The Chemical Akzo Nobel Problem and VBA™

Edward M. Rosen
EMR Technology Group, Chesterfield, Mo 63017

Introduction

The Chemical Akzo Nobel problem is a stiff system of 6 non-linear DAEs (Differential Algebraic Equations) of index 1. A solution[1] is presented by the PSIDE-1 Solver[2] written in FORTRAN. It is of interest to compare that solution to a VBA™ solution which is the subject of this communication.

For this problem VBA is attractive for the following reasons:

1. Most industry (and academia) has Excel of which VBA is part. No additional software is required.
2. VBA files can be transferred to other VBA users .
3. For small problems (like this one) speed of execution is not an issue.
4. Spreadsheet output allows easy access to data for graphing.

Origin of the Problem

The problem originates from Akzo Nobel Central Research in Arnhem, The Netherlands. It describes a chemical process, in which 2 species, FLB and ZHU, are mixed, while carbon dioxide is continuously added. The resulting species of importance is ZLA. The names of the chemical species are fictitious.

Mathematical Description of the Problem

The reactions are given by:

2 FLB + ½ CO2 $\overset{k1}{\rightarrow}$ FLBT + H2O

ZLA + FLB $\overset{k2/K}{\Longleftrightarrow}$ FLBT + ZHU

FLB + 2 ZHU +CO_2 $\overset{k3}{\rightarrow}$ LB + nitrate

FLB.ZHU + ½ CO_2 $\overset{k4}{\rightarrow}$ ZLA + H_2O

FLB + ZHU $\Leftrightarrow$ FLB.ZHU

The rate equations for each reaction:

$r_1 = k_1 [FLB]^4 \ [CO_2]^{1/2}$

$r_2 = k_2 \ [FLBT] \ [ZHU]$

$r_3 = k_2/K \ [FLB] \ [ZLA]$

$r_4 = k_3[FLB] \ [ZHU]^2$

$r_5 = k_4 \ [FLB.ZHU]^2 \ [CO_2]^{1/2}$

$K_s = [FLB.ZHU]/([FLB][ZHU])$ (1)

The last equation describes an equilibrium.

The inflow of carbon dioxide per volume unit is denoted by F_{in} and satisfies

$$F_{in} = klA \ (p(CO_2)/H - [CO_2])$$

The rate of change of concentrations of each chemical species depends on the rate equations:

$FLB'[t] = -2r_1 + r_2 - r_3 - r_4$

$CO_2'[t] = -0.5 \ r_1 - r_4 - 0.5 \ r_5 + F_{in}$

$FLBT'[t] = r_1 - r_2 + r_3$

$ZHU'[t] = -r_2 + r_3 - 2r_4$

$ZLA'[t] = r_2 - r_3 + r_5$ (2)

Reaction parameters for each of the reactions are given Table 1:

Initial values are: FLB [0] = 0.444, $CO_2[0] = 0.00123$, FLBT[0] = 0, ZHU[0] =0.007, ZLA[0] = 0

Parameters for Chemical Akzo Nobel Problem	
Symbol	Value
k_1	18.7
k_2	0.58
k_3	0.09
k_4	0.42
K	34.4
KlA	3.3
K_s	115.83
$p(CO_2)$	0.9
H	737

Table 1. Parameters for the Chemical Akzo Nobel Problem

Numerical Solution of the Problem

The integration of equations (2) is carried out with a fourth order Runge-Kutta routine implemented in VBA™ function Integ (Appendix I). Five equations are integrated as the product K_s FLB[t] ZHU[t] is substituted for FLT.ZHU[t] in equation (1) making the system index 0.

Table 2. consists of the first 10 rows of the spreadsheet developed in this study. A stepsize of 0.01 min is chosen for the integration step size (many larger step sizes failed). The first row (row 7) shows the starting values for FLB, CO_2 and ZHU. The value of FLB.ZHU[t] is calculated from the product K_s FLB[t] ZHU[t].

The first 6 columns of row 8 are highlighted (for the output) and the following is entered:
=Integ($A7, $B7:$F7, I6) with **Ctrl+Shft+Enter**.

