CBE 450 Chemical Reactor Fundamentals
Fall, 2009
Sample Codes
David Keffer
Department of Chemical and Biomolecular Engineering
University of Tennessee
dkeffer@utk.edu

This document provides the sample input for a Newton-Raphson code in the function funkeval and the sample input for a Runge-Kutta code in the function sysodeinput to iteratively solve for a reactor volume of a CSTR that meets a specified conversion.

In essence, each iteration of the Newton-Raphson routine requires that we solve the system of ODEs three times. The first time provides the value of the exit concentration at the current guess of the system volume. Once we have the exit concentration of A, we can compute the conversion. The second two calls are used to calculate the numerical derivative of the function with respect to conversion.

The global statement is the simplest way to pass the changing value of the volume guess from the Newton-Raphson routine to the ODE input file.

The code would be invoked at the command line by starting the Newton-Raphson code,

[x0,err] = newraph_nd(500.0).

```
function f = funkeval(x)
global V
V = x;
CAin = 10.0; % mol/liter
conv_spec = 0.95;
m = 2;
n = 1000;
xo = 0;
xf = 100;
yo = [10,0];
[y,x] = sysode(m,n,xo,xf,yo);
CA_out = y(n+1,1);
conv_calc = 1 - CA_out/CAin;
f = conv_spec - conv_calc;

function dydt = sysodeinput(x,y,nvec);
% define global variables
global V
%
%  one reaction
%  2A --> B
%
CA = y(1);
CB = y(2);
```

```
nuA = -2.0;
nuB = 1.0;
% constant volume;
F = 1; % liter/sec
Fin = F;
Fout = F;
CAin = 10.0; % mol/liter
CBin = 0.0; % mol/liter
%V = 10; % liter
R = 8.314; % J/mol/K
T = 300; % K
ko = 1.0e-1; % liter/mol/sec
Ea = 2500; % J/mol
k = ko*exp(-Ea/(R*T));
rate = k*CA*CA;
%
%  molar balance on A - concentration
%
dydt(1) = Fin/V*CAin - Fout/V*CA + nuA*rate;
%
%  molar balance on B - concentration
%
dydt(2) = Fin/V*CBin - Fout/V*CB + nuB*rate;
```