
CACHE News

No. 49

Fall, 1999

On the Use of IMSL Routines in Excel 7.0 <i>Edward M. Rosen, EMR Technology Group</i>	1
Engineering Computing with Maple: Solution of PDEs via the Method of Lines <i>Ross Taylor, Clarkson University</i>	5
Using Computational Fluid Dynamics as a Teaching Aid for Chemical Engineering <i>Liz Marshall, Fluent Inc.</i>	9
New CACHE Products	
Process Design Case Study Volume 7 Design of an Ethylbenzene Production Plant Preliminary <i>Ignacio E. Grossmann, Carnegie Mellon University</i>	12
Visual Encyclopedia of Chemical Engineering Equipment for Macintosh and Windows 95/NT <i>Susan Montgomery, University of Michigan</i>	13
Material and Energy Balances Multimedia Modules for Macintosh and Windows 95/NT <i>Susan Montgomery, University of Michigan</i>	15
Conferences	
7th International Symposium on Process Systems Engineering (PSE) <i>July 16-21, 2000, Keystone, Colorado</i>	18
Foundations of Molecular Modeling and Simulation (FOMMS): Applications for Industry <i>July 23-28, 2000, Keystone, Colorado</i>	20
Standard Order Form	22
Supporting Departments	24

On the Use of IMSL Routines in Excel 7.0

Edward M. Rosen, EMR Technology Group

The IMSL™ library is a collection of nearly 1000 mathematical and statistical subroutines written in FORTRAN. The Library is supplied as part of Digital Visual FORTRAN Professional Edition (1). As such it forms an attractive source of routines that can be of use in carrying out a variety of chemical engineering calculations using Excel 7.0.

The basic mechanics of calling FORTRAN subroutines from Excel 7.0 has been discussed previously (2). However, use of the IMSL™ Library raises some additional coding issues that are indicated in the example below. It is assumed that a compiler is available for the generation of the DLL's (dynamic link libraries) such as Digital Visual FORTRAN (3).

An Example

The Van der Pol equation (1)

$$u'' + \mu (u^2 - 1) u' + u = 0, \mu > 0$$

is a single ordinary differential equation with a periodic limit cycle (4). For the value of $\mu = 5$, the equations are integrated from $t = 0$ until the limit has clearly developed at $t = 26$. The (arbitrary) initial conditions used here are $u(0) = 2$ and $u'(0) = -2/3$. The equation is solved as a differential-algebraic system by defining the first order system:

$$\begin{aligned} e &= 1/\mu \\ y_1 &= u \\ g_1 &= y_2 - y_1' = 0 \\ g_2 &= (1 - y_1^2) y_2 - e(y_1 + y_2') = 0 \end{aligned}$$

Note that the initial condition for y_2' ($=0$) is not consistent (g_2 is not equal to 0 at $t = 0$). The IMSL™ routine DASPG solves for this starting value using user written subroutine GCN which is called by DASPG.

Implementation in Excel 7.0

IMSL™ subroutine DASPG solves a first order differential-algebraic system of equations, $g(t, y, y') = 0$ using the Petzold-Gear BDF method. It is called by

```
CALL DASPG (N, T, TOUT, IDO, Y, YPR, GCN)
```

where

N - Number of differential equations (Input)
T - Independent variable, t (Input/Output)

TOUT - Final value of the independent variable (Input)
IDO - Flag indicating the state of the computation (Input/Output)
IDO = 1 Must be set on first entry T = t
IDO = 2 Set by DASPG after initial call
IDO = 3 Must be set on last entry to release workspace
Y - Array of size N containing the dependent variable values, y. This array must contain initial values. (Input/Output)
YPR - Array of size N containing derivative values, y'. This array must contain initial values. The routine will solve for consistent values of y' to satisfy the equations at the starting point.
GCN - User supplied subroutine to evaluate $g(t, y, y')$. The call is

```
CALL GCN(N, T, Y, YPR, GVAL)
```

where GCN must be declared EXTERNAL in the calling program. The routine will solve for values of $y'(t_0)$ so that $g(t_0, y, y') = 0$.

GVAL - Array of size N containing the function values $g(t, y, y')$, (Output)

Figure 1 is a listing of subroutine DAE which calls IMSL™ routine DASPG. The routine has two attribute statements (following the subroutine definition) beginning with !DEC\$. These are needed to communicate with the VBA (Visual Basic for Applications) macro which is part of the spreadsheet.

Subroutine DAE carries out one step of the integration. It is compiled into a DLL in the DOS window with the command

```
DF /DLL:DAE DAE.FOR
```

Figure 2 is a listing of the array function DAEV which is written in VBA. This function calls the DAE DLL whose location is specified in the Declare statement at the top of the function. N is specified as LONG to be consistent with FORTRAN.

The spreadsheet is shown in Figure 3. Note that N, Mu and the time increment are all parameters which are passed on to the array function DAEV and then to the subroutine DAE. The DAEV function is invoked on the spreadsheet (for the first time increment) with:

= DAEV(\$B\$4, \$C10,\$D10,\$E10,\$F10,\$G10)

where the area C11 to G11 is first selected and then **Ctrl+Shift+Enter** is entered. Subsequent rows (time increments) are then copied from this row.

Figure 3 gives the results of the computation. A number of the rows have been deleted for clarity.

Table 1 shows the final results obtained and is compared to that given in the example supplied with Digital Visual FORTRAN. In this work initialization (which calculates y' from $g(t, y, y')$) took place each step. In Reference (1) initialization took place only in the first step.

Table 1 Results of the Integration

	This Work	Reference(1)
Time	26.0	26.0
$y_1 = u$	1.46062	1.45330
$y_2 = u'$	-0.241912	-0.24486
$y_1' = u'$	-0.242119	-0.24713
$y_2' = u''$	-0.089932	-0.09399
g_1	0.000207	0.00227
g_2	4.81E-05	0.000442

Figure 4 is a plot of y_1 vs time. Reference (4) shows similar plots.

Conclusions

IMSL™ subroutines may be effectively called from Excel 7.0 but their use generally requires some coding in both FORTRAN and VBA. The effort, however, seems well worthwhile and gives new life to this valuable legacy code.

References

1. Digital Visual FORTRAN Professional Edition 5.0.A.
2. Rosen, E. M., "Calling FORTRAN Subroutines from Excel 7.0" *CACHE News*, No. 48, Spring, 1999
3. Web site: <http://www.compaq.com/fortran>
4. Davis, H. T. *Introduction to Nonlinear Differential and Integral Equations*, U.S. Atomic Energy Commission, Washington, D. C. (1960)

```
SUBROUTINE DAE (N, XU, T, TOUT, Y, YPR)
!DEC$ ATTRIBUTES DLLEXPORT::DAE
!DEC$ ATTRIBUTES ALIAS:' DAE' ::DAE
USE NUMERICAL_LIBRARIES
EXTERNAL GCN
INTEGER N, IDO
REAL Y(N), YPR(N), T, TOUT, ZU
COMMON /TRANS/ZU
ZU = XU
IDO = 1
10 CALL DASPG (N, T, TOUT, IDO, Y, YPR, GCN)
IF (IDO .EQ. 3) GO TO 99
IF (IDO .EQ. 2) THEN
    IDO = 3
    GO TO 10
END IF
99 CONTINUE
RETURN
END
SUBROUTINE GCN (N, T, Y, YPR, GVAL)
C SPECIFICATIONS FOR ARGUMENTS

INTEGER N
REAL T, Y(N), YPR(N), GVAL(N), XU
C SPECIFICATIONS FOR LOCAL VARIABLES
REAL EPS
COMMON /TRANS/ZU
C
EPS = 1./ZU

GVAL(1) = Y(2) - YPR(1)
GVAL(2) = (1.0 - Y(1)**2)*Y(2) - EPS*(Y(1) + YPR(2))
RETURN
END
```

