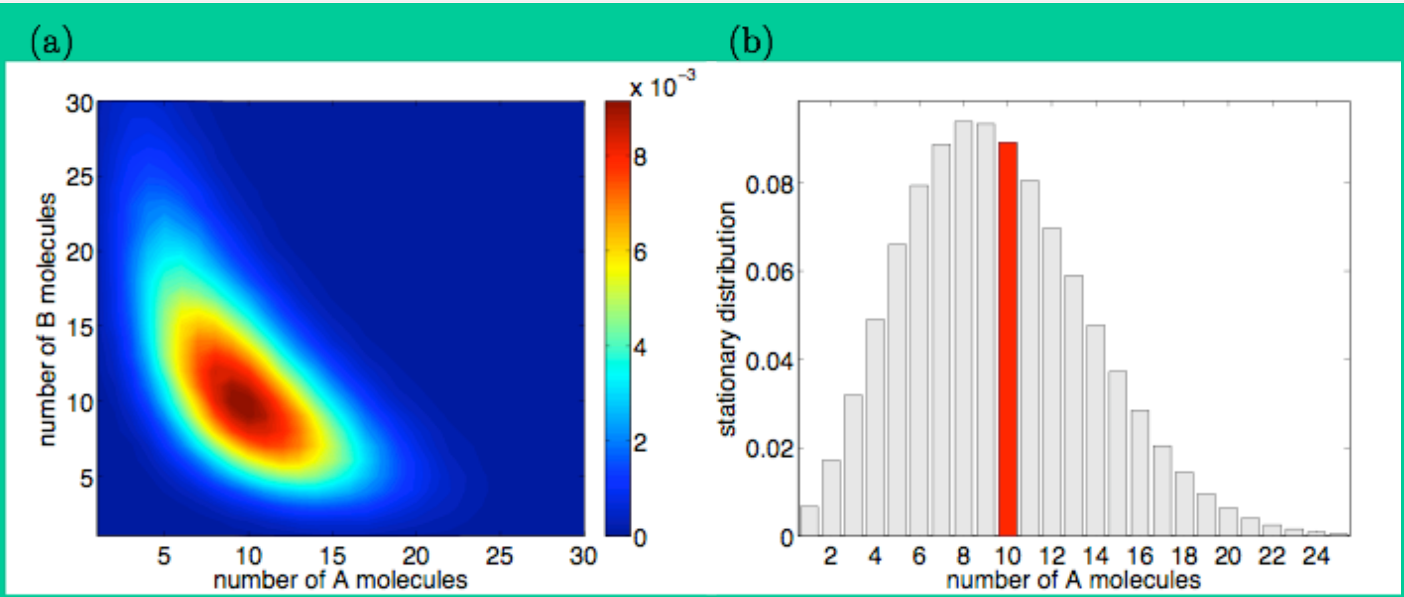


FIG. 2.4. (a) *Stationary distribution $\phi(n,m)$ obtained by long time simulation of (a4)–(d4) for $k_1 = 10^{-3}\ sec^{-1}$, $k_2 = 10^{-2}\ sec^{-1}$, $k_3 = 1.2\ sec^{-1}$ and $k_4 = 1\ sec^{-1}$.* (b) *Stationary distribution of A obtained by (2.35).*

Stochastic vs. Continuous

For many simple systems:

stochastic solution looks like noisy deterministic solution

Yet in some cases, stochastic description gives qualitatively different results

- swapping between two stationary states
- noise-induced oscillations
- Lotka-Volterra with small populations
- sensitivity in signalling

Two Stationary States

Reactions: $$2A \underset{k_2}{\overset{k_1}{\rightleftarrows}} 3A, \qquad \emptyset \underset{k_4}{\overset{k_3}{\rightleftarrows}} A$$

F. Schlögl, Z. Physik 253 (1972) 147–162

Rate equation: $$\frac{dA}{dt} = k_1 A^2 - k_2 A^3 + k_3 - k_4 A$$

With: $k_1 = 0.18\ \text{min}^{-1}$ $k_2 = 2.5 \times 10^{-4}\ \text{min}^{-1}$ $k_3 = 2200\ \text{min}^{-1}$ $k_4 = 37.5\ \text{min}^{-1}$

Stationary states: $A_{s1} = 100$, $A_{s2} = 400$ (stable) $A_u = 220$ (unstable)

=> Depending on initial conditions (A(0) <> 220),
the deterministic system goes into A_{s1} or A_{s2} (and stays there).