The integration is carried out (in function integ) and the integrated time ($A8) and values for FLB to ZLA ($B8:$F8) are inserted into the highlighted output area. The I6 specifies the first element of the parameters (Table 1) which start (as a column) at location I6.

After row 8 is completed the next entries up to row 18007 are copied from row 8.

Reaction System										
A	B	C	D	E	F	G	H	I	J	K
Time	FLB	CO_2	FLBT	ZHU	ZLA	FLB.ZHU		prm		
min								5.00	N	1
0.000	0.4444	0.00123	0.0000	0.0070	0.0000	0.3603		0.01	h	2
0.010	0.4439	0.00110	0.0002	0.0070	0.0000	0.3599		18.70	k1	3
0.020	0.4434	0.00098	0.0005	0.0070	0.0000	0.3595		0.58	k2	4
0.030	0.4430	0.00087	0.0007	0.0070	0.0001	0.3592		0.09	k3	5
0.040	0.4426	0.00077	0.0009	0.0070	0.0001	0.3588		0.42	k4	6
0.050	0.4422	0.00069	0.0011	0.0070	0.0001	0.3585		34.40	K	7
0.060	0.4418	0.00061	0.0013	0.0070	0.0001	0.3582		3.30	klA	8
0.070	0.4415	0.00054	0.0015	0.0070	0.0001	0.3579		115.83	Ks	9
0.080	0.4412	0.00047	0.0016	0.0070	0.0001	0.3577		0.90	pCO_2	10
0.090	0.4409	0.00042	0.0018	0.0070	0.0001	0.3574		737.00	H	11

Table 2. First 10 rows of the spreadsheet

The value of CO_2 drops very rapidly before it rises steeply. Figure 1 is a plot of the concentration values of CO_2 during the first 0.3 minutes.

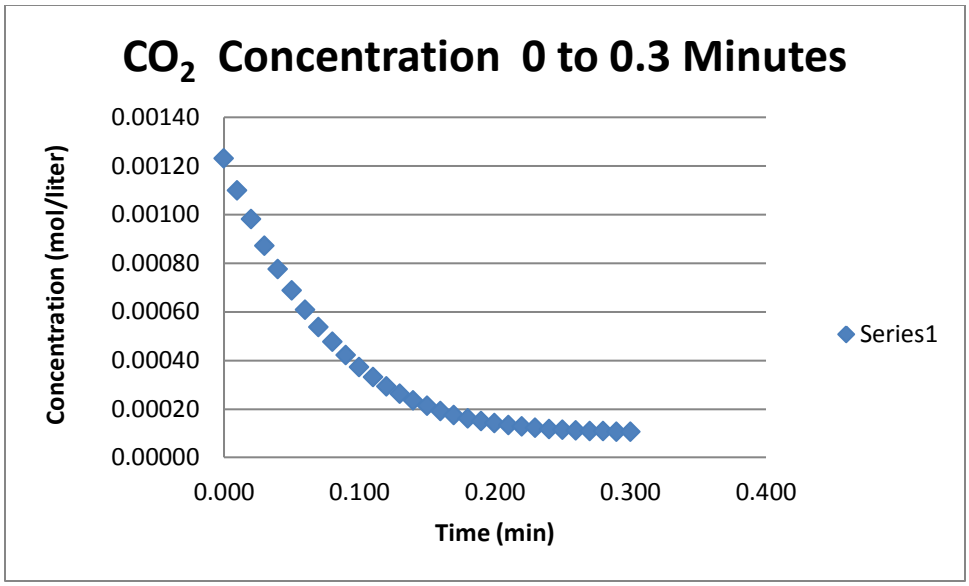

Figure 1. Concentration values of CO_2 0 to 0.3 Minutes

Since it is desired to determine the solution out to 180 minutes, some 18000 time increments are need (stepsize 0.01 min).

.

Graphs of the solution

Since the number of spreadsheet increments are more than can be plotted, just every 144 points were selected for graphing (Appendix II). Function TRY (Appendix II) places the selected points in columns AA to AG (not shown).

Plots of the 126 selected points for each component are shown in Figures 2 to 7.

