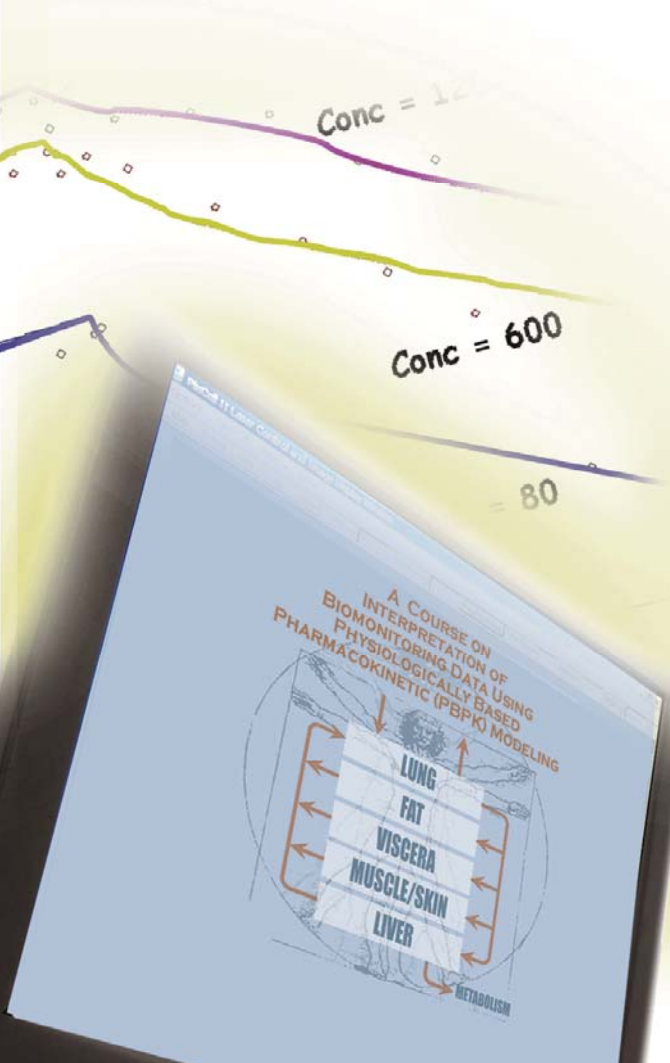


Simulation Language for PBPK Modeling

Center for Human Health Assessment
A Course on Physiologically Based Pharmacokinetic (PBPK)
Modeling and Risk Assessment

February 11 – February 15, 2008



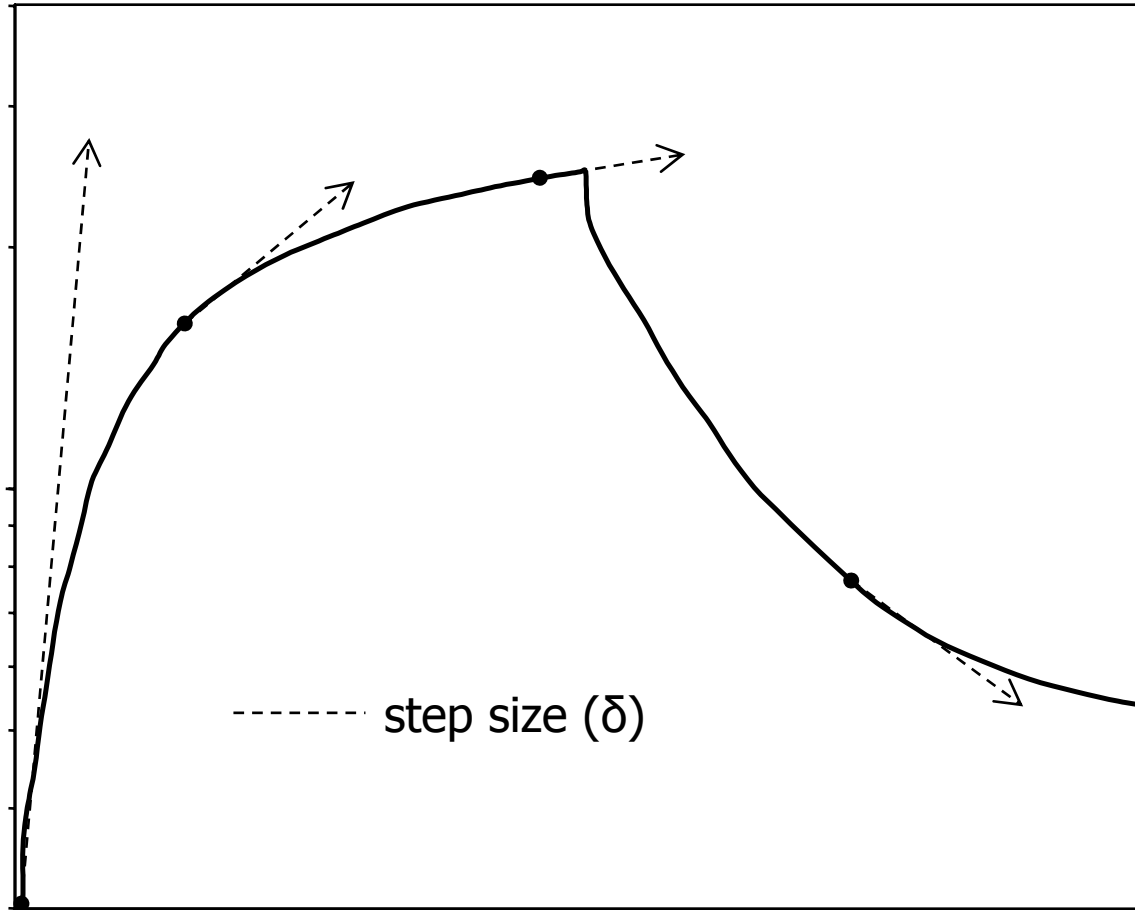
Numerical Integration Algorithms: “Inching along in the Dark”