Two States – Stochastic

Erban, Chapman, Maini, arXiv:0704.1908v2

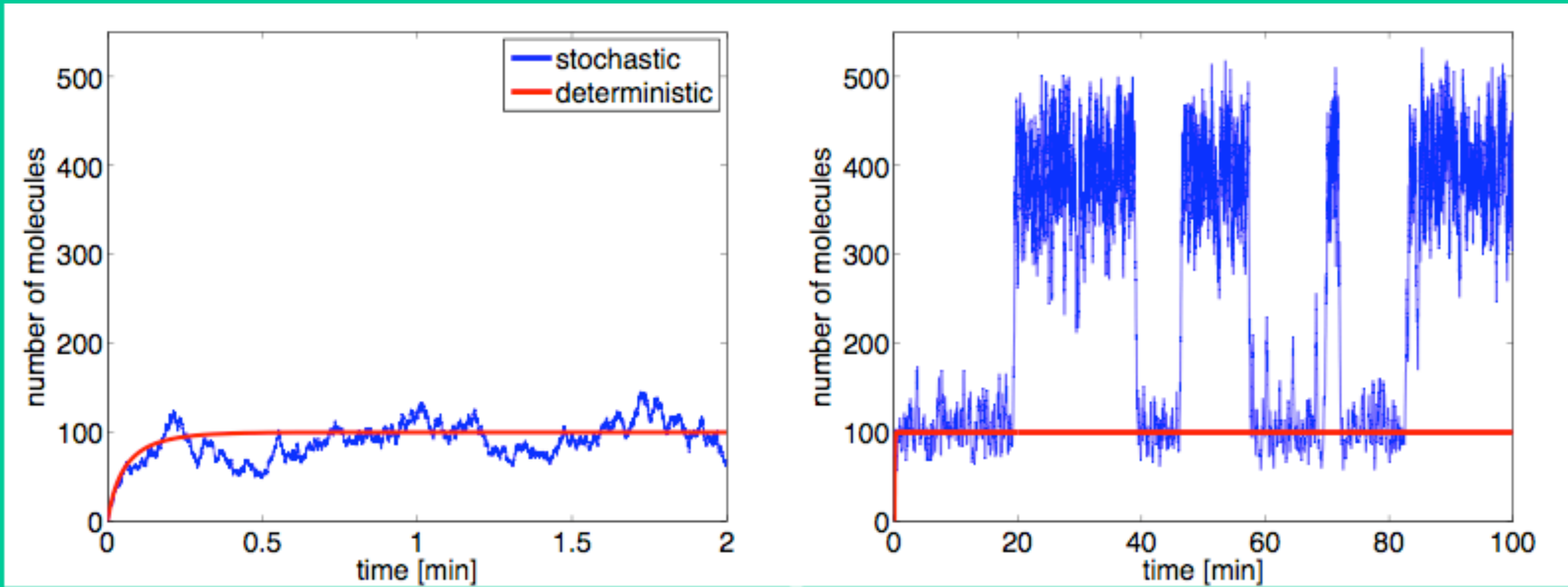


FIG. 5.1. *Simulation of (5.1). One realization of SSA (a5)–(d5) for the system of chemical reactions (5.1) (blue line) and the solution of the deterministic ODE (5.2) (red line).* (a) *The number of molecules of A as a function of time over the first two minutes of simulation.* (b) *Time evolution over 100 minutes.*

