### Euler’s Method

- Numerical simulation can approximate time evolution of $\mathbf{X}(t)$ assuming $\mathbf{X}(t_0) = \mathbf{x}_0$ (initial value problem).
- Initial instantaneous rate of change for each $S_i$ at time $t_0$:
  $$\frac{dX_i(t_0)}{dt} = f_i(t_0) \quad \text{where} \quad 1 \leq i \leq n.$$

- If the rate of change remains constant for all $t \geq t_0$, then
  $$X_i(t) = x_{i0} + \int_{t_0}^{t} f_i(t_0) \, dt.$$  

### Euler’s Method (cont)

- Not true in general, but may be reasonable to assume value remains close to $f_i(t_0)$ for some small time step $\Delta t$ (step size).
- With this assumption,
  $$X_i(t_1) \approx X_i(t_0) + f_i(t_0) \Delta t$$

- In general, for any $t_j = t_0 + j \Delta t$ where $j = 0, 1, 2, 3, \ldots$:
  $$X_i(t_{j+1}) \approx X_i(t_j) + f_i(t_j) \Delta t$$

- This algorithm is known as Euler’s Method.

### Original Function

- Using the fundamental theorem of calculus, exact solution for each species, $S_i$, must satisfy:
  $$X_i(t_{j+1}) = X_i(t_j) + \int_{t_j}^{t_{j+1}} f_i(t_j) \, dt$$

- Can be computed exactly if we know the area under the curve.
Euler’s Method

- Euler’s method assumes $f_t(X(t_j))$ is constant.
- Result is an area of $f(X(t_j))\Delta t$.

\[
\begin{align*}
\text{Area of this trapezoid is:} & \\
& \frac{1}{2} [f(t_n, X_n) + f(t_{n+1}, X_{n+1})] \Delta t
\end{align*}
\]

- Drawback: only considers value of $f$ at the start of the step.

Midpoint Method (Second-Order Runge-Kutta)

- Approximates rate of change in $\Delta t$ interval using rate at the midpoint:

\[
X(t_{j+1}) \approx X(t_j) + \left[f(X(t_j)) + \frac{1}{2} f(X(t_j) + f(X(t_j)) \Delta t) \right] \Delta t
\]

\[
\begin{align*}
\text{Result is an area of } f(X(t_j)), \ldots, f(X(t_{n-1})) & .
\end{align*}
\]

This is performed by the following equations:

\[
\alpha_1 = f(X(t_j))
\]

\[
\alpha_2 = f(X(t_j) + \Delta t \alpha_1)
\]

\[
\alpha_3 = f(X(t_j) + \Delta t \alpha_2)
\]

\[
\alpha_4 = f(X(t_j) + \Delta t \alpha_3)
\]

\[
X(t_{j+1}) = X(t_j) + \frac{\Delta t}{6} [\alpha_1 + 2 \alpha_2 + 2 \alpha_3 + \alpha_4]
\]

Trapezoid Method

- Area of this trapezoid is:

\[
\frac{1}{2} \left[ f(t_n, X_n) + f(t_{n+1}, X_{n+1}) \right] \Delta t
\]

- Do not know the value of $X(t_{j+1})$, so it must be approximated:

\[
X(t_{j+1}) \approx X(t_j) + \frac{1}{2} \left[ f(X(t_j)) + f(X(t_j) + f(X(t_j)) \Delta t) \right] \Delta t
\]

where $f(X(t_j))$ returns vector $\{f_0(X(t_j)), \ldots, f_n(X(t_j))\}$.

Adaptive Stepsize Control

- To obtain good results, should modify $\Delta t$ during the simulation.
- Simulation should slow down when rates are changing rapidly and speed up when the rates are changing slowly.
- With the step doubling approach, $X(t_{j+1})$, is found in one $\Delta t$ step and $X(t_{j+1})$ is found by taking two $\Delta t/2$ steps.
- Difference represents an estimate of the error as defined below:

\[
\text{error} = \text{max} \{ |X(t_{j+1}) - X(t_{j+1})| \}
\]

where max finds the maximum error in the vector.

Adaptive Stepsize Control (cont)

- This error estimate can then be used to determine a new stepsize:

\[
\Delta t = 0.9 \times \Delta t \times \frac{\text{tol}}{\text{err}}
\]

- If err > tol, simulation step should be performed again using new $\Delta t$.
- If err < tol, simulation step accepted and next step uses new $\Delta t$.
- $X(t_{j+1})$ is used as new state as it is more accurate.
Portion of the Phage λ Model

- Reactions for CII production and degradation:
  - 2CI → KCI
  - OR + CI2 → OR2 CI2
  - OR + RNAP → OR12 · RNAP
  - OR2 + 2CI2 → OR2 · 2CI2
  - OR12 · RNAP → OR12 · RNAP + 10CII
  - CII + P1 → CII · P1
  - CII · P1 → KI

- ODE model for CII using the Law of Mass Action:
  \[ \frac{d[CII]}{dt} = 10K_{RNAP}(OR_{12} · RNAP) + k_{c}[CII · P1] - k_{d}[CII |P1] \]

ODE Model Simplifications

- Simplifications are made to reduce model size and complexity.
- Rate of production of an enzyme (or gene) often described with a sigmoid function, such as the hill function:
  \[ f(x) = \frac{K^nx^n}{1 + K^nx^n} \] OR \[ \frac{1}{1 + K^nx^n} \]

where 1/K is the value of x where f(x) is 1/2.