Figure 1 Listing of FORTAN Subroutine DAE

```

Declare Sub DAE Lib"D:\DIGITA~3\DLLS\DAE.DLL" _
    (ByRef N As Long, ByRef Mu As Single, ByRef T As Single, _
    ByRef TOUT As Single, ByRef Y As Single, ByRef YPR As Single)

Public Function DAEV(PRM, TT, Y1, Y2, YP1, YP2)

    \ PRM(1) = Number of Equations
    \ PRM(2) = Value of Mu
    \ PRM(3) = Time Increment
    \ TT = the current time
    \ Y1 = the current value of y(1)
    \ Y2 = the current value of y(2)
    \ YP1 = y' (1)
    \ YP2 = y' (2)

    Dim T As Single
    Dim TOUT As Single
    Dim N As Long
    Dim Mu As Single
    Dim Delt As Single

    N = PRM(1)
    Mu = PRM(2)
    Delt = PRM(3)

    ReDim Y(1 To N) As Single
    ReDim YPR(1 To N) As Single
    ReDim EV(1 To 5) As Single

    T = TT
    TOUT = T + Delt

    Y(1) = Y1
    Y(2) = Y2
    YPR(1) = YP1
    YPR(2) = YP2

    Call DAE(N, Mu, T, TOUT, Y(1), YPR(1))
    EV(1) = T
    EV(2) = Y(1)
    EV(3) = Y(2)
    EV(4) = YPR(1)
    EV(5) = YPR(2)

    DAEV = EV

End Function

```

Figure 2 VBA Array Function DAEV

Solution of the Van der Pol Equation Using IMSL Routine DASPG

Solved as a differential-algebraic system by defining a first order system

Number of Equations 2
 Value of Mu 5
 Step Size 0.1

Increment	Time	Y(1)	Y(2)	YPR(1)	YPR(2)
0	0	2	-0.666667	-0.666667	0
1	0.1	1.958621	-0.260536	-0.260224	1.730957
2	0.2	1.938178	-0.16948	-0.169267	0.394544
3	0.3	1.922501	-0.14883	-0.148938	0.085246
4	0.4	1.907866	-0.145198	-0.145205	0.008979
5	0.5	1.893321	-0.145719	-0.146074	-0.01274
...
101	10.1	-0.865072	0.889704	0.89167	1.995529
102	10.2	-0.764277	1.141023	1.140842	3.136632
103	10.3	-0.631139	1.552824	1.557498	5.332912
104	10.4	-0.443047	2.274297	2.275965	9.593467
105	10.5	-0.15501	3.607771	3.608247	17.75351
106	10.6	0.31255	5.878559	5.877309	26.14278
107	10.7	1.011322	7.632876	7.630804	-1.864551
108	10.8	1.668629	4.766888	4.774143	-44.20763
...
256	25.6	1.551515	-0.21188	-0.21202	-0.060861
257	25.7	1.529896	-0.218337	-0.21849	-0.066558
258	25.8	1.507588	-0.225421	-0.22559	-0.073157
259	25.9	1.484524	-0.233237	-0.233423	-0.08086
260	26	1.46062	-0.241912	-0.242119	-0.089932

Figure 3 Spreadsheet Output (Abbreviated)

Van der Pol Equation

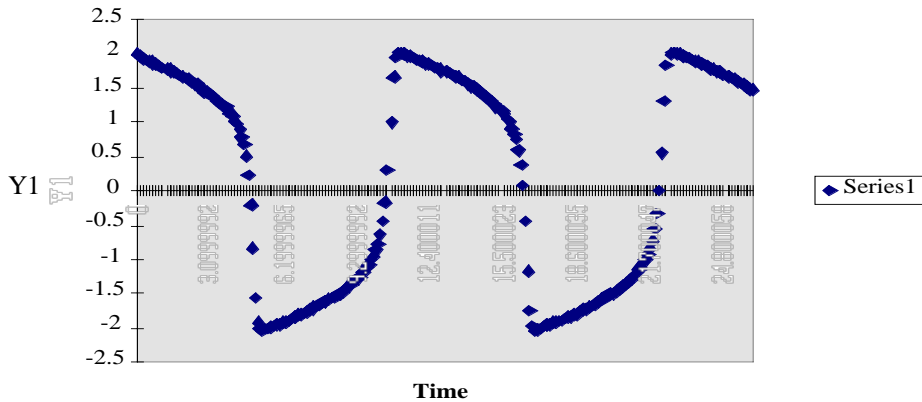


Figure 4. Plot of y1 vs Time

Engineering Computing with Maple: Solution of PDEs via the Method of Lines

Ross Taylor, Clarkson University

Introduction

The last two issues of CACHE News have included articles demonstrating how partial differential equations can be solved. Cutlip and Shacham (1998) used their own Polymath package to carry out the necessary computations using the Method of Lines (MOL), while Rosen (1999) has shown how the Excel spreadsheet can be used for other finite difference computations. Our purpose here is to show how easy it can be solve such problems using the computer algebra system Maple V.

The Method of Lines (MOL) is a versatile approach to obtaining numerical solutions to partial differential equations. The method has been made popular in science and engineering largely through the work of W.E. Schiesser and coworkers (see, for example, Schiesser, 1991; Silebi and Schiesser, 1992).

MOL can be used to deal with a wide variety of different types of PDE systems involving various types of PDE and/or boundary conditions. The fundamental idea behind MOL is that the partial derivatives in any spatial direction are replaced by finite difference approximations, leaving only the time derivatives (or first derivatives in one spatial direction if the equation is independent of time). Application of MOL to a PDE may lead to a system of purely ordinary differential equations (ODEs). Thus, all that may be needed to solve these equations is a good numerical ODE solver. ODE solvers are widely available including in the computer algebra system Maple V. Quite often, however, application of MOL leads to a mixed system of differential and algebraic equations or DAEs. While DAE systems sometimes can be solved by converting them to ODE systems, or by tricking an ODE solver into thinking that they are ODEs, it makes sense to solve the DAE system using a DAE solver. The availability of BESIRK, a DAE solver for Maple (Schwalbe et al., 1996), makes implementation of MOL in Maple very easy indeed.

In what follows we demonstrate the use of Maple by considering the same example used to illustrate the article of Cutlip and Shacham (1998). Actual Maple code (and results) are provided for a problem in which we find the temperature profiles in a slab as a function of space and time.

Example

Heat transfer in a slab is described by the following PDE:

```
> restart;  
> PDE := diff(T(x,t), t) =  
> alpha*diff(T(x,t), x,x) : PDE;
```

$$\frac{\partial}{\partial t} T(x, t) = \alpha \left(\frac{\partial^2}{\partial x^2} T(x, t) \right)$$

T is the temperature and α is the thermal diffusion coefficient. Initially, the slab is at a uniform temperature (T_0).

```
> IC := t=0, T(x,t)=T[0] : IC;
```

$$t = 0, T(x, t) = T_0$$

At $t = 0$ the temperature at one side is changed (to a value not yet specified, we shall call it (T_A))

```
> BC1 := x=0, T(x,t)=T[A] : BC1;
```

$$x = 0, T(x, t) = T_A$$

The other side is insulated and the temperature gradient is zero.

```
> BC2 := x=L, diff(T(x,t), x)=0 : BC2;
```

$$x = L, \frac{\partial}{\partial x} T(x, t) = 0$$

We are going to use the method of lines to solve this problem. The PDE applies to a sequence of values of T which are identified by subscript i . The derivatives of T with respect to x are to be replaced by second order finite difference approximations. The conversion to appropriate form is accomplished with the `'convert/fddiff'` command that is part of this authors `fdpack` package and which we now load into Maple (along with some other utilities).

```
> read 'e:/maple/numerics/pde/fdpack.mpl':  
> read 'e:/maple/utils/utils.mpl':  
> read 'e:/maple/thermo/plots/tplot.mpl':
```

`'convert/fddiff'` can derive finite difference approximations of any order, and with any (appropriate) number of forward and backward steps. In this particular case the temperature at each node is to remain a function of temperature only, and we only replace the spatial derivatives with a second order difference approximation. This is accomplished by including a zero in the order argument and none in the indexletters argument to `'convert/fddiff'` (in the second position here since t appears in the second position in the arguments to temperature in the original PDE).

```
> PDE1 := convert(PDE, fdiff,
> order=[2,0], forward=[1,0],
> indexletters=[i,none]): PDE1;
```

$$\frac{\partial}{\partial t} T_i = \frac{\alpha(T_{i+1} - 2T_i + T_{i-1}))}{h_x^2}$$

