Fortran 95: Finite Difference Solutions to Differential Equations

Shane Stevens

January 24, 2011

Contents

1 Chemical Kinetic Model of Consecutive Reactions **1**
1.1 Differential Equations for above: 1
1.2 Finite Difference Equations for Differential Equations: 1

2 Fortran Code **2**

3 R Version of Program **2**

List of Figures

1 Plot of R code generated above 3

1 Chemical Kinetic Model of Consecutive Reactions

$$A \xrightarrow{k_1} B \tag{1}$$
$$B \xrightarrow{k_2} C \tag{2}$$

1.1 Differential Equations for above:

$$\frac{d[A]}{dt} = -k_1[A] \tag{3}$$
$$\frac{d[B]}{dt} = k_1[A] - k_2[B] \tag{4}$$
$$\frac{d[C]}{dt} = k_2 B \tag{5}$$

1.2 Finite Difference Equations for Differential Equations:

$$[A(i)] = [A(i-1)] - k_1[A(i-1)]\Delta t \tag{6}$$
$$[B(i)] = [B(i-1)] + (k_1[A(i-1)] - k_2[B(i-1)])\Delta t \tag{7}$$
$$[C(i)] = [C(i-1)] + k_2[B(i-1)]\Delta t \tag{8}$$

2 Fortran Code

```
PROGRAM finite_differences
IMPLICIT NONE

DOUBLE PRECISION, DIMENSION (2000,8) :: data=0.0
DOUBLE PRECISION, :: k1=5.0, k2=20.0, dt=0.005
INTEGER :: i, j

DO i=2, 2000
  data(i,1) = data(i-1,1) + data(i-1,1)*(-k1)*dt
  data(i,2) = data(i-1,2) + (data(i-1,1)*k1*dt - data(i-1,2)*k2*dt)
  data(i,3) = data(i-1,3) + data(i-1,2)*k2*dt
  data(i,4) = data(i-1,1)*k1/k2
  data(i,5) = 500.0*DEXP(-k1*i*dt)
  data(i,6) = k1/(k2-k1)*5000.0*( DEXP(-k1*i*dt) - DEXP(-k2*i*dt) )
 data(i,7) = 500.0*( 1- DEXP(-k1*i*dt) - k1/(k2-k1) * (DEXP(-k1*i*dt)-DEXP(-k2*i*dt) ) )
  data(i,8) = i*dt
END DO

OPEN(UNIT=25, FILE='output.csv', ACTION='WRITE', STATUS='REPLACE')
WRITE(25,30) ( (data(i,j), j=1, 8), i=1, 2000)
30 FORMAT ( 7(ES20.12, ','), ES20.12)

END PROGRAM finite_differences
```

3 R Version of Program

```
# Set Variables:
k1 = 5
k2 = 20
dt = 5e-04
data = rep(0, 16000)
dim(data) = c(2000, 8)
data = data.frame(data)
data[1, 1] = 500

for (i in 2:2000) {
    data[i, 1] = data[i - 1, 1] - data[i - 1, 1] * k1 * dt
    data[i, 2] = data[i - 1, 2] + (data[i - 1, 1] * k1 * dt - data[i - 1, 2] *
        k2 * dt)
    data[i, 3] = data[i - 1, 3] + data[i - 1, 2] * k2 * dt
    data[i, 4] = data[i - 1, 1] * k1/k2
    data[i, 5] = 500 * exp(-k1 * i * dt)
    data[i, 6] = k1/(k2 - k1) * 500 * (exp(-k1 * i * dt) - exp(-k2 * i * dt))
    data[i, 7] = 500 * (1 - exp(-k1 * i * dt) - k1/(k2 - k1) * (exp(-k1 * i *
        dt) - exp(-k2 * i * dt)))
    data[i, 8] = i * dt
}
```

