

BIOE 498 / BIOE 599: Computational Systems Biology for Medical Applications

CSE 599V: Advancing Biomedical Models

Lecture 6: Model Solutions & System Response

Joseph L. Hellerstein*

Herbert Sauro**

*eScience Institute, Computer Science & Engineering

**BioEngineering



Model Solvers

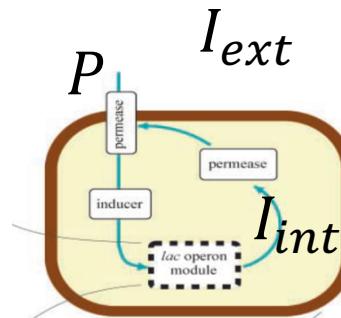
- Motivating example – lac operon
- Alternative methods
- Software
- In class exercise



Motivation: Model of the lac Operon

$$\frac{dI_{int}}{dt} = a(I_{ext} - I_{int})P - bI_{int}$$

$$\frac{dP}{dt} = cI_{int} - eP$$



I_{ext} = concentration of external inducer (lactose)

I_{int} = concentration of internal inducer

P = concentration of permease

Given values of I_{ext} , $I_{int}(0)$, $P(0)$, parameters a, b, c, e

Find $I_{int}(t)$, $P(t)$



Initial Value Problem (IVP)

Given

- N ordinary differential equations in N variables
- Values for the N variables at time 0

Find

- Values of the N variables over time

Notation

x_n is the value of the n -th variable at time t

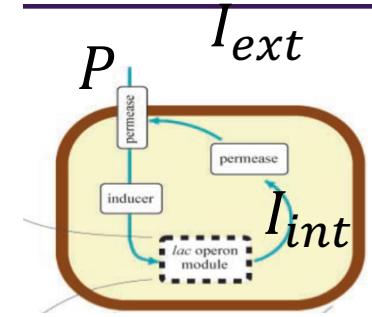
$\frac{dx_n}{dt}$ is the derivate of the n -th variable w.r.t. time

$f_n(\cdot)$ is an analytic function (infinitely differentiable)

$$\frac{dx_n}{dt} = f_n(x_1, \dots, x_N)$$



IVP for lac Operon Model



$$\frac{dI_{int}}{dt} = a(I_{ext} - I_{int})P - bI_{int}$$

$$\frac{dP}{dt} = cI_{int} - eP$$

- What is N ?
2
- What is f_n for $\frac{dI_{int}}{dt}$?
 $a(I_{ext} - I_{int})P - bI_{int}$
- What are reasonable initial values for I_{int}, P ?
 - Something small, but not 0.