Mathematically, PBPK Models represent a system of ordinary differential equations (ODEs) defining an initial value problem.

A number of simulation languages are available to solve this type of problem numerically.

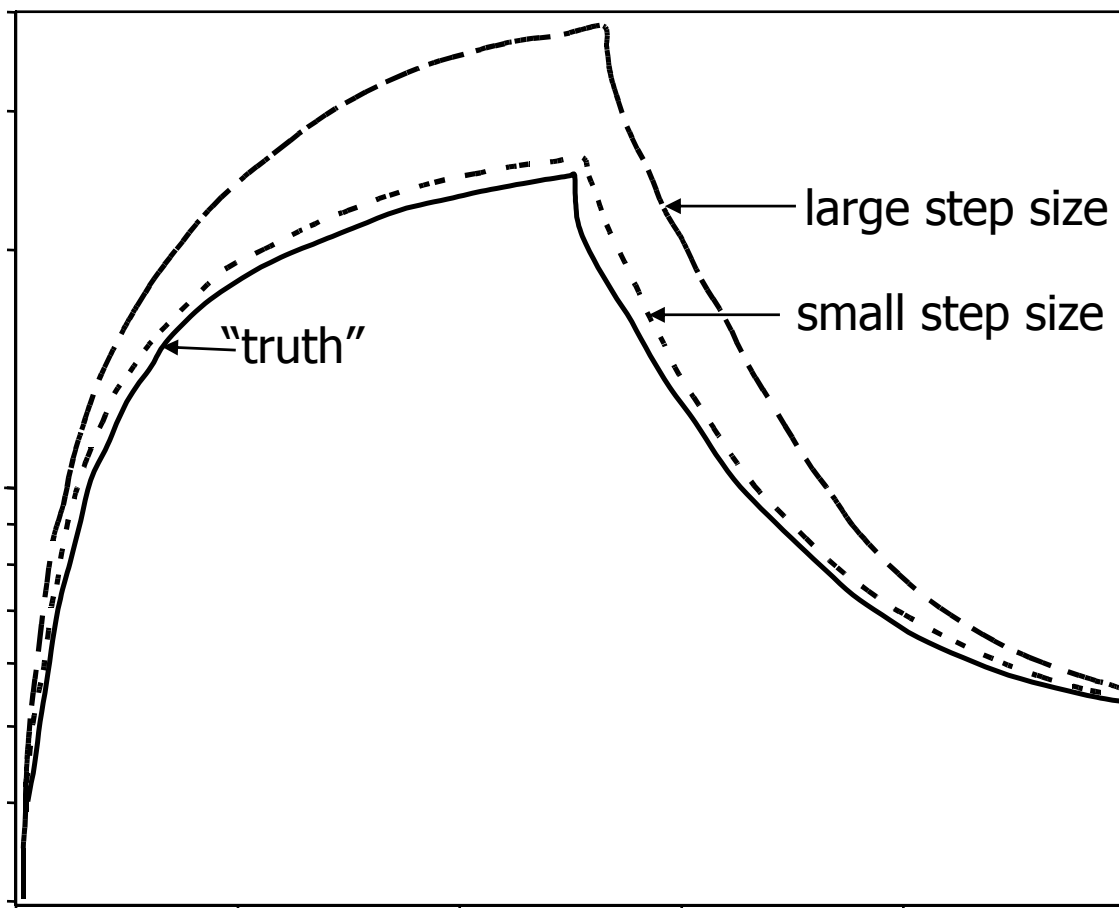
The approach generally used to solve ODE initial value problems is referred to as “numerical integration”.