=> Fluctuations can drive the system from one stable state into another

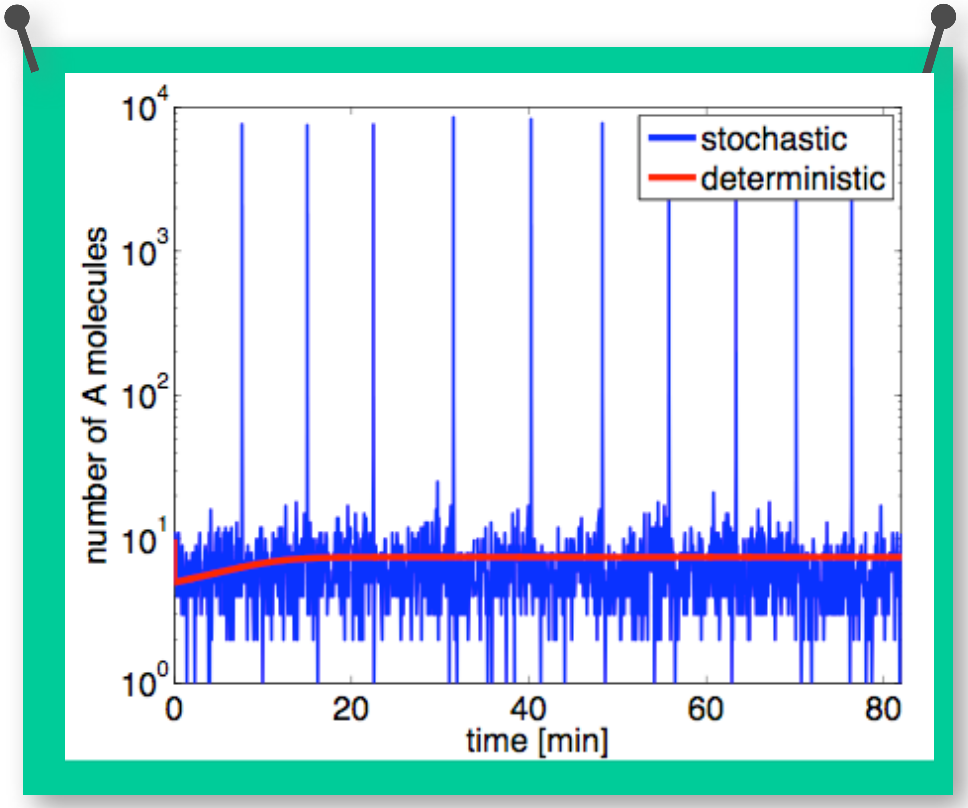
Self-Induced Stochastic Resonance

System $2A + B \xrightarrow{k_1} 3A$ $\varnothing \underset{k_3}{\overset{k_2}{\Longleftrightarrow}} A$ $\varnothing \xrightarrow{k_4} B$

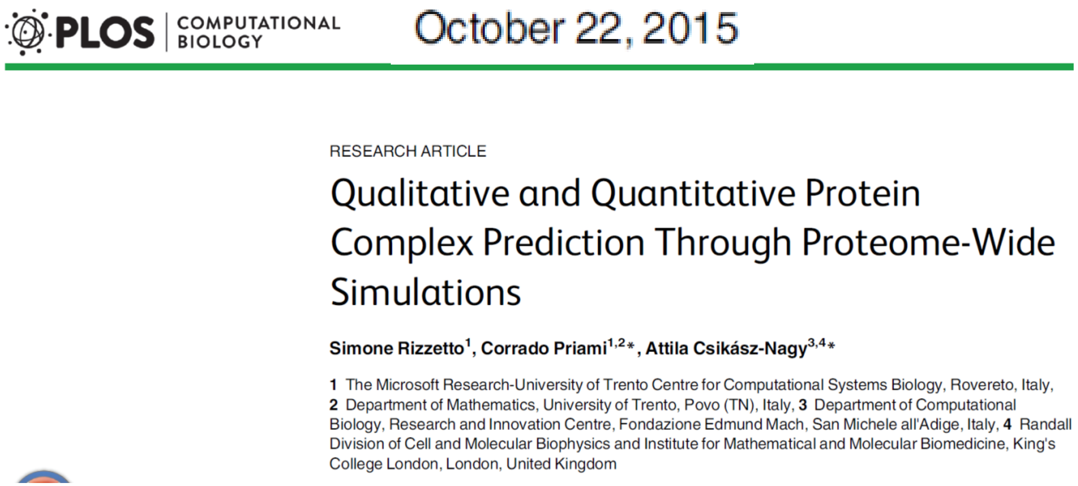
Compare the time evolution from initial state (A, B) = (10, 10) in deterministic and stochastic simulations.

=> deterministic simulation converges to and stays at fixed point (A, B) = (10, 1.1e4)