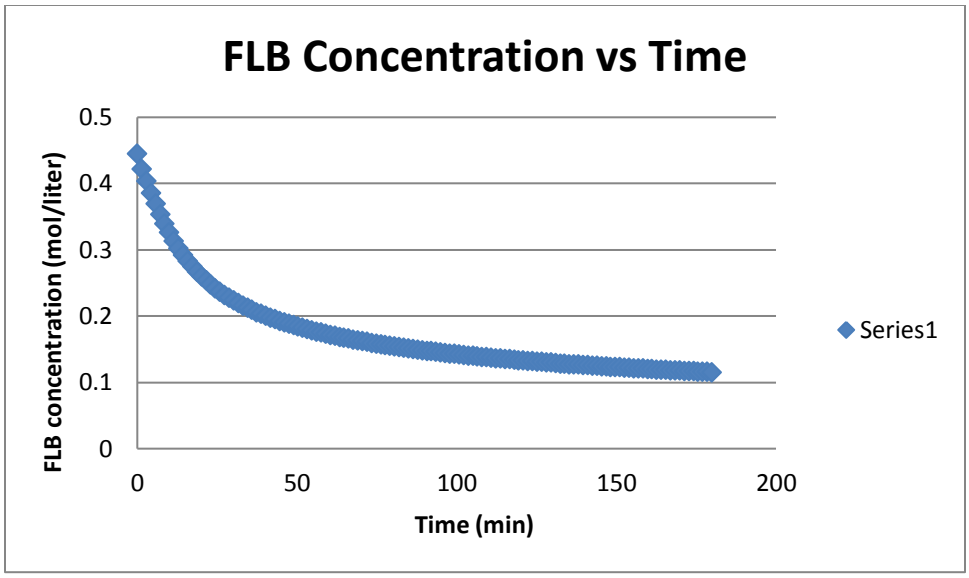

Figure 2. FLB Concentration vs Time

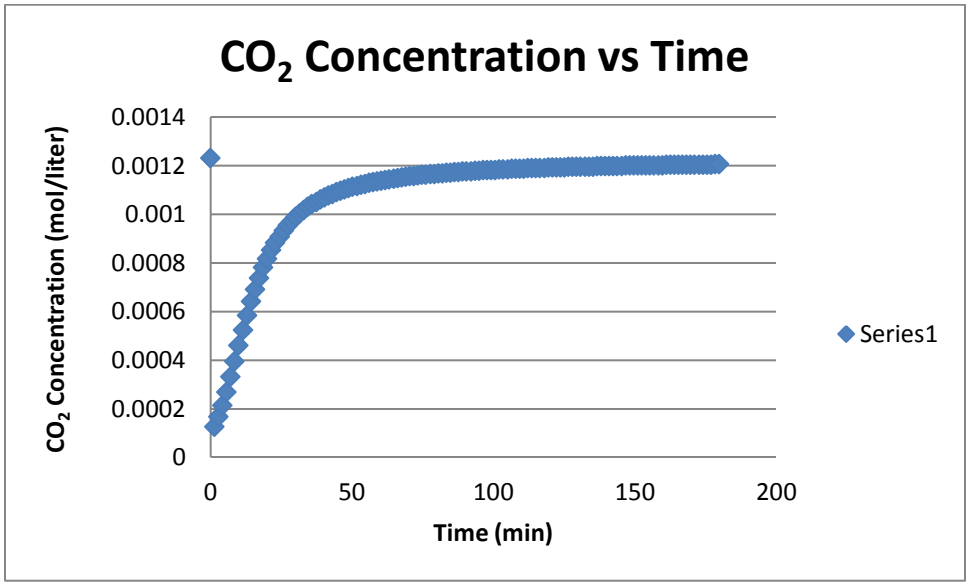

Figure 3. CO_2 Concentration vs Time

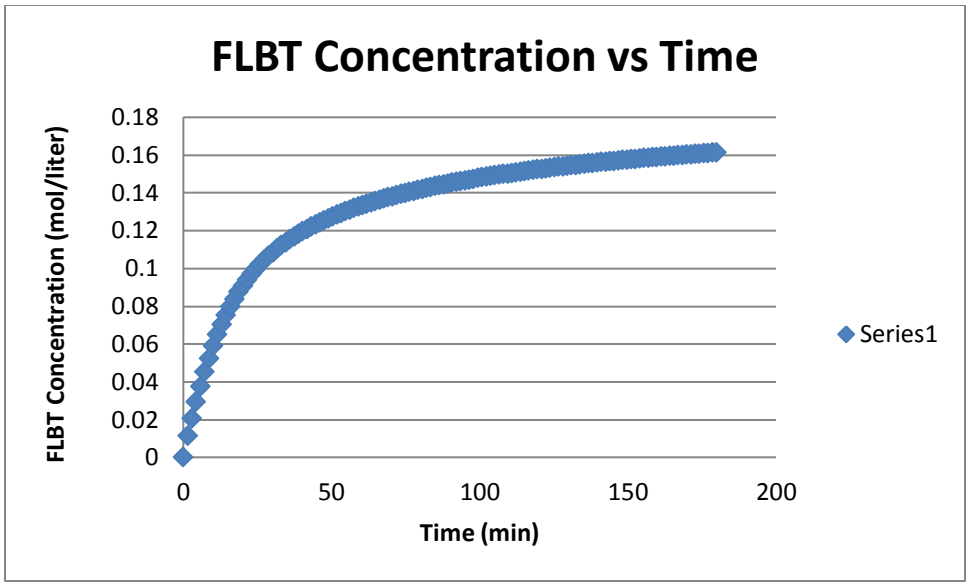

Figure 4. FLBT Concentration vs Time

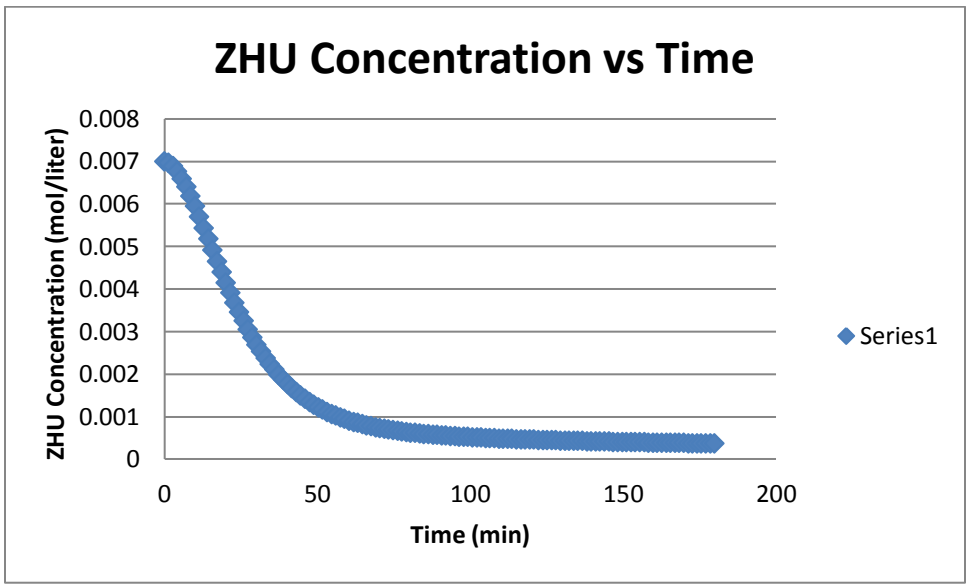

Figure 5. ZHU Concentration vs Time

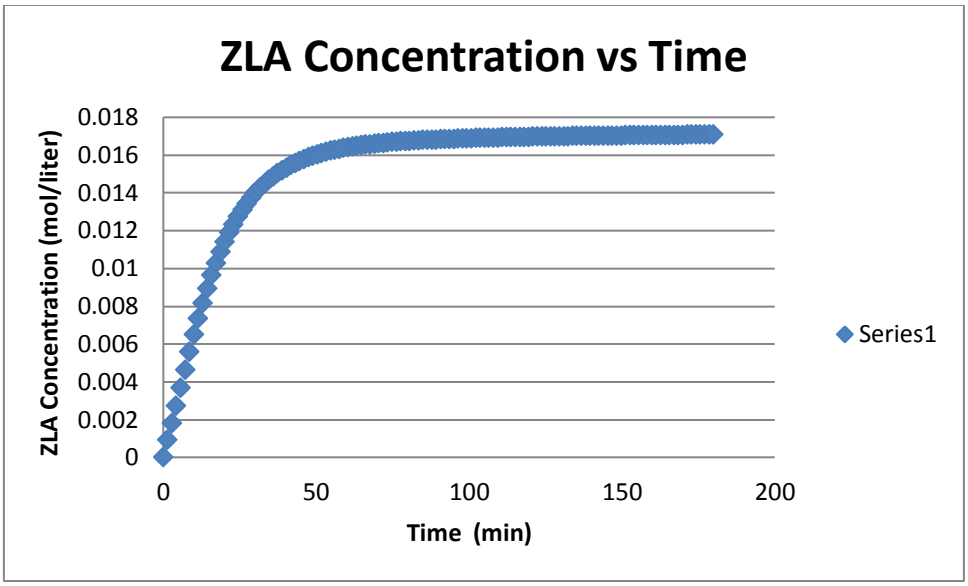

Figure 6. ZLA Concentration vs Time

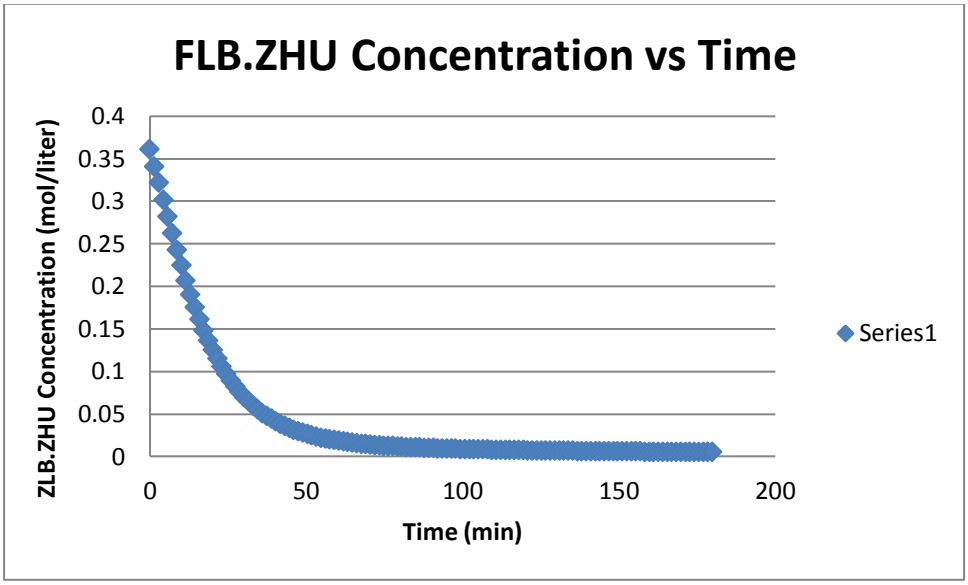

Figure 7. FLB.ZHU Concentration vs Time

Conclusions

Table 3. compares the results of the (PSIDE-1) FORTRAN and VBA™ integrations after 180 minutes. There are only minor differences.

A possible downside of the VBA™ solution is the large size of the spreadsheet (18,007 rows).

The Chemical Akzo Nobel Problem is one of a number of problems designed to test various IVP (Initial Value Problem) codes. The test problems and solutions are detailed in [3].

Concentrations After 180 min

Component	PSIDE-1*	VBA™
FLB	1.15079E-01	1.15084E-01
CO_2	1.20383E-03	1.20383E-03
FLBT	1.61156E-01	1.61354E-01
ZHU	3.65615E-04	3.65004E-04
ZLA	1.70801E-02	1.70725E-02
FLB.ZHU	4.87353E-03	4.86558E-03

*Reference [1] gives concentrations to 16 significant figures

Table 3. Comparison of FORTRAN and VBA™ Solutions After 180 Minutes

References

1. http://www.dm.uniba.it/~testset/report/chemakzo.pdf
2. http://www.dm.uniba.it/~testset/solvers/pside.php
3. http://www.dm.uniba.it/~testset