Numerical Integration Algorithms: “Inching along in the Dark”



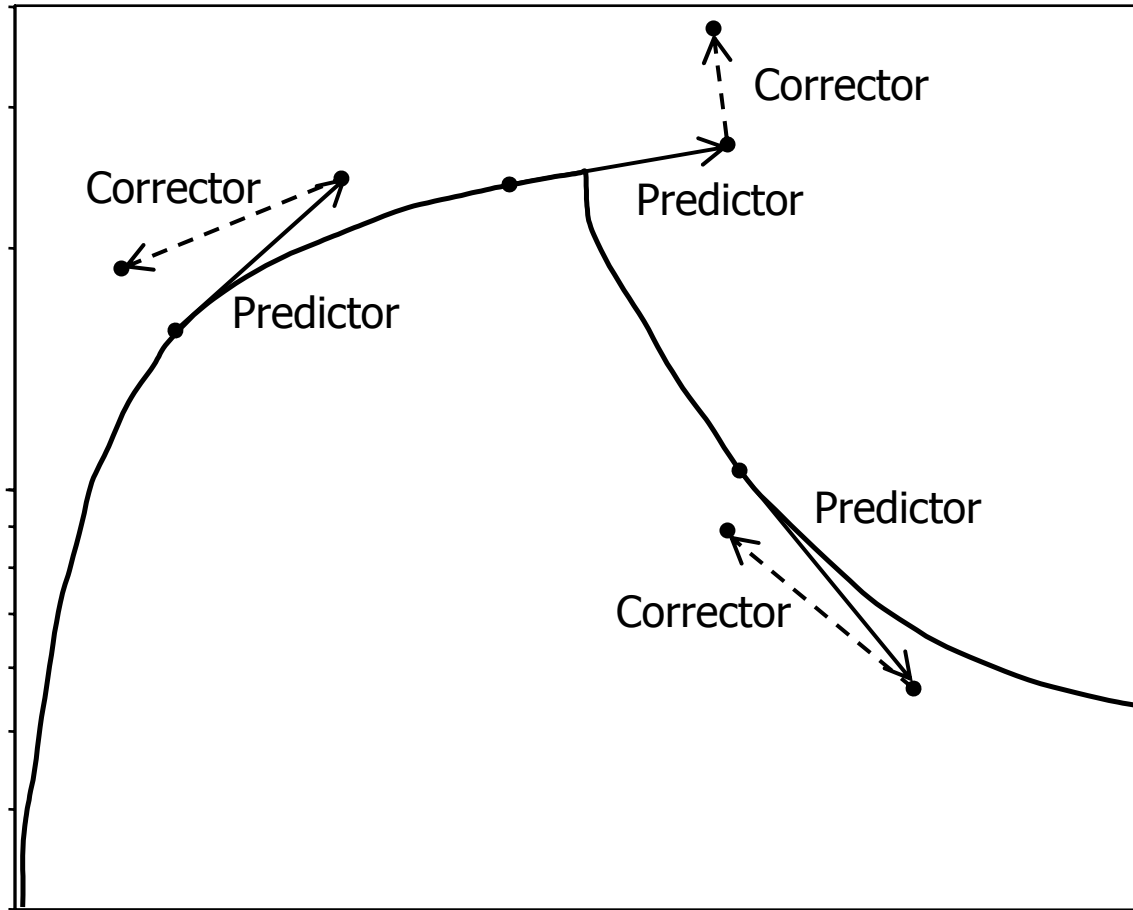
$$Y(t + \delta) = Y(t) + Y' \bullet \delta$$

Numerical Integration Algorithms: "Inching along in the Dark"