=> periodic oscillations in the stochastic model



Stochastic dynamics of PP complex assembly

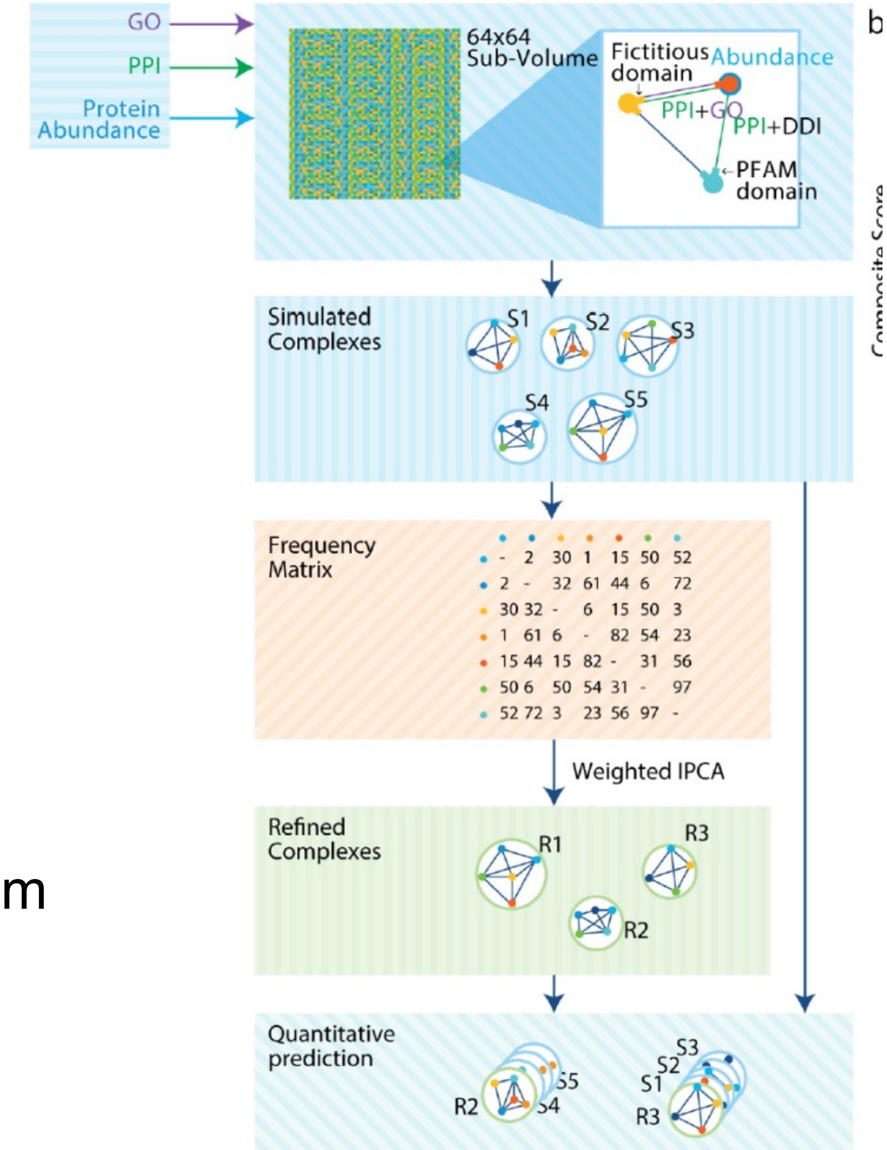

PLOS COMPUTATIONAL BIOLOGY October 22, 2015

RESEARCH ARTICLE

Qualitative and Quantitative Protein Complex Prediction Through Proteome-Wide Simulations

Simone Rizzetto[1], Corrado Priami[1,2]*, Attila Csikász-Nagy[3,4]*

1 The Microsoft Research-University of Trento Centre for Computational Systems Biology, Rovereto, Italy, 2 Department of Mathematics, University of Trento, Povo (TN), Italy, 3 Department of Computational Biology, Research and Innovation Centre, Fondazione Edmund Mach, San Michele all'Adige, Italy, 4 Randall Division of Cell and Molecular Biophysics and Institute for Mathematical and Molecular Biomedicine, King's College London, London, United Kingdom