Euler Algorithm for IVP

Given: Differential equations and initial values for n variables

$$\frac{dx_n(t)}{dt} = f_n(x_1(t), \dots, x_N(t)), \quad x_n(0)$$

Find Values of variables at time T

$$x_n(T)$$

Key consideration: How big is dt

1. For $t = 0..T-1$

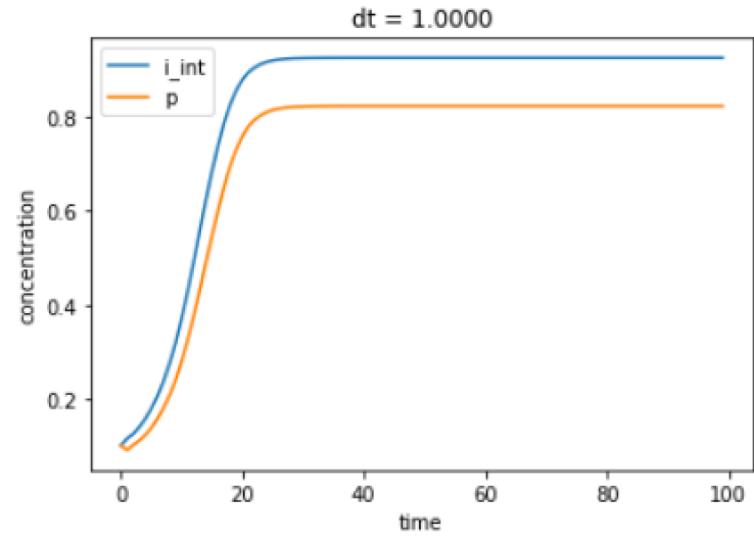
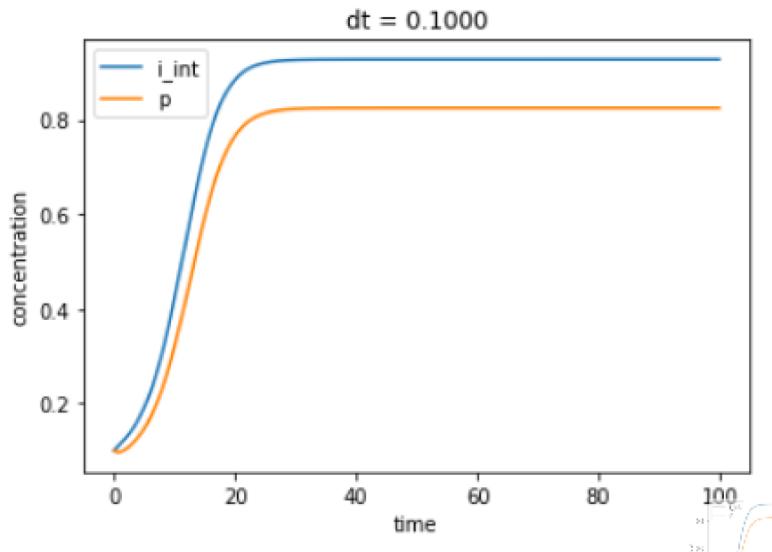
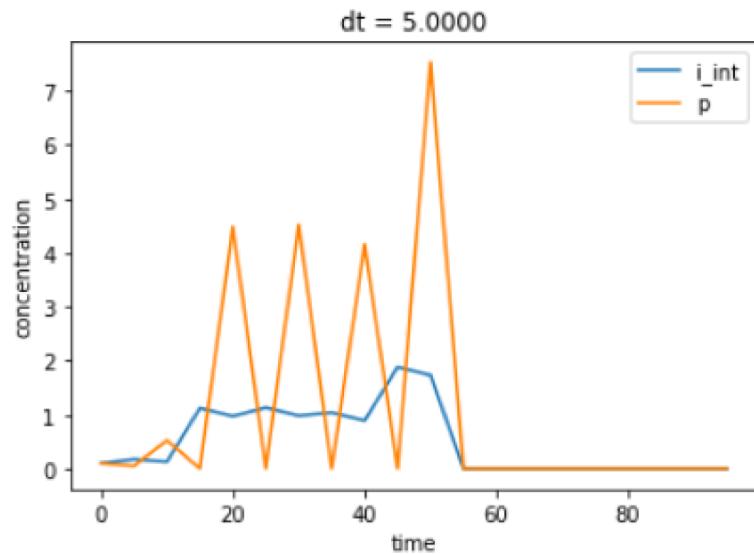
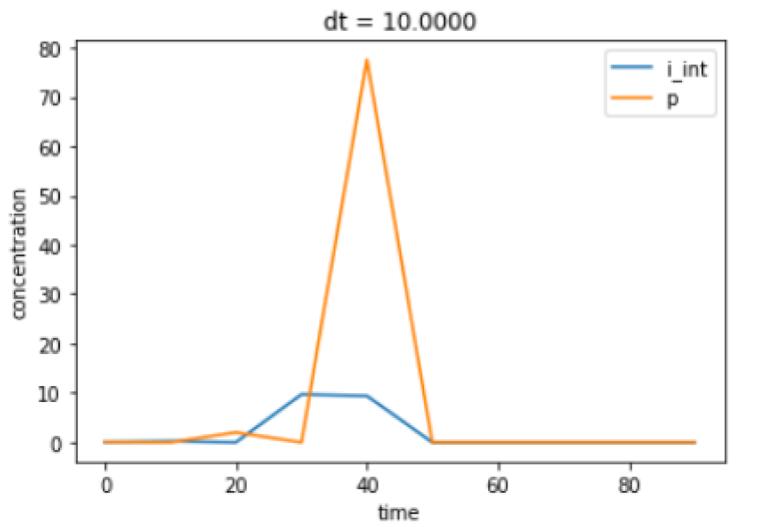
1. For $n = 0..N$

$$1. \quad \text{delta} = f_n(x_1(t), \dots, x_n(t))dt$$

$$2. \quad x_n(t + 1) = x_n(t) + \text{delta}$$



Euler for lac Operon Model



Requirement for a Numerical Solution

- The function is “smooth enough”
 - Two ways of expressing “smooth enough”
 - Lipschitz continuity
 - Smooth and uniformly monotone decreasing
- If smooth enough, then
 - a solution will exist and be unique
 - we will be able to approximate it accurately with a wide variety of method

Euler's method is only first-order accurate ($O(dt)$).



Variants of Euler's Method

- If we use

$$y'(t) = \frac{y(t) - y(t-h)}{h} - \frac{1}{2}hy''(\theta), \quad O(h) \quad h = dt$$

then we get the backward Euler method

Method 1

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}) :$$

- If we use

$$y'(t) = \frac{y(t+h) - y(t-h)}{2h} - \frac{1}{6}h^2y'''(\theta_{t,h}), \quad O(h^2)$$

then we get the midpoint method

Method 2

$$y_{n+1} = y_{n-1} + 2hf(t_n, y_n).$$

6.5 Single-step Method: Runge-Kutta

- The Runge-Kutta family of methods is one of the most popular families of accurate solvers for initial value problems.