Model for Repression

- n molecules of repressor R bind to the operator O.
  - ORn + nR \rightarrow ORnO
  - ORnO \rightarrow \frac{k_2}{k_1} R_nO

- Using the Law of Mass Action:
  \[ \frac{d[R]}{dt} = k_{1}[R_n] - k_{1}[R]^n \]
  \[ \frac{d[O]}{dt} = k_{2}[R_nO] - k_{2}[R][O] \]

The Fraction \([R]\) of Free Operator
**Model for Activation**

- $n$ molecules of activator $A$ bind to the operator $O$.
  
  \[
  nA \xrightarrow{k_1} A_n
  \]

- Using the Law of Mass Action:
  
  \[
  \frac{d[A]}{dt} = k_{-1} [A_n] - k_1 [A]^n
  \]

- Using assumptions and Equations 3 and 4:
  
  \[
  f([A]) = \frac{K_1 [A]^n}{1 + K_1 [A]^n}
  \]

**Model for Activation (cont)**

- Assuming reactions are rapid (i.e. $\frac{d[A]}{dt} = \frac{d[A]}{dt} = 0$):
  
  \[
  [A_0] = K_1 [A]^n
  \]

  \[
  [A_0] = K_1 [A]^n
  \]

- Reduced reactions for CII production and degradation:

**Portion of the Phage λ Model**

- Reactions for CII production and degradation:
  
  \[
  2CI \xrightarrow{K_{OR12}} CI_2
  \]
  \[
  OR + CI_2 \xrightarrow{K_{OR2}} CI_2
  \]
  \[
  OR + 2CI \xrightarrow{K_{OR2}} CI_2 \]
  \[
  OR + CI_2 \xrightarrow{K_{OR2}} CI_2 \]

**Reduction of the Phage λ Model**

- Reduced reactions for CII production and degradation:
  
  \[
  O_R + CI + RNAP \xrightarrow{f(CI, RNAP)} 2CI + P_1
  \]

- Reduced ODE model for CII using the Law of Mass Action:
  
  \[
  \frac{d[Cl]}{dt} = 10 k_{OR12} [OR + CI_2 \cdot P_1] - k_0 [Cl] [P_1]
  \]

**Reduced ODE model for CII using the Law of Mass Action:**

- Where
  
  \[
  f(CI, RNAP) = \frac{K_{OR12} [RNAP]}{1 + K_{OR12} [CI]^2 + K_{OR12} [RNAP]^2 + K_{OR12} [CI] [RNAP]}
  \]

  \[
  g(CI) = \frac{K_0 [Cl]}{1 + K_0 [Cl]}
  \]

**Reduced ODE model for CII using the Law of Mass Action:**

- Where
  
  \[
  f(CI) = \frac{1}{1 + K_0 [Cl]^2}
  \]

  \[
  g(CI) = \frac{K_0 [Cl]}{1 + K_0 [Cl]}
  \]
Qualitative ODE Analysis

- Evolution of systems over time.
- Analyze the qualitative behavior of non-linear systems.
- Determine how a trajectory will behave based on any initial condition.

One Dimensional Systems

- Can visualize as a flow on a line.
- Vector fields are a simple graphical technique to analyze ODEs.
- Equilibrium points are points where there is no flow.
- Stability determines the direction of flow near an equilibrium point.

Bifurcation

- Qualitative changes in dynamics.
- Bifurcation point is the parameter values at which a bifurcation happens.
- Shows how systems behave as control parameters are varied.

Saddle-Node Bifurcation

\[
\frac{dx}{dt} = -r + x^2
\]
Saddle-Node Bifurcation

\[ \frac{dx}{dt} = -r + x^2 \]

Transcritical Bifurcation

\[ \frac{dx}{dt} = -rx + x^2 \]
Transcritical Bifurcation

\[ \frac{dx}{dt} = -rx + x^2 \]

Pitchfork Bifurcation

\[ \frac{dx}{dt} = rx - x^3 \]
How can we analyze a two dimensional system?

\[
\frac{dx}{dt} = x^2 + y - 3 \\
\frac{dy}{dt} = y
\]

- Use a phase plane diagram.
- Sketch the nullclines.
- Find the equilibrium points.
- Determine stability of the equilibrium points.

Sources
- Web page for ODE solvers.
- Numerical recipes in C.