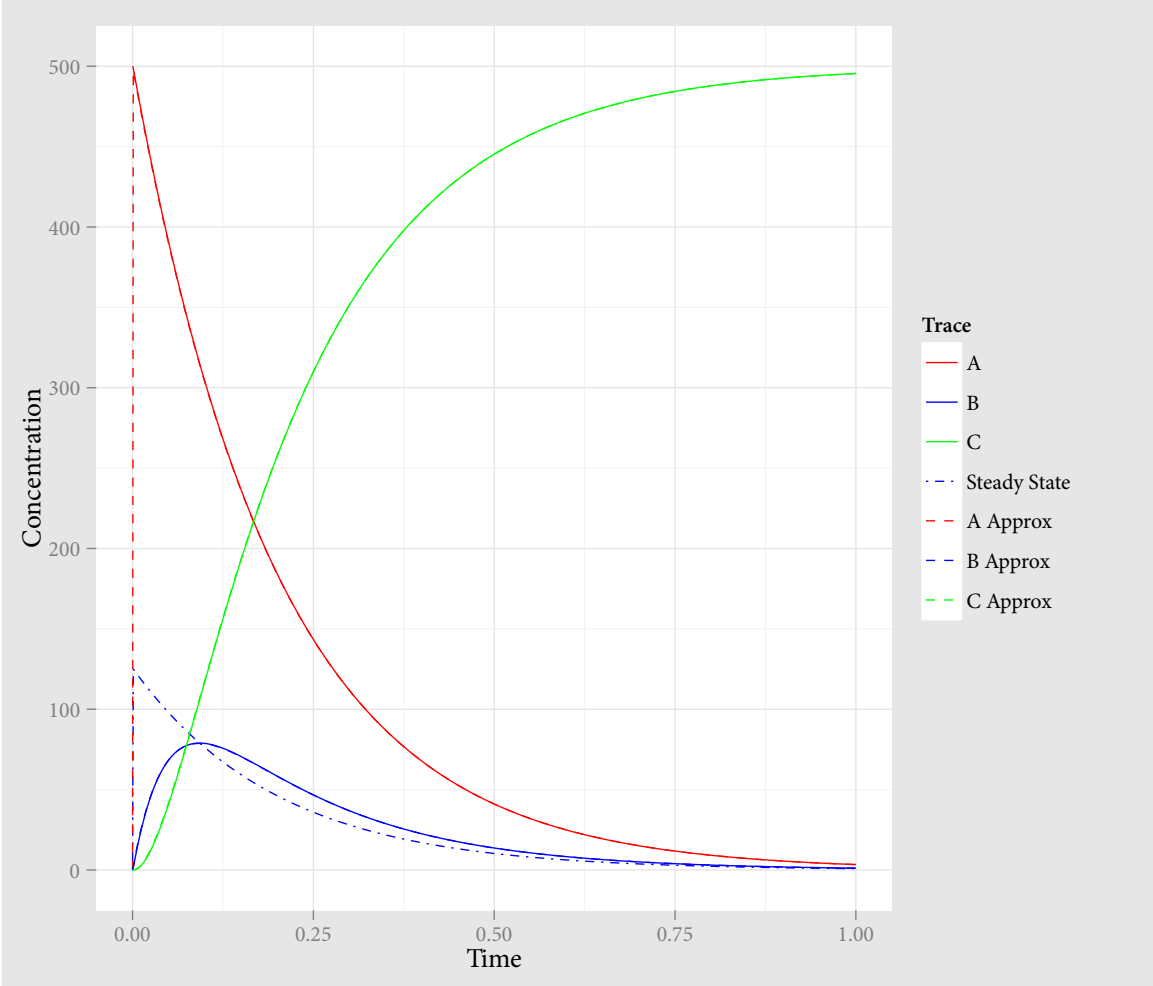

Figure 1: Plot of R code generated above

```
names(data)=c('A','B','C','Steady State','A Approx','B Approx','C Approx','Time')
data2=melt(data,id.vars='Time',
        measure.vars=c('A','B','C','Steady State','A Approx','B Approx','C Approx'))
names(data2)=c('Time','Trace','Concentration')
```

```
ggplot(data2,aes(Time,Concentration,color=Trace,alpha=Trace))+
  geom_line(aes(linetype=Trace))+
  scale_linetype_manual(values=c(rep('solid',3),'dotdash',rep('dashed',3)))+
  scale_color_manual(values=c('red','blue','green','blue2','red','blue','green'))+
  scale_alpha_manual(values=c(rep(1,3),rep(.4,4)))+theme_igray()
```