Recall the usual predictor-corrector formulation of the trapezoid method:

$$\bar{y}_{n+1} = y_n + hf(t_n, y_n)$$

$$y_{n+1} = y_n + \frac{1}{2}h [f(t_{n+1}, \bar{y}_{n+1}) + f(t_n, y_n)].$$

We can directly substitute the predictor into the corrector to write this as a single recursion:

$$y_{n+1} = y_n + \frac{1}{2}h [f(t_{n+1}, y_n + hf(t_n, y_n)) + f(t_n, y_n)]. \quad (6.44)$$



Variants of Euler's Method

- By integrating the differential equation:

$$y(t+h) = y(t) + \int_t^{t+h} f(s, y(s)) ds. \quad (6.23)$$

and apply the trapezoid rule to (6.23) to get

$$y(t+h) = y(t) + \frac{1}{2}h[f(t+h, y(t+h)) + f(t, y(t))] - \frac{1}{12}h^3 y'''(\theta_{t,h}), \quad O(h^3)$$

where $\theta_{t,h} \in [t, t+h]$, and we remind the reader that

$$\underline{f(t, y(t)) = y'(t)} \quad \Rightarrow \quad \frac{d^2}{dt^2} f(t, y(t)) = y'''(t).$$

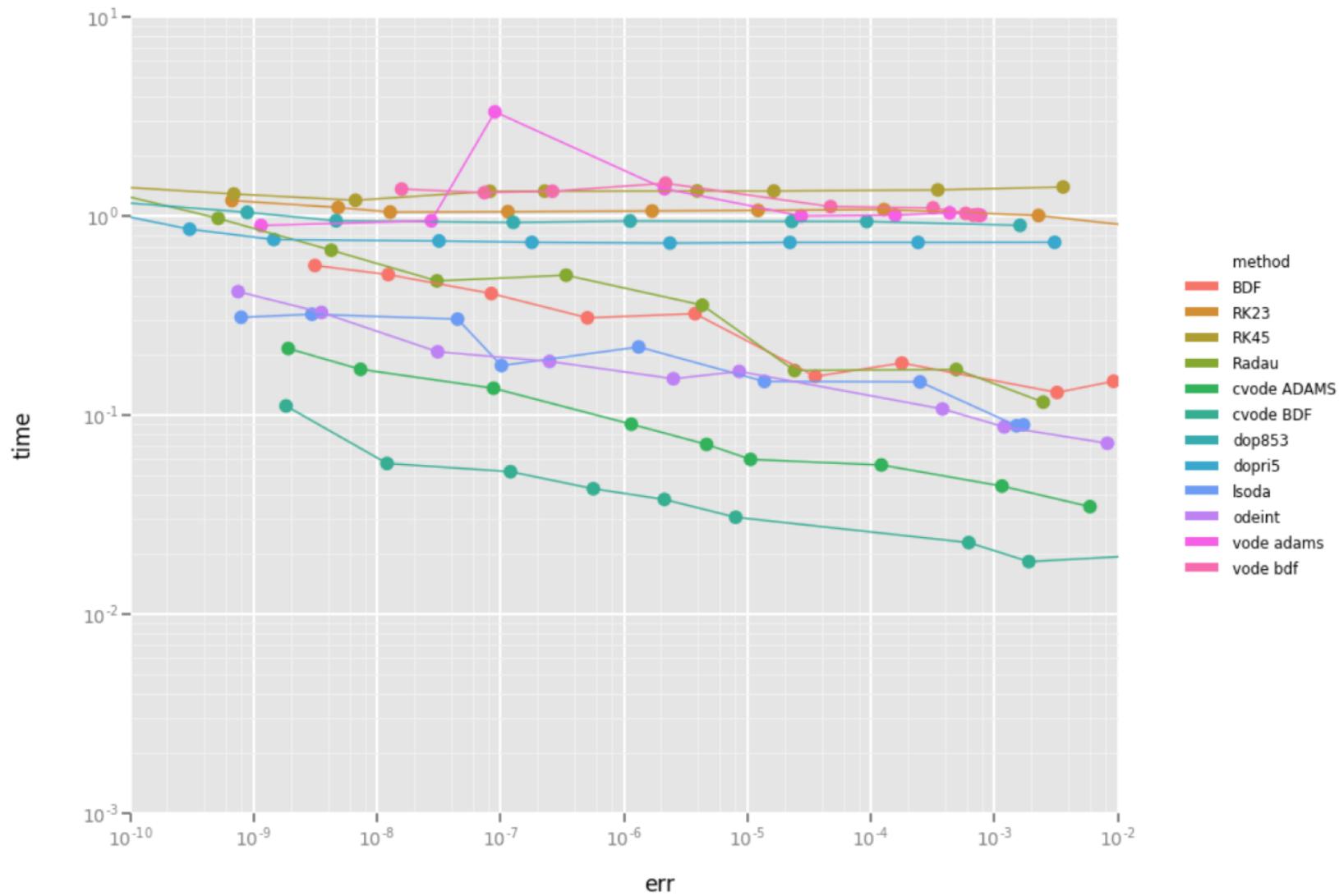
Thus

Method 5

$$y_{n+1} = y_n + \frac{1}{2}h(f(t_{n+1}, y_{n+1}) + f(t_n, y_n)). \quad (6.25)$$



Performance of Python Solvers



MATLAB Solver

The `ode45` solver implements a variable step size Runge-Kutta method by using the Dormand-Prince method. The basic syntax for `ode45` is:

```
[t,y] = ode45(@myModel, [t0, tend], yo, [], p);
```

where

`myModel` is the function containing the differential equations.

`t0, tend` are the initial and final values for the independent variable, t .

`yo` is a vector of initial conditions.

`p` is the set of parameters for the model, and can be any size.

```
function dy = myModel(t, y, p)
dy = zeros (2,1);
vo = p(1);
k1 = p(2);
k2 = p(3);
dy(1) = vo - k1*y(1);
dy(2) = k1*y(1) - k2*y(2);

p = [10, 0.5, 0.35]
y0 = [0, 0]
[t, y] = ode45 (@myModel, [0, 20], y0, [], p)
```



Python Solver

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint

vo = 10
k1 = 0.5
k2 = 0.35

# Declare the model
def myModel(y, t):

    dy0 = vo - k1*y[0]
    dy1 = k1*y[0] - k2*y[1]
    return [dy0, dy1]