The above equation holds for all interior points, $1 < i < n$, where n is the number of grid points. It is not valid for the boundary lines ($i = 1, i = n$) since that would require us to include fictitious points ($i = 0, i = n+1$). The next step is to specify n .

```
> n:=11;
```

```
n:=11
```

The line spacing is

```
> hspec := h[x]=L/(n-1): hspec;
```

$$h_x = \frac{1}{10} L$$

The first boundary condition may be expressed as

```
> BC1a := convert(BC1[2], fdiff,
> order=[2,0], forward=[0,0],
> indexletters=[1,none]): BC1a;
```

$$T_1 = T_A$$

The insulated surface boundary condition becomes

```
> BC2a:=convert(BC2[2], fdiff,
> order=[2,0], forward=[0,0],
> indexletters=[n,none]): BC2a;
```

$$\frac{1}{2} \frac{3T_{11} - 4T_{10} + T_9}{h_x} = 0$$

The initial condition becomes

```
> inclist := [t=0, seq(T[m]=T[0],
> m=1..n)]: inclist;
```

```
[t = 0, T1 = T0, T2 = T0, T3 = T0, T4 = T0,
```

```
T5 = T0, T6 = T0, T7 = T0, T8 = T0,
```

```
T9 = T0, T10 = T0, T11 = T0]
```

We make a list of all the equations:

```
> eqnlist := [BC1a, seq(PDE1,
> i=2..n-1), BC2a]: eqnlist;
```

$$[T_1 = T_A, \frac{\partial}{\partial t} T_2 = \frac{\alpha(T_3 - 2T_2 + T_1)}{h_x^2},$$

... much more - similar - output omitted ...

$$\frac{\partial}{\partial t} T_{10} = \frac{\alpha(T_{11} - 2T_{10} + T_9)}{h_x^2}, \frac{1}{2} \frac{3T_{11} - 4T_{10} + T_9}{h_x} = 0]$$

This is a differential algebraic equation (DAE) system, not a purely ODE system. We can integrate this system using BESIRK which we now read into Maple.

```
> read 'e:/maple/numerics/integ/BESIRK':
```

The parameters in the model are given numerical values:

```
> params := {alpha=2e-5, T[0]=100, L=1,
> T[A]=0}: params;
```

$$\{\alpha = .00002, T_0 = 100, L = 1, T_A = 0\}$$

We now substitute the parameters into the list of equations:

```
> eqnlist2 := subs(hspec, params, eqnlist):
```

and into the list of initial values:

```
> inclist2:=subs(params, inclist):
```

The integration is carried out using the following command;

```
> parta := BESIRK(eqnlist2,
```

```
> inclist2, 0..6000):
```

The output from BESIRK is an array, the first row of which lists the names of each variable. Subsequent rows provide the numerical values of these variables. We can select some of the results with the `datatable` function from the BESIRK package. The output shown below this command has the same structure as the actual output, but the numbers have been rounded to 0.01 degrees Celcius. Also, the lines surrounding the numbers do not appear in the actual Maple output.

```
> datatable(parta, [t, seq(T[i],
```

```
> i=[1,3,5,7,9,11])]);
```

t	T_1	T_3	T_5	T_7	T_9	T_{11}
0	100	100	100	100	100	100
6000	0	31.71	58.49	77.46	88.22	91.66

BESIRK uses the 3rd order semi-implicit Runge-Kutta method of Michelsen (1976), combined with a Bulirsch-Stoer extrapolation technique. It is a very efficient integrator and found the solution at $t = 6000$ seconds in a single large integration step. While these results are in almost perfect agreement with the numerical results given by Cutlip and Shacham it is not possible to obtain an accurate plot of the temperature profiles with only two points. We repeat the computation with a limit on the maximum step size is to ensure that the resulting time history appears to be relatively smooth.

```
> parta := BESIRK(eqnlist2, inclist2,
```

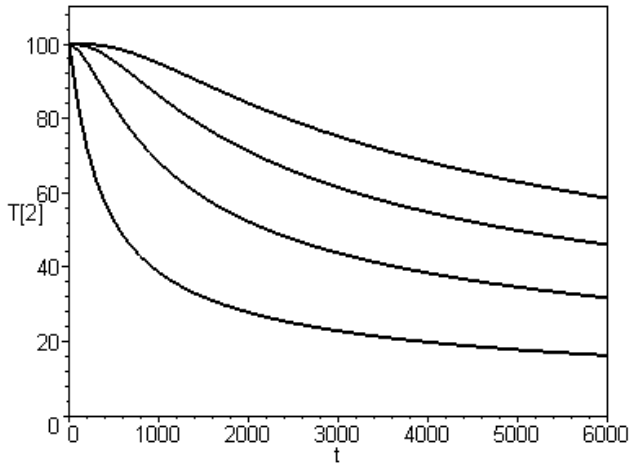
```
> 0..6000, hmax=100):
```

We may plot the solution as a function of the independent variable using the `datatableplot` procedure which is part of the `tplot` package. The second argument to this function is a list of the two variable names that are to be plotted.

```

> datatableplot(parta, [[t,T[2]],
> [t,T[3]], [t,T[4]], [t,T[5]]],
> color=black, axes=boxed,
> view=[0..6000,0..110], thickness=2);

```

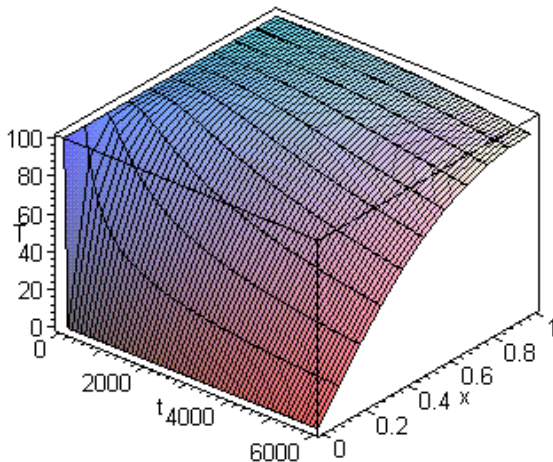


A 3D plot of the solution as a function of time using the MOLplot procedure which is part of the BESIRK package. Maple is capable of displaying surfaces in a variety of different styles (here Maple has used its default patch style).

```

> MOLplot(parta, subs(params,
> rhs(hspec)), 2..62, axes=boxed,
> contour, labels=['t', 'x', 'T'],
> orientation=[-50, 50]);

```



The actual time (in seconds on a 450 MHz Pentium PC) needed to compute this solution can be found using the following command:

```

> BESIRKtime;
2.333

```

The second part of the example in Cutlip and Shacham is to repeat the above calculations with 20 and 30 sections. To complete this part of the assignment it suffices to re-run the above commands, the sole change being to set the number of lines to 21 and 31 respectively. The third part of the problem is to repeat the first part where heat convection is present at the slab surface with $h = 25 \text{ W/m}^2\text{K}$ and a thermal conductivity (in the solid) of $k = 10 \text{ W/mK}$. The boundary condition for this situation becomes:

```

> BC1 := x=0, h*(T[infinity]-T(x,t))
> = - k*diff(T(x,t), x): BC1;

```

$$x = 0, h(T_\infty - T(x, t)) = -k \left(\frac{\partial}{\partial t} T(x, t) \right)$$

The boundary condition at the surface of the slab may be converted to finite difference form as follows:

```

> BC1a:=convert(BC1[2], fddiff,
> order=[2,0], forward=[2,0],
> indexletters=[1,none]): BC1a;

```

$$h(T_\infty - T_1) = -\frac{1}{2} \frac{k(-T_3 + 4T_2 - 3T_1)}{h_x}$$

Proceeding as before, the parameters in the model are given numerical values as follows

```

> params := {alpha=2e-5, T[0]=100, L=1,
> T[A]=0, T[infinity]=0, k=10, h=25};
> params;

```

$$\{a = .00002, T_0 = 100, L = 1, T_A = 0, k = 10, h = 25, T_\infty = 0\}$$

We form a list of all the equations (with the parameters included),

```

> eqnlist2 := subs(hspec, params,
> [BC1a, seq(PDE1, i=2..n-1), BC2a]);

```

The initial condition is unchanged from before.

```

> inclist2:=subs(params, inclist):