Stochastic simulations,

4096 compartments on 2D lattice,

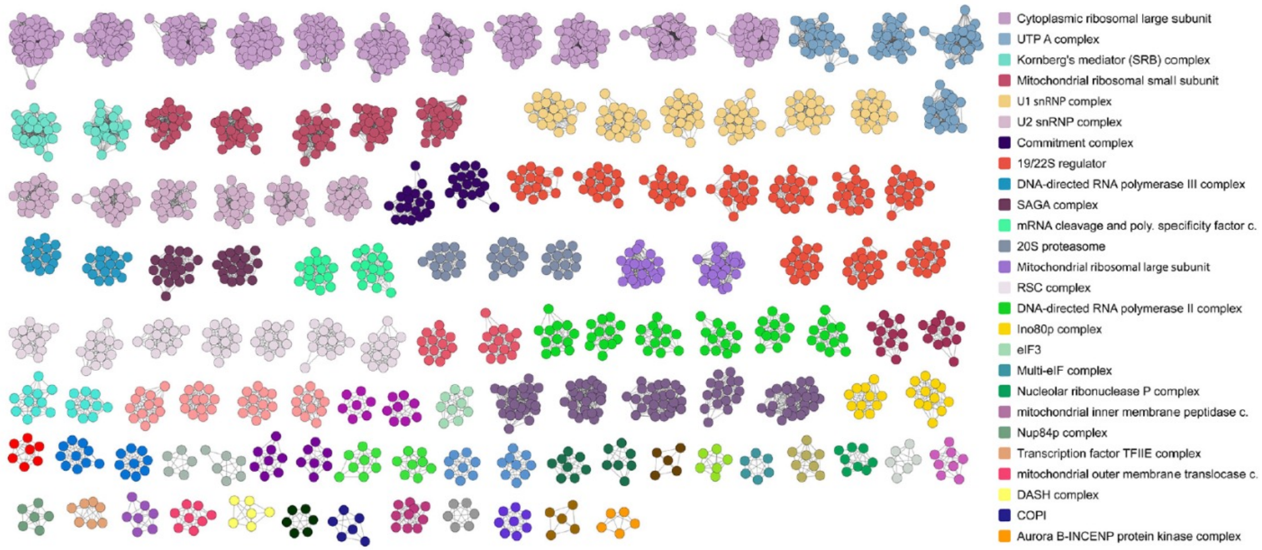
association rates set to 100,

dissociation rates set to 1

CUDA implementation of Gillespie algorithm

Gillespie-type simulation of PP complex assembly

Protein complex	Experimental abundance	SiComPre predicted abundance
Ribosome	187,000	50,000
RNA polymerase I	1,500	2,200
RNA polymerase II	3,000	3,900
RNA polymerase III	150	144
Nuclear Pore Complex	200	462
Eisosome	75	80,372
Nucleosome	57,000	90,300
Anaphase-Promoting Complex	3,000	1,406



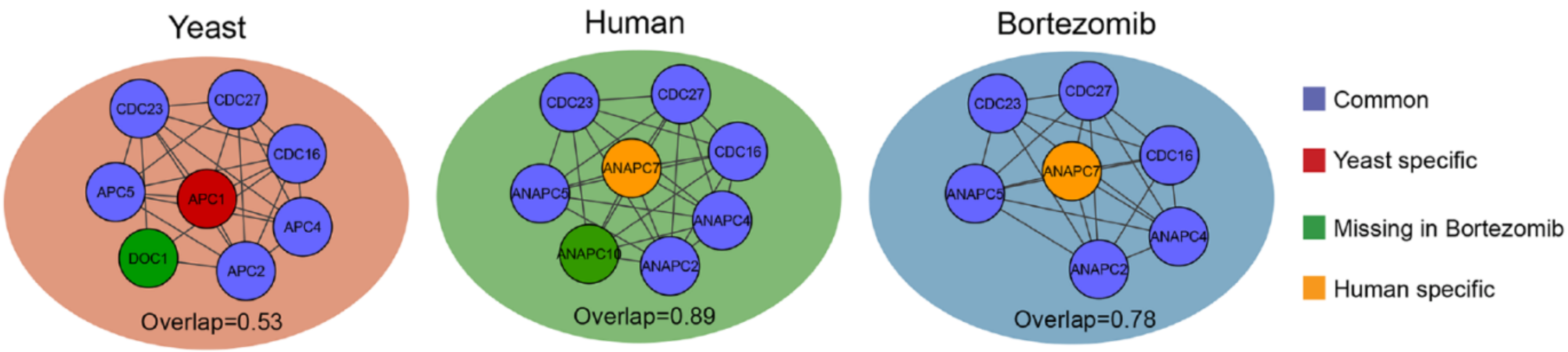


Fig 4. Variations in SiComPre predicted anaphase promoting complexes. The predicted structures of the APC complex in yeast, human and human after Bortezomib treatment. The reported overlap scores were calculated by comparing to the reference protein complexes discussed above. The lower score observed for the yeast is due to the larger APC complex size found in yeast [9].

Summary

- Mass action kinetics
  => solving (integrating) differential equations for time-dependent behavior
  => Forward-Euler: extrapolation, time steps

- Stochastic Description
  => why stochastic?
  => Gillespie algorithm
  => different dynamic behavior