time = linspace(0.0, 20.0, 100)
yinit = array([0.0, 0.0])
y = odeint (myModel, yinit, time)
```



In Class Exercise

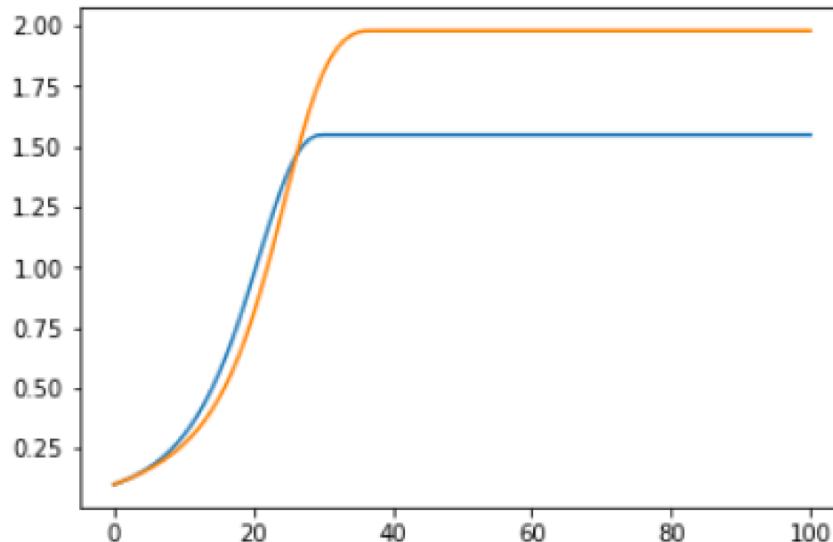
Numerically Evaluate lac Operon Model

$$\frac{dI_{int}}{dt} = a(I_{ext} - I_{int})P - bI_{int}$$

$$\frac{dP}{dt} = cI_{int} - eP$$

$$a = 0.5; b = 0.03; c = 0.8; d = 0.9; I_{ext} = 1.0$$

$$I_{int}(0) = 0.1; P(0) = 0.1$$

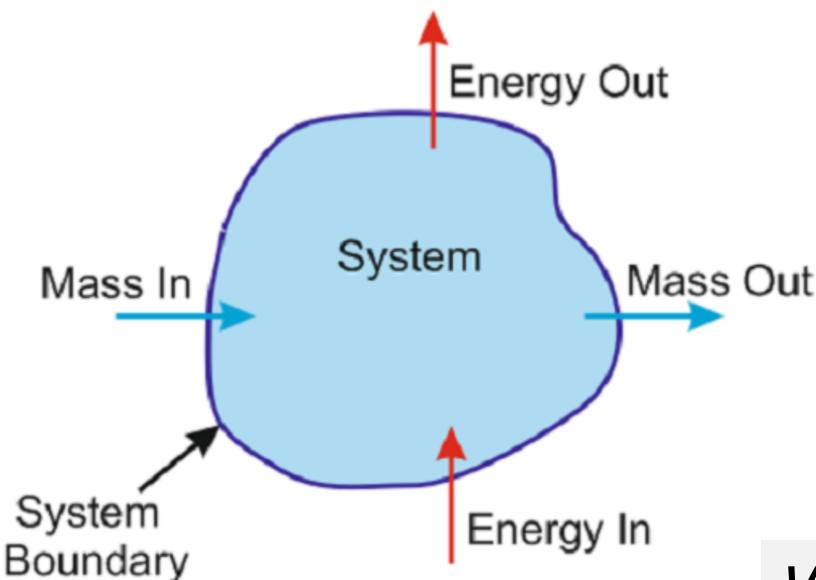


System Responses

- Thermodynamics in brief
- Reversible system in equilibrium
- Open system in steady state
- Solutions in Tellurium



Thermodynamics: Key Definitions

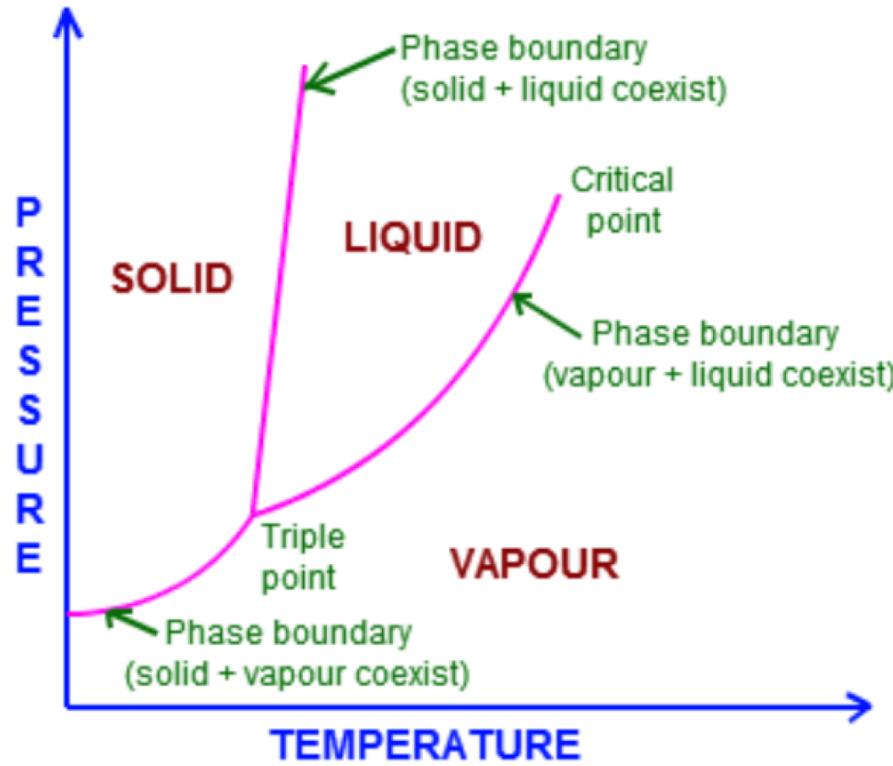


What type of system is a cell?

System Types

	No Mass Flow	Mass Flow
No Energy Flow	Isolated	X
Energy Flow	Closed	Open

Thermodynamic Equilibrium



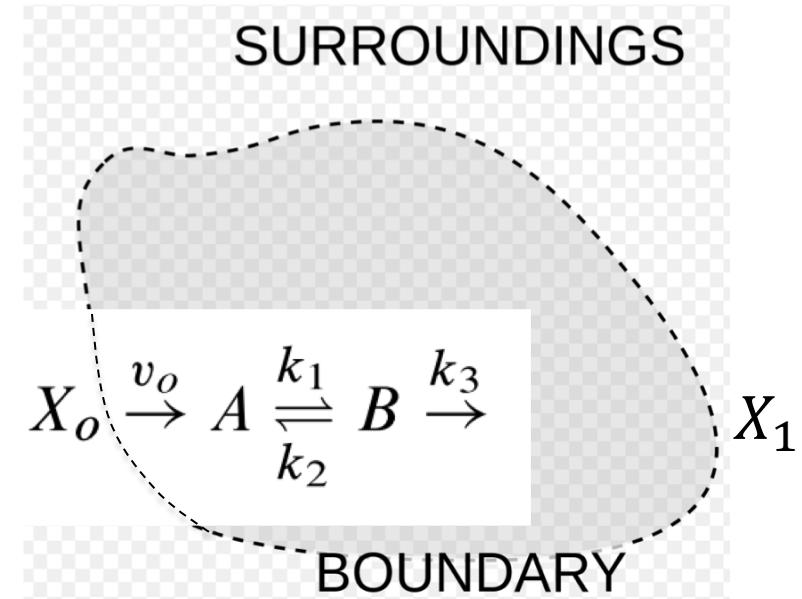