```

The numerical solution is obtained using BESIRK,

```

> partc := BESIRK(eqnlist2, inclist2,
> 0..6000, hmax=100):

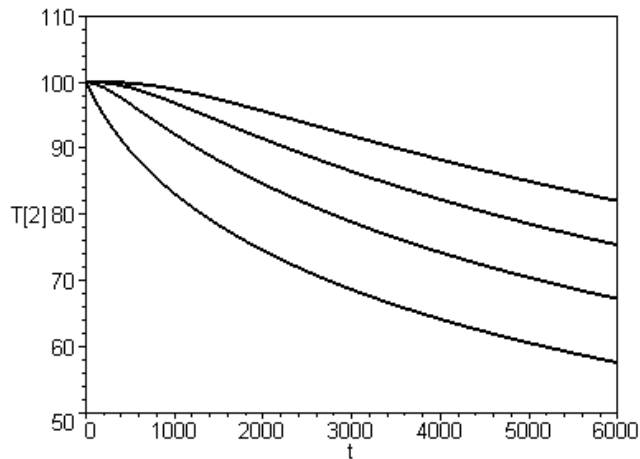
```

and the results displayed:

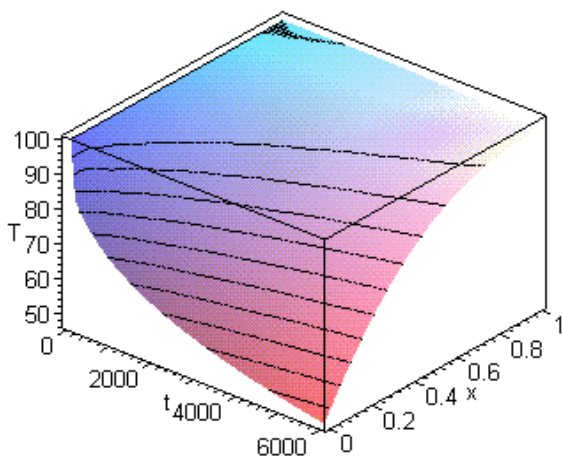
```

> datatableplot(partc, [[t,T[2]],
> [t,T[3]], [t,T[4]], [t,T[5]]],
> color=black, axes=boxed,
> view=[0..6000, 50..110], thickness=2);

```

```
> MOLplot(partc,subs(params,
> rhs(hspec)),2..62,axes=boxed,
> style=patchcontour,
> labels=['t','x','T'],
> orientation=[-50,50]);
```



Here we have used the `patchcontour` style for the 3D surface plot.

The Maple worksheet for the example described above and the code packages used here are available on the authors web site (<http://www.clarkson.edu/~chengweb/faculty/taylor/maple/>). Additional worksheets illustrating the application of Maple to solve PDEs by the method of lines or by finite difference methods also can be found there.

References

1. Cutlip, M.B., and M. Shacham. The Numerical Method Of Lines for Partial Differential Equations, *CACHE News*, No. 47, 18-21, 1999.
2. Michelsen, M.L. *AIChE J.*, **22**, 554, 1976.
3. Rosen, E.M. Excel 7.0: Partial Differential Equations, *CACHE News*, No. 47, 18-21, 1999.
4. Schiesser, W.E. *The Numerical Method Of Lines*, Academic Press, New York, 1992.
5. Schwalbe, D., H.A. Kooijman, and R. Taylor. Solving Stiff Differential Equations and Differential Algebraic Systems with Maple V Maple Tech, **3**(2), 47-53, 1996.
6. Silebi, C.E., and Schiesser, W.E., *Dynamic Modeling of Transport Process Systems*, Academic Press, New York, 1992.

Using Computational Fluid Dynamics as a Teaching Aid for Chemical Engineering

Liz Marshall, *Fluent Inc.*

Computational fluid dynamics (CFD) is the term given to the numerical solution of the Navier-Stokes equations of fluid motion. It encompasses not only the solution of the fluid equations and mass continuity, but also the transport of many other scalar variables that may be important in the flow. For example, the solution of heat transfer, species transport, reactions, turbulence, and multiple phases (solids and liquids) can be involved in a simulation. CFD has become widely accepted in the industrial workplace as both a design tool and troubleshooting device for many years. It has been used to model unit operations such as cyclone separators, spray dryers, stirred tanks, risers, bubble columns, valves, and packed bed reactors, for example. During this time, increases in computer speed, coupled with falling hardware prices, have made CFD readily available to university departments around the world as well. Initially used as a research tool, it is now entering the domain of the graduate and undergraduate curriculum to serve as a virtual laboratory or visualization tool for concepts being taught in the classroom. Used in a variety of ways, CFD can be employed to teach students about basic fluid behavior, to educate them on the workings of real-world commercial codes, and to train them on how to make the leap from process equipment to CFD model, through appropriate assumptions and approximations. Together, these skills improve the students' awareness of the physics involved in chemical engineering applications, and give them the advantage of hands-on CFD experience as they enter the workplace after graduation.

On an introductory level, CFD can be used to illustrate fluid flow fundamentals that are part of their coursework. Commercial codes with a graphical user interface are easy to learn to use with minimal training. Guided by one or two tutorials, students can quickly learn how to set up different models for a variety of application examples. One limiting factor, however, is the creation of the problem geometry and grid, which can be time-consuming for beginners. The grid is a set of intersecting lines that divide up the domain into small computational cells, where the conservation (transport) equations are solved using algebraic methods. The grid hurdle can be overcome through the use of pre-built grids, supplied to the students, or automatic grid (and model) building tools, such as those found in the software MixSim (for stirred tanks).

One simple example that is easy to do and offers insight into fluid behavior is laminar pipe flow. A simulation can be performed in a 2D axisymmetric environment, where the

computational domain extends from the pipe axis to the wall and for some arbitrary pipe length that exceeds the minimum required for uniform inlet flow to fully develop (say, 10 or 20 pipe diameters). The task for the students: to “measure” axial pressure gradient as a function of Reynolds number once the flow has become fully developed in the pipe (near the exit). From the pressure gradient, students can compute a resistance coefficient for pipe flow, λ (Schlichting, 1968):

$$\lambda = -\frac{2D}{\rho u^2} \frac{dP}{dx}$$

where D is the pipe diameter, ρ is the fluid density, and u is the average pipe velocity (or the inlet velocity). It can be shown that in the laminar regime, $\lambda = 64/\text{Re}$ (Schlichting, 1968). Students can test this dependence over a range of Reynolds numbers by computing an experimental value for λ and comparing to the analytic result. Given a pre-built grid, students can set the inflow boundary conditions and fluid properties and perform the first calculation, recording the computed Reynolds number and calculation results: pressure and x-position at two locations. The students can then change one variable, such as inlet velocity, compute the new Reynolds number, re-run the calculation, and record the new measurements for pressure at the same two locations. After 5 or 10 runs, the results for λ can be computed, plotted on a graph, and compared with the analytic solution (Figure 1).

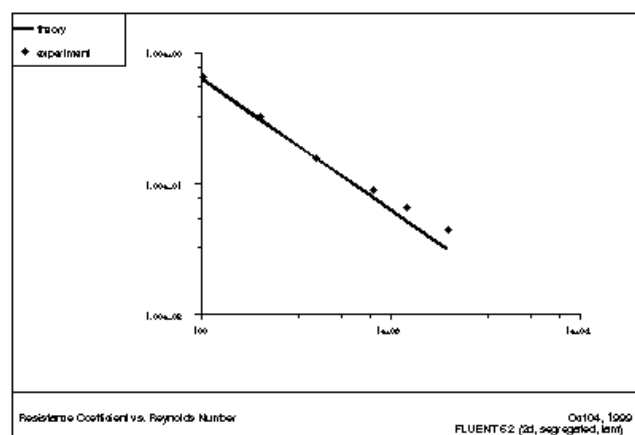


Figure 1. Plot of resistance coefficient, λ , as a function of Reynolds Number. “Experimental” (CFD) values are compared with theory.

Students can make some runs in the transitional regime ($Re > 2000$) to see when the simple dependence starts to break down. A similar exercise can be repeated for fully turbulent flow at a later time in the course when turbulence is introduced.