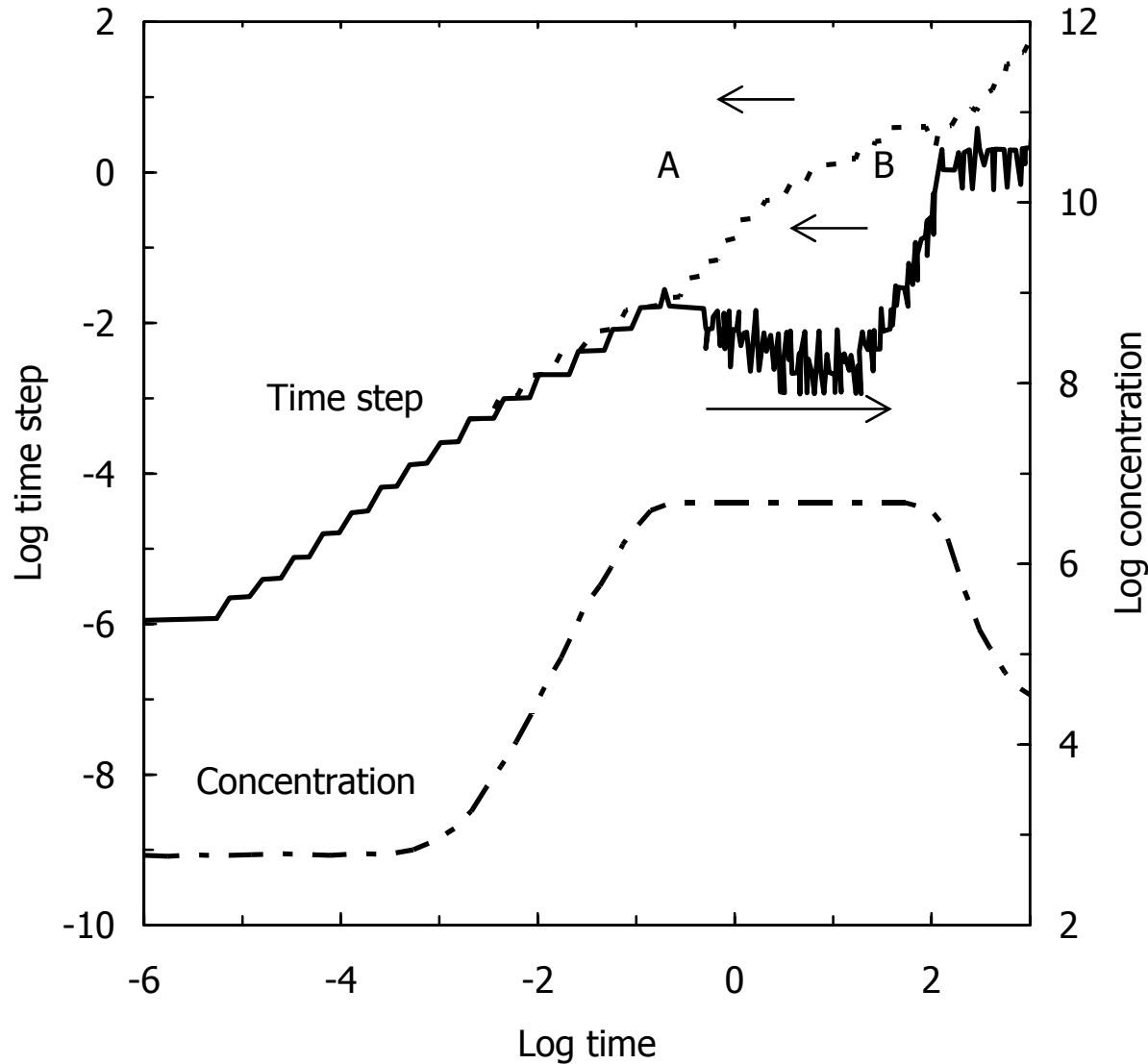
The larger the step size, the larger the potential for systematic error

Numerical Integration Algorithms: "Inching along in the Dark"



Automatic step size adjustment based on error tolerance

PBPK models with widely varying compartment time constants behave as stiff systems, fooling automatic step-size algorithms



Comparison of the performance of a stiff (A) and nonstiff (B) ODE solver for a problem exhibiting steady-state behavior (broken line). The time step of the nonstiff program is severely constrained during the steady-state period. The nonstiff program had about 75 times as many steps and took about 50 times as long to solve the problem.