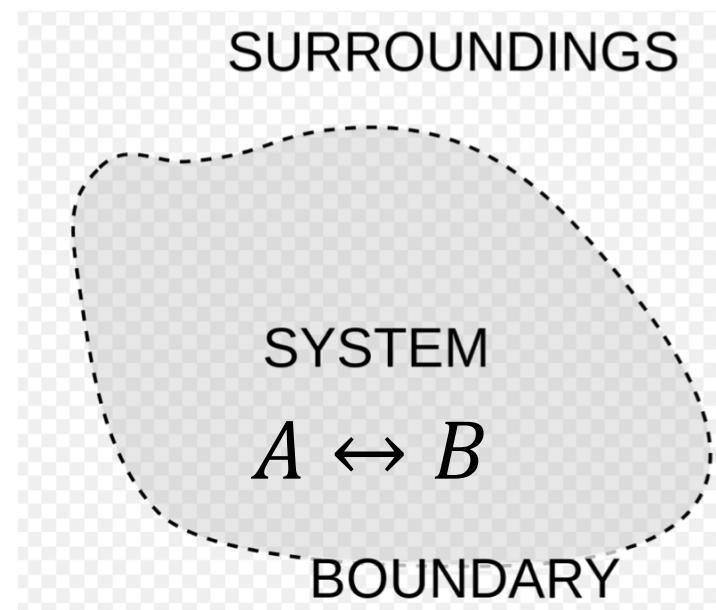
A system is in thermodynamic equilibrium if no simple process lowers its energy.



Chemical Systems

A chemical system is in **equilibrium** if no reaction lowers its energy and hence no net reaction take place.

An chemical system is in **steady state** if no concentrations are changing as time goes to infinity.



Equilibrium and Steady State

A chemical system is in **equilibrium** if no reaction lowers its energy and hence no net reaction take place.

An chemical system is in **steady state** if no concentrations are changing as time goes to infinity.

- Equilibrium implies steady state.
 - No reactions and so no change in concentrations
- Steady state does NOT imply equilibrium
 - Reactions may still happen



Computing Species Concentrations