A second example for more advanced students illustrates the flow patterns produced by different impellers in a stirred tank. Two-dimensional simulations performed using MixSim can take advantage of a library of time-averaged impeller data for a large selection of impellers. The setup of a model takes minutes, and the calculation is automatically launched. Results can be compared through the use of streamlines or velocity vectors. For example, students can visualize the difference between the flow field produced by a Rushton turbine (Figure 2) and one produced by a pitched-blade turbine (Figure 3) (both compliments of Lightning) operating in the same tank and at the same speed.

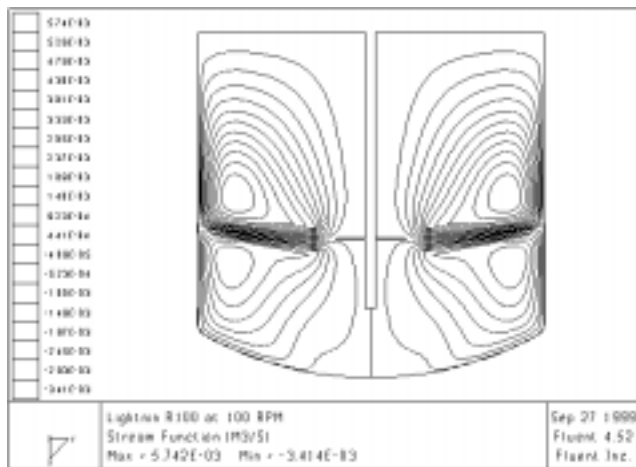


Figure 2. The flow pattern created by a Rushton turbine

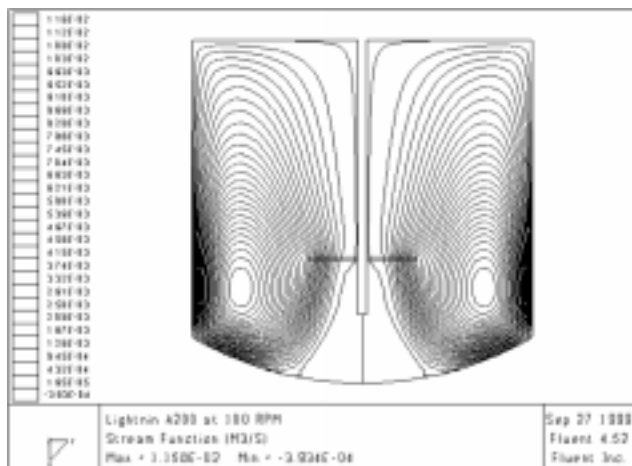


Figure 3. The flow pattern created by a pitched-blade turbine

Blending times can be computed in a transient calculation that uses a frozen flow field, and tracks the species concentration in the tank (at point locations or as average values with standard deviation). The flow for the pitched-blade turbine is used to illustrate this. In Figure 4, a small amount of tracer is added to the tank as a layer across the top.

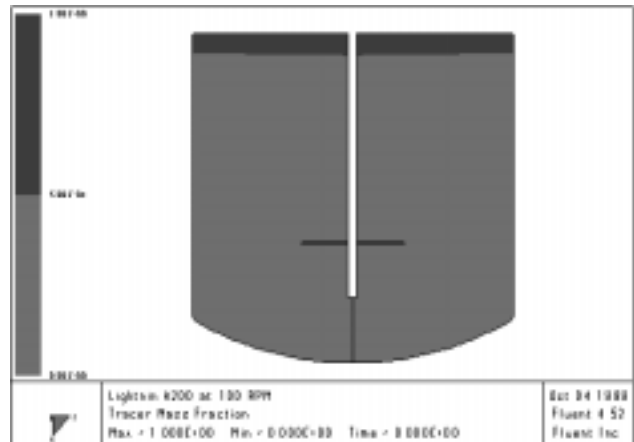


Figure 4. A layer of tracer species is added to the top of the vessel at time $t=0$.

The properties of the tracer are identical to those of the background fluid in the tank. With this assumption, the fluid properties will not change as the fluids mix, so the converged flow field shown in Figure 3 can be used for a transient species calculation to track the blending. In Figure 5, the ratio of the standard deviation to the average mass fraction of tracer is shown to tend toward zero as time increases and the mixture becomes more uniform.

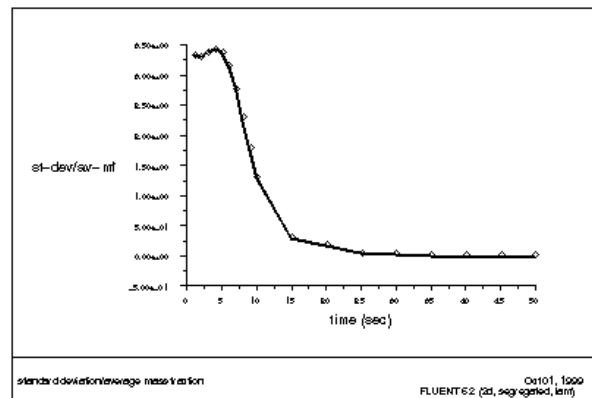


Figure 5. The ratio of standard deviation to average value of tracer mass fraction is plotted against time, showing nearly complete blending after 50 seconds.

The “measured” value of about 50 seconds for complete blending can be compared to a correlation for the time to achieve 99% uniformity in a vessel with a single pitched blade turbine (Fasano, et al., 1994):

$$t_{99} = \frac{4.605}{0.641N \left(\frac{D}{T}\right)^{2.19} \left(\frac{T}{Z}\right)^{0.5}}$$

where N is the shaft speed in revolutions/sec, D is the impeller diameter, T is the tank diameter, and Z is the liquid height. Using the parameters for the case illustrated here, a time of 60 seconds is predicted to achieve 99% uniformity. The CFD result is in reasonably good agreement with the correlated value. As additional exercises, students can vary the parameters in the above equation (and repeat the CFD analysis) to see if the correlation correctly predicts the trends (i.e. more rapid blending for faster shaft speed, and so on).

A third example allows students to visualize flow patterns inside 3 dimensional objects that would ordinarily be difficult to observe by ordinary means. For example, the blending of two species in a tee-junction can be inferred by measurement in the downstream section of pipe, but the details at the site of the junction itself are difficult to probe. With CFD on a ready-built 3D grid, students can examine 3D surfaces of species concentration (or contours of mass fraction on a plane, as is shown in Figure 6) to illustrate the effects of incoming velocity ratio or fluid property ratios on the mixing. They can also see the extent of any recirculation that forms downstream of the junction.

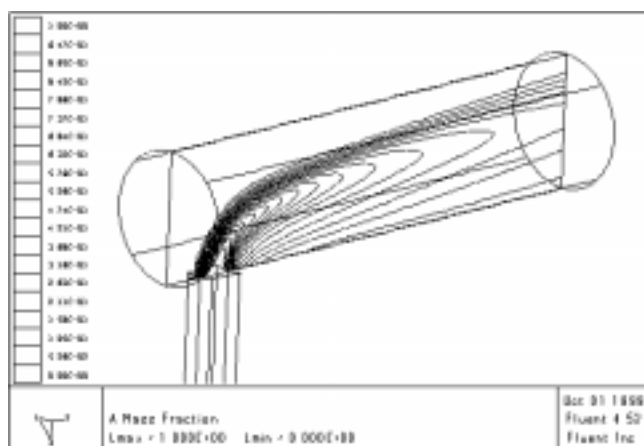


Figure 6. Introduction of “species a” into a pipe at a 3D tee-junction is easily visualized using CFD

Examples such as this provide visual reinforcement to concepts learned in the classroom, and for many students, it is the visual connection that helps them remember the key concepts for later use.

Other valuable experience can be gained from CFD exercises in coursework. First, there is the familiarity with the tool itself. In-school experience helps shorten the learning curve for continued use of CFD codes on the job. Second, while the basics of the more advanced numerical methods used need not be a part of the undergraduate curriculum, much of the basic formulation can help students better understand broader concepts. For lessons in applied mathematics, for example, consider the transformation of a differential equation into an algebraic conservation equation. This lesson can be demonstrated on the board for a unit cell, start-

ing with a volume integration on the cell that is converted to a surface integration by way of the divergence theorem. Students can then perform “experiments” on simple CFD problems that use only a handful of rectangular cells. Students can record fluxes of conserved quantities such as mass and momentum into and out of a single cell (or a group of cells) to demonstrate that the laws of conservation are obeyed. At a higher level, students can explore the differences in face values as they are generated by a variety of interpolation schemes. (Interpolation schemes are used to convert cell-centered values of problem variables to cell-face values.) In the process, they can understand when a linear interpolation between point values works (heat conduction in a uniform material) and when it doesn’t (pipe flow), and why. Parallels can be drawn between similar systems, and contrasts can be made between different ones. Students will emerge with a better sense of the underlying physics governing the systems studied.