Appendix I

```
Option Explicit

Private Function integ(x, y, prm)

Dim N, IR, NN, I As Integer
```

```
Dim h, xx As Double
N = prm(1)
NN = N + 1
ReDim yy(1 To N) As Double
ReDim ddd(1 To NN)
h = prm(2)
xx = x
For I = 1 To N
  yy(I) = y(I)
Next
IR = rk4a(N, h, xx, yy, prm)
xx = xx + h
ddd(1) = xx
 For I = 2 To NN
  ddd(I) = yy(I - 1)
Next I
integ = ddd
End Function


Public Function rk4a(N, h, x, y, prm)
'Modified from Pedro L. Claveria abril/2002
'based in EMR Technology Group Library
'n = number of equations
'h  = step size for integration
```

```
'x = independent variable

'y = vector of dependent variables

'prm = vector parameters

ReDim ccc(N), fff(N)

ReDim k1(N), k2(N), k3(N), k4(N)

ReDim y2(N), y3(N), y4(N)


Dim muda1, muda2, muda3, muda4 As Double

Dim I As Integer

'Calculation of k1

muda1 = dydx(x, y, prm, fff)

For I = 1 To N: k1(I) = fff(I): Next

'Calculation of k2

For I = 1 To N: y2(I) = y(I) + 0.5 * h * k1(I): Next

muda2 = dydx(x + h / 2, y2, prm, fff)

For I = 1 To N: k2(I) = fff(I): Next

'Calculation of k3

For I = 1 To N: y3(I) = y(I) + 0.5 * h * k2(I): Next

muda3 = dydx(x + h / 2, y3, prm, fff)

For I = 1 To N: k3(I) = fff(I): Next

'Calculation of k4

For I = 1 To N: y4(I) = y(I) + h * k3(I): Next

muda4 = dydx(x + h, y4, prm, fff)

For I = 1 To N: k4(I) = fff(I): Next

'New values of the dependent variables
```

```
For I = 1 To N
   ccc(I) = y(I) + (h / 6) * (k1(I) + 2 * k2(I) + 2 * k3(I) + k4(I))
Next I
For I = 1 To N
   y(I) = ccc(I)
Next I

rk4a = 0
End Function

Private Function dydx(xx, yy, prm, fff)
Dim r1, r2, r3, r4, r5, Fin, xxx As Double
'prm (1) = N
'prm (2) = h
'prm (3) = k1
'prm (4) = k2
'prm (5) = k3
'prm (6) = k4
'prm (7) = K
'prm (8) = k1A
```

```
'prm (9) = Ks

'prm (10) = pCO2

'prm (11) = H

' yy(1) = FLB

' yy(2) = CO2

' yy(3) = FLBT

' yy(4) = ZHU

' yy(5) = ZLA

Fin = prm(8) * ((prm(10) / prm(11)) - yy(2))

r1 = prm(3) * (yy(1) ^ 4) * (yy(2) ^ 0.5)

r2 = prm(4) * yy(3) * yy(4)

r3 = (prm(4) / prm(7)) * yy(1) * yy(5)

r4 = prm(5) * yy(1) * (yy(4) ^ 2)

xxx = prm(9) * yy(1) * yy(4)

r5 = prm(6) * (xxx ^ 2) * (yy(2) ^ 0.5)

fff(1) = -2 * r1 + r2 - r3 - r4

fff(2) = -0.5 * r1 - r4 - 0.5 * r5 + Fin

fff(3) = r1 - r2 + r3

fff(4) = -r2 + r3 - 2 * r4

fff(5) = r2 - r3 + r5

dydx = 0

End Function
```

Appendix II

```
Private Function TRY()

'Generation of Selected Plot Points

Dim GG(7), J, K, L, JJ As Integer

JJ = 1

For J = 7 To 18007 Step 144

   For K = 1 To 7

    GG(K) = Cells(J, K)

   Next K

    For L = 1 To 7

    Cells(JJ, 26 + L) = GG(L)

   Next L

    JJ = JJ + 1

  Next J

End Function
```