- Solve simple differential equations for chemical systems
- Given an initial amount of A , A_0 , find A , B over time.
 $A \rightarrow B$
- Equations

$$\frac{dA}{dt} = -kA \quad \frac{dB}{dt} = kA$$



Solution by Separation of Variables

$A \rightarrow B$

$$\frac{dA}{A} = -kdt$$

$$\int \frac{dA}{A} = - \int kdt$$

$$\ln(A) = -kt + C$$

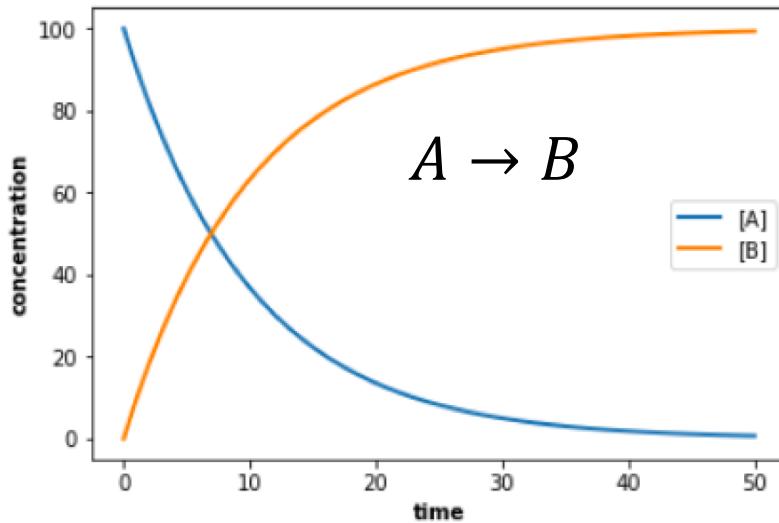
So, $A = C'e^{-kt}$, where $C' = e^C$

We find C' by observing that $A(0) = A_0 = C'e^{-(k)(0)} = C'$

Result: $A = A_0e^{-kt}$



Interpreting the Solution



$$A = A_0 e^{-kt}$$

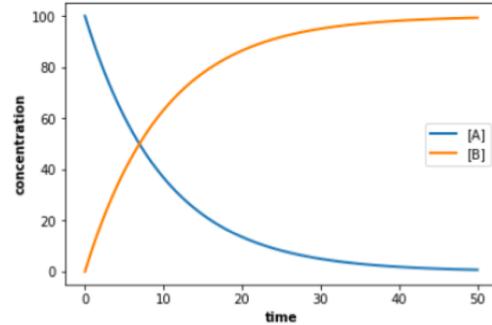
- What is A at time 0?
- How much A is there at steady state, $t \rightarrow \infty$?
- How much B is there at steady state?
- If we know $A(t)$, how do we find $B(t)$?



Solving in Tellurium