Despite the fact that CFD can be used in many ways as a virtual laboratory, it serves another important purpose that should not be overlooked. It forces the student to think about the process to be modeled in an objective way, in which the individual components are “ranked” in order of significance. The student must first ask: What am I trying to learn from the simulation? Once this direction is established, other questions follow. For example, is a 2D model possible? If so, what flow features will be missing? Where should the computational domain boundaries be placed? Where is a fine grid needed and where can a coarse grid be used? Is the gas law required and why or why not? Answers to these questions force the student to think about a process in a manner that might not happen otherwise. It is critical thinking such as this that helps to cement the important features of unit operations into the student’s experience base, for later use on the job.

In summary, CFD is a tool that is ready to be exploited in the chemical engineering curriculum at both the undergraduate and graduate levels. Computers are inexpensive and powerful enough that simulations of significance can be run as an aid to classroom teaching. There are many levels of exposure that are possible, from fundamental to advanced, and at each level, valuable insight can be gained into the systems being studied in the classroom. As an added benefit, students will have fun with CFD, particularly if they can make color images or animations of fluid behavior that they will remember for years to come.

References

- Fasano, J. B., A. Bakker, and W. R. Penney, “Advanced Impeller Geometry Boosts Liquid Agitation,” *Chemical Engineering*, Aug. 1994.
- Schlichting, Hermann, *Boundary Layer Theory*, New York: McGraw-Hill Book Company, 1968.

ANNOUNCEMENTS

Process Design Case Study Volume 7

Design of an Ethylbenzene Production Plant Preliminary Design and Economic Analysis

The objective of this case study is the preliminary design and economic evaluation of an alkylation process for the production of ethylbenzene from ethylene and benzene. The intent is to expose the student to a traditional chemical engineering process. The scope of the study is such that groups of 3-4 students should be able to complete the design in 3 months. The major focus of this design is the creation and development of a suitable process flowsheet, simulation of the flowsheet by the simulator HYSYS, economic evaluation and cost minimization, and HAZOP analysis of the final process.

The problem begins with a statement of the design specifications. Students are then required to consider alternatives for the reactor type (fixed or fluidized bed), reactor conditions (liquid or vapor phase) and separation scheme. These choices affect the layout of the flowsheet and lead to several design alternatives. Students are expected to analyze the various process alternatives based on economic, safety, and environmental factors. Included in this case study is a floppy disk containing the HYSYS simulation file and GAMS input files for modeling the heat integration.

The problem statement was posed by Professor Costas D. Maranas and prepared by James E. Lease, Gregory L. Moore, Ryan M. Scofield and Joseph D. Sevick in the department of Chemical Engineering at the Pennsylvania State University.

To order this case study, please fill out form below:

Name: _____

Address: _____

This case study is available in two forms:

- (a) Hardcopy with binder and disk with HYSYS input file
CACHE Supporting department \$40
Non-supporting department \$80
- (b) CD-ROM containing text of case study and HYSYS file. Large flowsheet diagram is also provided.

CACHE Supporting department \$15
Non-supporting department \$30

Number of copies option (a) _____

Number of copies option (b) _____

Total Price _____

Please send order and payment to:

The CACHE Corporation
P.O. Box 7939
Austin, Texas 78713-7939
Fax: (512) 295-4498

ANNOUNCEMENTS

Visual Encyclopedia of Chemical Engineering Equipment for Macintosh and Windows 95/NT

Susan Montgomery
University of Michigan, Chemical Engineering Department,
Multimedia Education Laboratory

Introduction

We have developed a visual encyclopedia of chemical engineering equipment for use in all undergraduate chemical engineering courses. The purpose of this encyclopedia is to provide students with a basic understanding of what chemical engineering equipment looks like and how it works. It is intended for use for use as a general reference for students in courses spanning the introductory course through the capstone design course, as well as for classroom presentation, for example prior to covering a given chemical engineering topic.

The over 100 pieces of chemical engineering equipment are organized into the nine categories shown in Table 1, and an index/glossary is also provided for ease of access.

Table 1 - Categories of Chemical Engineering Equipment

- Process Parameters
 - Flowmeters
- Transport and Storage
- Chemical Separations
- Mechanical Separations
- Heat Transfer
- Reactors
- Polymer Processing
- Materials Handling

Organization

The description for each piece of equipment is divided into: General Information, Equipment Design, Usage Examples, Advantages / Disadvantages, and References. This leads the user from a brief description of how the equipment works to more in-depth details of its operation; examples of industries in which the equipment is used; comparisons with other types of equipment; and resources for more information. Note that information needed to design equipment is not included.

At the heart of the encyclopedia are pictures of actual equipment, obtained from equipment manufacturers, animations showing how the equipment works, and pictures of applications of the equipment in a wide variety of fields. Figures 1 through 4 show typical screens from some of the applications.

For more information on the encyclopedia and other materials produced by our laboratory and distributed by CACHE, please visit <http://www.engin.umich.edu/labs/mel>. It contains an online example that demonstrates a subsection of the encyclopedia.

Acknowledgments

This encyclopedia was developed with funds from National Science Foundation Grant 9555125. Their support is gratefully acknowledged.

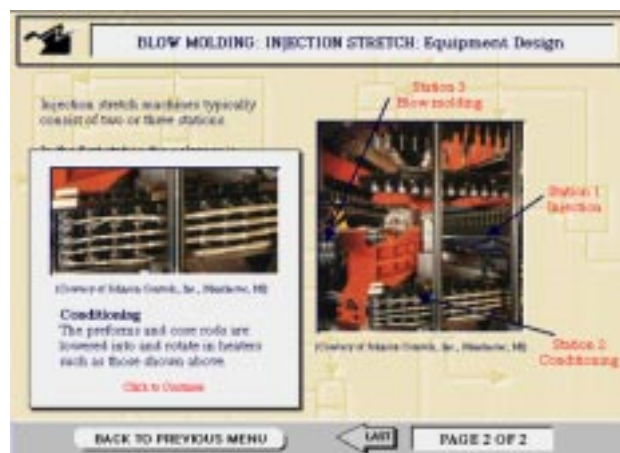


Figure 1. Equipment design of blow molding with clickable pop-up windows

ANNOUNCEMENTS



Figure 2. Animation depicting an inverted bucket steam trap in action

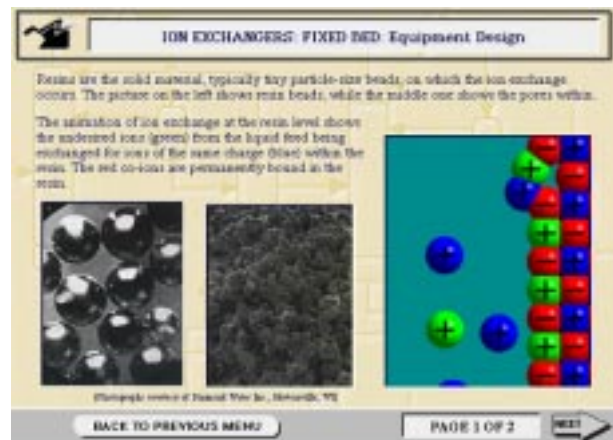


Figure 3. Introduction to fixed bed ion exchange



Figure 4. Typical usage example screen

ANNOUNCEMENTS

Material and Energy Balances Multimedia Modules for Macintosh and Windows 95/NT

Susan Montgomery
University of Michigan, Chemical Engineering Department
Multimedia Education Laboratory

We have developed a set of multimedia computer modules to help students in the introductory chemical engineering balances course understand and practice the key concepts of the course. Modules are divided into two categories, as shown in Table 1, and include modules that allow students to solve material balance problems in three different applications, and modules that provide practice using charts and used to solve material and energy balance problems. These modules are described in more detail below.