Stiff ODE solvers:
Gear (LSODES)
Rosenbrock

Popular Simulation Languages for PBPK Modeling

acslXtreme: AEGIS Technologies Group, Inc.
<http://www.aegistg.com>

Berkeley Madonna: University of California at Berkeley
<http://www.berkeleymadonna.com>

MATLAB/Simulink: The MathWorks, Inc.
<http://www.mathworks.com>

ModelMaker: ModelKinetix
<http://www.modelkinetix.com>

LANGUAGE COMPARISON 1: BASIC FEATURES

“Feature”	ACSL	acslXtreme	MATLAB	MATLAB/ SIMULINK	Model Maker	SCoP	Berkeley Madonna
Graphical interface	N	Y	N	Y	Y	N	Y
Equation interface	Y	Y	Y	N	Y	Y	Y
Built-in integration algorithm	Y	Y	N	Y	Y	Y	Y
Discrete Blocks	Y	Y	N	Y	Y	N	N
Scripting	Y	Y	Y	Y	N	Y *	N +
Code sorting	Y	Y	N	n/a	n/a	N	Y

* Menu keystroke list

+ Under development

Example of Statement Order in Unsorted Code

```
function dy = sodes(tspan,y0)
%MATLAB styrene PBPK model ODE definition file

global c e %Make constants and variables externally available

%Transfer initial condition information to state variables
e.at = y0(1);
e.af = y0(2);
e.am = y0(3);
e.al = y0(4);
e.adose = y0(5);

%Styrene in tissue (rest of body) compartment
e.ct = e.at/e.vt; % (mg/l)
e.cvt = e.ct/c.pt; %tissue venous blood (mg/l)

%Styrene in fat compartment
e.cf = e.af/e.vf; % (mg/l)
e.cvf = e.cf/c.pf; %Fat venous blood (mg/l)
```


Calling the Integration Algorithm

```
while hours < c.tstop
    %Keep track of time
    hoursold = hours; hours = hours + e.cint;
    days = hours/24; hod = hours-floor(days)*24;    %hour of day
    weeks = days/7; dow = (hours - floor(weeks)*168.)/24;    %day of week 1 - 7

    %Turn exposure on or off
    if hod >= c.offset & hod <= c.offset + c.length & dow <= 5 %expose 5 days per week
        e.ci = c.conc*c.mw/24450.;    %Exposure concentration (mg/l)
    else e.ci = 0.; end

    %Numerical integration
    tspan = [hoursold hours];    %interval over which model is integrated
    y0 = [e.at; e.af; e.am; e.al; e.adose];    %initial conditions for the state variables
    [T,Y] = ode15s('sodes',tspan,y0);    %"sodes.m" contains the model's equations

    %Save data for plotting
    counter = counter + 1;
    data(1,counter) = hours; data(2,counter) = e.at; data(3,counter) = e.af;
    data(4,counter) = e.am; data(5,counter) = e.al; data(6,counter) = e.cv;
end
```

LANGUAGE COMPARISON 2: COST/SPEED

“Feature”	ACSL	acslXtreme	MATLAB	MATLAB/ SIMULINK	Model Maker	SCoP	Berkeley Madonna
Cost *	n/a	\$12,000	\$4,000	\$6,000	\$360	\$150	\$300
Compiled (faster)	Y	Y	Y +	N*	N	Y	Y
Interpreted (more convenient)	N	N	Y	Y*	Y	N	N

* Partially compiled

+ Extra cost

- Student versions
- Academic discounts

LANGUAGE COMPARISON 3: ADVANCED FEATURES

“Feature”	ACSL	acslXtreme	MATLAB	MATLAB/ SIMULINK	Model Maker	SCoP	Berkeley Madonna
Optimization	Y ⁺	Y	Y ⁺ *	Y ⁺ *	Y	Y	Y
Sensitivity analysis	Y*	Y	Y*	Y*	Y	Y	Y
Monte Carlo	Y*	Y*	Y*	Y*	Y	N	Y*
Units checking	N	N	N	N	N	Y	N
Database of physiological values	N	N	N	N	N	N	N

* User supplied/developed

⁺ Extra cost

Summary: Simulation Languages for PBPK Modeling

ACSL / SCoP: popular in past
no longer supported / upgraded

acslXtreme: powerful, expensive
becoming more MATLAB-like

Berkeley Madonna: fast, fun, flexible
scripting under development

MATLAB/Simulink: powerful, labor intensive

ModelMaker: good Monte Carlo front-end
no scripting