```
te.setDefaultPlottingEngine('matplotlib')
model = """
    model test
        species A, B;
        A = 100.0;
        B = 0.0;
        J1: A -> B; k1*A;
        k1 = 0.1;
    end"""

r = te.loada(model)
r.simulate(0, 50, 100)
r.plot(xtitle="time", ytitle="concentration")
```



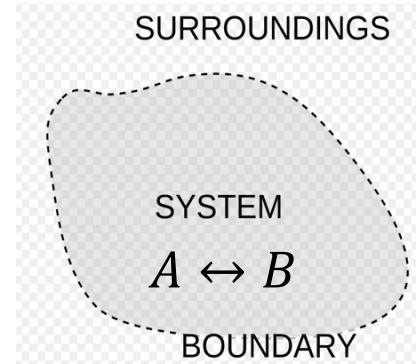
Reversible Reaction, Closed System

Given $A(0) = A_0, B(0) = B_0$

$$\frac{dA}{dt} = -k_1 A + k_2 B \quad \frac{dB}{dt} = k_1 A - k_2 B$$

Note that $B = B_0 + A_0 - A$

$$\frac{dA}{dt} = -k_1 A + k_2(B_0 + A_0 - A) = -(k_1 + k_2)A + k_2(A_0 + B_0)$$



Solving the differential equation:

Homogeneous eqn (function of A): $\frac{dA}{dt} = -(k_1 + k_2)A \equiv A(t) = C'e^{-(k_1+k_2)t}$

Particular soln (soln for the constant $k_2(A_0+B_0)$): Set $A = c$.

$$\text{Solve } \frac{dc}{dt} = -(k_1 + k_2)c + k_2(A_0 + B_0). \text{ So, } c = \frac{k_2}{k_1 + k_2}(A_0 + B_0)$$

Full soln (homogeneous + particular): $A(t) = \frac{k_2}{k_1 + k_2}(A_0 + B_0) + C'e^{-(k_1+k_2)t}$

Find C' using initial value: $A(0) = A_0 = \frac{k_2}{k_1 + k_2}(A_0 + B_0) + C' \quad C' = \frac{k_1 A_0 - k_2 B_0}{k_1 + k_2}$

Result (Use C'): $A(t) = \frac{k_2}{k_1 + k_2}(A_0 + B_0) + \frac{k_1 A_0 - k_2 B_0}{k_1 + k_2} e^{-(k_1+k_2)t}$

A Closer Look at the Solution to $A \leftrightarrow B$

$$A(t) = \frac{k_2}{k_1 + k_2} (A_0 + B_0) + \frac{k_1 A_0 - k_2 B_0}{k_1 + k_2} e^{-(k_1 + k_2)t}$$

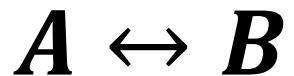
- What is the steady, $A(\infty)$? What is the transient?
- What is the initial reaction velocity $v_{initial} = \frac{dA}{dt} |_{t=0}$?

$$v_{initial} = k_1 A_0 - k_2 B_0$$

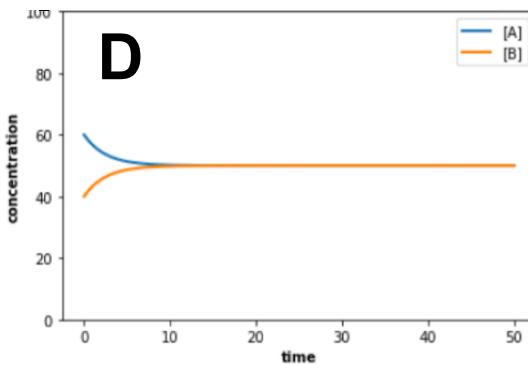
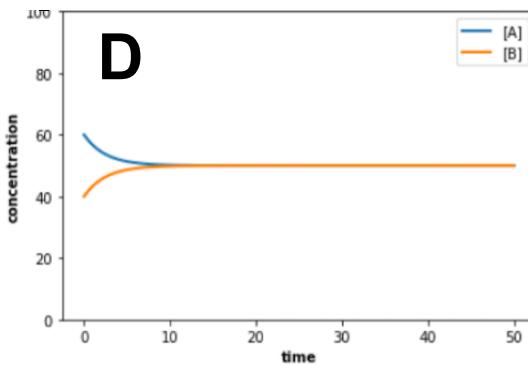
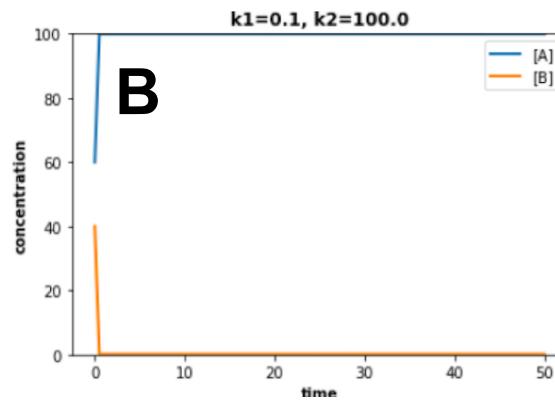
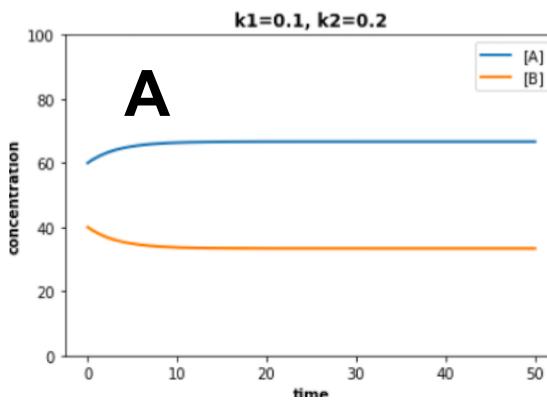
- What is $B(t)$?

$$\begin{aligned} B(t) &= A_0 + B_0 - A(t) \\ &= \frac{k_1}{k_1 + k_2} (A_0 + B_0) + \frac{-k_1 A_0 + k_2 B_0}{k_1 + k_2} e^{-(k_1 + k_2)t} \end{aligned}$$





$$A(t) = \frac{k_2}{k_1 + k_2} (A_0 + B_0) + \frac{k_1 A_0 - k_2 B_0}{k_1 + k_2} e^{-(k_1 + k_2)t}$$



Questions:

1. What are A_0, B_0 ?
2. Where is $k_1 > k_2$?
3. What is A_{EQ} ?

Solution In Tellurium

model test