Table 1 - Summary of Multimedia Modules

Material Balance Problems

- Biological Systems
- Car pre-paint system
- Wastewater treatment plant

Chart Tutorials

- Pxy and Txy diagrams
- Psychrometric charts
- Enthalpy-concentration diagrams

Biological Systems

This module helps students in the introductory chemical engineering course understand and practice the application of material balances to biological systems of different dimensions using an algorithm for the solution of material balance problems. In-depth interactions focus on mass balance problems involving cell metabolism, an artificial kidney, and a cheese factory, as shown in Figure 1.

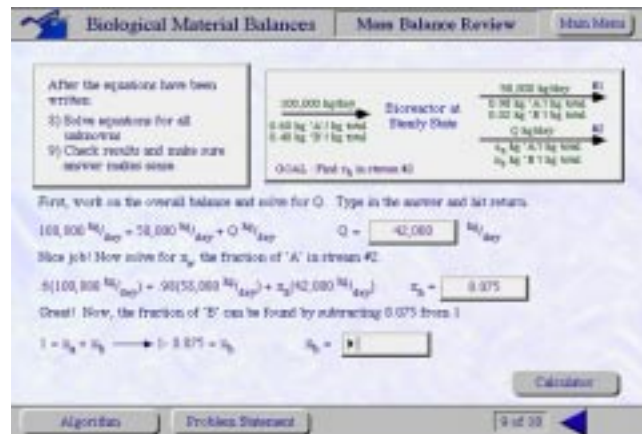


Figure 1. Sample interaction screen, biological systems module

Car Pre-Painting System

This multimedia tour is used as part of an open-ended project for a material balance course, in which students create the flowsheet for the phosphate coating system at Ford Motor Company's Wixom plant, then suggest modifications for waste minimization. A sample data screen is shown in Figure 2.

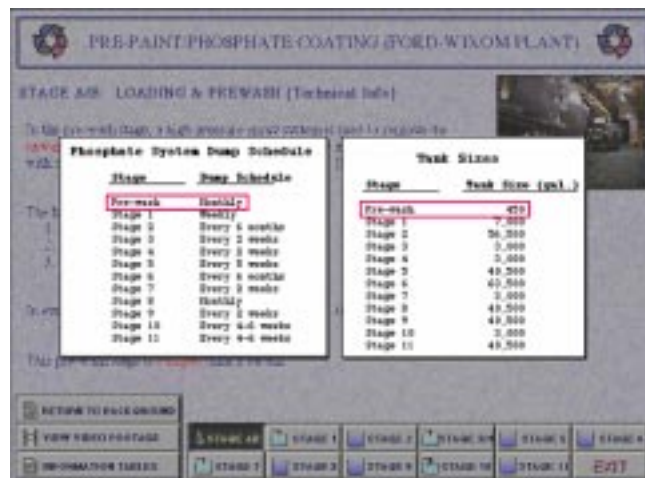


Figure 2. Data sheet for a process step, car pre-painting system module

Wastewater Treatment Plant

This module introduces students to the applications of material balances in wastewater treatment processes. After reviewing the basics of wastewater treatment, students can take a tour of the Ann Arbor Wastewater Treatment Plant. A quiz includes material balance questions corresponding to each section of the plant, such as the one shown in Figure 3.

ANNOUNCEMENTS



Figure 3. Mass balance on secondary treatment process, wastewater treatment plant module

P-xy and T-xy Diagrams

This interactive module helps students understand the use of Pxy and Txy diagrams in describing binary liquid-vapor systems with tutorials as shown in Figure 4. The module also includes an introduction to industrial uses, and three quizzes to evaluate their understanding of the diagrams and their use.

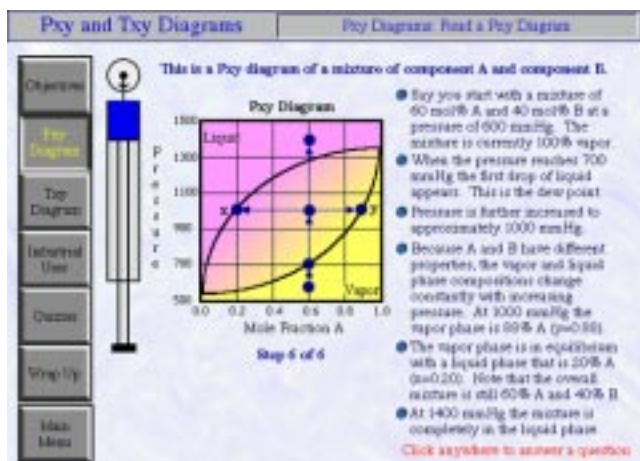


Figure 4. Tutorial on reading Pxy diagrams, Pxy and Txy diagrams module

Psychrometric Charts

Using this module, students learn how to read and use psychrometric charts, and are exposed to industrial applications in which they are used, as shown in Figure 5. Three quizzes help students evaluate whether they understand the components of the chart, how to locate a point on the chart, and how to follow a process using the chart.



Figure 5. Cooling tower usage example, psychrometric charts module

Enthalpy Concentration Diagrams

This module helps students review how to read and use enthalpy concentration diagrams, and explore applications in which enthalpy concentration diagrams are used. A quiz helps students determine if they understand the components of the diagram, how to locate a point on the diagram, and how to follow a process using the diagram to solve material and energy balance problems. A sample quiz question is shown in Figure 6.

ANNOUNCEMENTS

The screenshot shows a quiz question: "1. 40 lb_m of a 50-mass% NH₃ solution is heated to 160°F. What is the amount of vapor present?" The options are 22 lb_m, 30 lb_m (circled in red), 10 lb_m, and 20 lb_m. The correct answer is 30 lb_m. The interface includes buttons for "Objective", "Diagram", "Time", "Quiz", and "Main Menu". A "Correct!" message states: "All you had to do was use the lever rule to find V". The lever rule calculation is shown as:
$$\frac{V}{F} = \frac{x_2 - x_1}{x_2 - x_3}$$

$$V = 0.75 \times 40 \text{ lb}_m = 30 \text{ lb}_m$$
 A "Click here to see an enlarged diagram" link is present. To the right of the question is a graph of Specific Enthalpy (Btu/lb_m) vs. Mass fraction NH₃ (x) showing a vapor-liquid phase diagram with tie-lines and a lever rule construction.

Figure 6. A multiple choice energy balance quiz question, Enthalpy concentration diagrams module

Instructor's Booklet

A booklet that accompanies these modules describes in more detail each of the sections, and also shows which chapters in Felder and Rousseau's and Himmelblau's textbooks [1, 2] each module corresponds to.

For more information on this CD and other materials produced by our laboratory and distributed by CACHE, please visit <http://www.engin.umich.edu/labs/mel>.

Acknowledgements

This CD was developed with funds from National Science Foundation Grant 9555125. Their support is gratefully acknowledged.

References

1. Felder, R.M. and R.W. Rousseau, Elementary Principles of Chemical Processes, 3rd ed. New York: Wiley, 1999.
2. Himmelblau, D.M. Basic Principles and Calculations in Chemical Engineering, 6th ed. Upper Saddle River, NJ: Prentice-Hall, 1996.

CACHE Products

To order CACHE Products, complete the Standard Order Form and send with payment to:

CACHE Corporation
P.O. Box 7939
Austin, TX 78713-7939

FAX: (512) 295-4498
Phone: (512) 295-2708
Email: cache@uts.cc.utexas.edu

We accept credit cards (Visa/Mastercard), purchase orders and checks.

Overseas orders are sent surface at no charge, airmail is extra.

ANNOUNCEMENTS

7th International Symposium on Process Systems Engineering (PSE)

**July 16-21, 2000
Keystone Resort & Conference Center
Keystone, Colorado, USA**

Sponsored by
CACHE Corporation
Computing & Systems Technology Division
of American Institute of Chemical Engineers

In association with
Asian Pacific Confederation of Chemical Engineering
European Federation of Chemical Engineering
Inter American Confederation of Chemical Engineering

Objective and Scope

PSE 2000 is the seventh in the triennial series of international symposia on process systems engineering initiated in 1982. The purpose of the meeting is to bring together the community of researchers and practitioners who are involved in the creation and application of computing based methodologies for planning, design, operation, control, and maintenance of chemical processes. The special focus of PSE meetings is on the integration of the enabling technologies and application domains to address the special needs of the chemical process industries. Particular emphasis is given to the integration of and interfaces between application domains and the adaptation of process systems methodologies to nontraditional problem areas.

The composition of the meeting is international by design, with representation from the three main geographic zones of Asia and the Pacific, Europe and Africa, and the Americas. The conference was initiated by the Executive Committee of the Process Systems Engineering Symposium Series which draws its representation from the Asian Pacific Confederation of Chemical Engineering, the European Federation of Chemical Engineering, and the Inter American Confederation of Chemical Engineering. In keeping with the international scope of the conference series, the previous conferences in the series were held in Trondheim, Norway (1997); Kyongju, Korea (1994); Montebello, Canada (1991); Sydney, Australia (1988), Cambridge, England (1985); and Kyoto, Japan (1982). PSE 2000 will be the first time the conference is convened in the US.

Location

The Keystone Resort & Conference Center is located in the heart of the Rocky Mountains, yet only 90 minutes west of Denver, Colorado, on Interstate 70. The Resort, located in the ski resort of Keystone, a few miles from Lake Dillon, is serviced by regularly scheduled limo/mini-bus transportation directly from the Denver International Airport. The Conference Center, the largest and most versatile such complex in the Rocky Mountains, has received national awards for its facilities and amenities. A full range of accommodations from hotel rooms to condominiums and over 25 restaurants are available, all within a short walking distance from the Conference Center. A full range of outdoor activities are available to guests ranging from championship golf, tennis, hiking, mountain biking, white water rafting, horse-back riding, 4-wheel drive touring, and sailing.

Program Structure

The conference will open with a reception on Sunday evening, July 16, 2000 and will close at noon on Friday. The program will feature single track plenary sessions in the mornings and parallel sessions consisting of oral and poster presentations in the evenings. The morning sessions will consist of a keynote speaker followed by three plenary speakers. The plenary papers will be selected by the Program Committee from among the papers contributed to the conference on the basis of technical quality and innovativeness. The contributed program is anticipated to consist of 2/3 oral presentation and 1/3 poster presentations.

ANNOUNCEMENTS

Topics of interest include but are not limited to the following:

Integration of Design & Control

- Design for controllability
- Design for safety & environment
- Design of intentionally dynamic processes

Integration of Control & Operations

- Real time dynamic plant optimization
- Abnormal situation management
- Supply chain optimization

Integration of Design & Operations

- Design for maintainability
- Multipurpose facilities
- Design & verification of operating procedures

Non-traditional applications of process systems engineering

- Materials & formulation design
- Systems methodology in bioinformatics
- Systems approaches to business decisions

Education in process systems engineering

- Role of process systems in ChE education
- Innovations in computer-aided educational tools
- Training of practitioners in new systems tools

Special attention will be given to contributions from industry and to developments which feature a collaboration between industry and university researchers. Educational developments for both beginners and practitioners are welcomed.

Application & Fees

The conference application fee is \$495 for registrations received before April 15, 2000 and \$595 for registrations received after that date. The conference fee includes one copy of the proceedings volume, the Sunday conference opening reception, the conference banquet (Thursday evening), and refreshments during program breaks. A separate meal plan will be available for attendee selection.

For additional information, please refer to the conference web page: <http://www.atom.ecn.purdue.edu/~pse2000/>
or send an email to pse2000@ecn.purdue.edu

ANNOUNCEMENTS

Foundations of Molecular Modeling and Simulation (FOMMS): Applications for Industry

July 23-28, 2000
Keystone Resort
Keystone, Colorado, USA

Presented by
CACHE Corporation and
Thermodynamics and Transport Properties
Programming Group of the American Institute of
Chemical Engineers

Sponsors
U.S. National Science Foundation
Molecular Modeling Program of the Mexican
Petroleum Institute (IMP)
Molecular Simulations, Inc.
Warner-Lambert Company
U.S. National Institute of Standards and Technology
Taylor and Francis (publishers of Molecular Physics)

Background

Computational quantum chemistry and molecular simulation methods have now become useful tools for the chemicals, fuels, biologics, and materials industries. In the coming decades, we even can look forward toward the possibility of molecular design of products and materials, seamlessly integrated with the design of the processes to manufacture them. These developments have come about because of fundamental advances in theory, methods and computing technology.

In recognition of these extraordinary developments and opportunities, a new conference entitled "Foundations of Molecular Modeling and Simulation" (FOMMS) has been established by the non-profit educational foundation, CACHE Corporation, in collaboration with the Thermodynamics and Transport Properties programming area of the American Institute of Chemical Engineers. The first FOMMS conference will be held at Keystone Resort, CO, on July 23-28, 2000, with the theme "Applications for Industry".

The target audience for the conference includes practitioners (both from industry and academia), those interested in becoming practitioners (both in industry and academia), and industrial managers who wish to learn about the technology. Content will be balanced between molecular simulation and computational chemistry, along with a look to the future of chemical product design and the supporting computational environments.

All talks are invited, and there are no parallel sessions. Two poster sessions will provide opportunities for attendees to present their work. One afternoon will be devoted to a software/hardware demonstration session for providers to showcase their products and services. Large blocks of time, including receptions, will be available for informal discussions, aiding interaction between conference participants.

A proceedings volume will be produced and published as part of the CACHE proceedings series. Each invited talk will represent a reviewed and edited manuscript that will serve as a state-of-the-art review. Contributed posters will be represented by peer-reviewed shorter contributions. We aim to produce a proceedings volume that will be a landmark publication in the field.

Tentative Conference Oral Session Titles

Bridging Scales from the Molecular to the Process Level
Applying Molecular Modeling and Simulations in Industry
Molecular Simulations in Biological Applications
Molecular and Materials Modeling
Simulating Polymers
Developing Force Fields
Bridging Quantum Chemistry and Molecular Simulations
Catalysis and Reaction Engineering
Looking to the Future

ANNOUNCEMENTS

Confirmed Invited Speakers

Alexis T. Bell, *University of California, Berkeley*
Jozef Bicarano, *Dow Chemical Engineering*
J. Stephen Binkley, *Sandia National Laboratories*
Ann M. Chaka, *Lubrizol*
Anthony M. Dean, *Exxon Research and Engineering Co.*
Michael F. Doherty, *University of Massachusetts*
Jack Dongarra, *University of Tennessee*
Richard A. Friesner, *Columbia University*
Keith E. Gubbins, *North Carolina State University*
Gary A. Huber, *University of California-San Diego*
Klavs F. Jensen, *Massachusetts Institute of Technology*
Daniel Kleier, *DuPont*
Stephen L. Mayo, *California Institute of Technology*
Keiji Morojuma, *Emory University*
Jens K. Norskov, *Technical University of Denmark*
Julio M. Ottino, *Northwestern University*
Athanasios Z. Panagiotopoulos, *Univeristy of Maryland*
Michele Parrinella, *Max-Planck-Institut fur Festkorperforschung*

Gregory J. McRae, *Massachusetts Institute of Technology*
Jack A. Smith, *Union Carbide Corporation*
Ulrich W. Suter, *ETH Zurich*
Dominic J. Tildesley, *Unilever*
Donald G. Truhlar, *University of Minnesota*

Application and Fees

To promote an atmosphere of small group interactions, attendance at FOMMS 2000 will be limited to 200 participants and is by invitation based on this application. Applications must be postmarked by January 15, 2000. Invitations, which will contain detailed registration and payment information will be sent as soon as your application is accepted, but no later than February 1, 2000. The conference fee includes registration, proceedings, welcoming reception, hospitality and coffee breaks, and the conference banquet. The conference fee is \$675 for registrations paid on or before March 15, 2000 and \$775 for registrations paid between march 16, 2000 and May 14, 2000. Registrations and fees will not be accepted after June 1, 2000.

For more detailed information please see the web site at <http://www.ecs.umass.edu/FOMMS>

Industrial Contributors to CACHE

AEA/Hyprotech

Air Products and Chemicals

Aspen Technology

Dupont

Eastman Chemical Company

Fluent Inc.

Merck & Company

Mobil Technology Company

Parke-Davis

Union Carbide