```
species A, B;
```

```
A = 60.0;
```

```
B = 40.0;
```

```
J1: A -> B; k1*A;
```

```
J2: B -> A; k2*B;
```

```
k1 = 0.1;
```

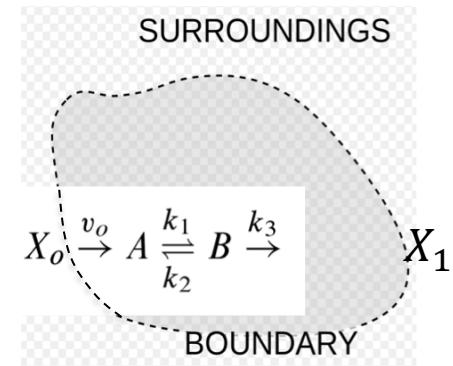
```
k2 = 0.2
```

```
end
```

Try it!



Reversible Reaction, Open System



$$A(t) = v_o \frac{1 - e^{-k_1 t}}{k_1}$$

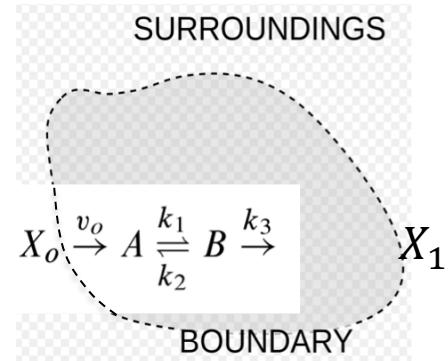
$$B(t) = v_o \frac{k_1 (1 - e^{-k_3 t}) + k_3 (e^{-k_1 t} - 1)}{k_3 (k_1 - k_3)}$$

(7.3)

(from text book)

- What is $A(0), B(0)$?
- What is $A(\infty), B(\infty)$?
- What is $\frac{dA}{dt} |_{t \rightarrow \infty}$?

Class Exercise



Write a Tellurium simulation for the above system

$$A_0 = 40; \quad B_0 = 60;$$

$$k_1 = 0.1; \quad k_2 = 0.2; \quad k_3 = 0.1 \quad v_{in} = 1$$

Questions

- What are the steady state values of A, B?
- At what time is A = B?
- How do the above change if k₁=0.2, k₂=0.1?



What You Should Know

- Numerical solutions to differential equations
 - Euler algorithm, how improve efficiency
 - How use python solver
- System response
 - Thermodynamics system types: isolated, closed, open
 - Transient and steady state responses
 - Steady state vs. equilibrium
 - Solutions in Tellurium

