

A COMPARISON OF SIX NUMERICAL SOFTWARE PACKAGES FOR EDUCATIONAL USE IN THE CHEMICAL ENGINEERING CURRICULUM

Mordechai Shacham
Department of Chemical Engineering
Ben-Gurion University of the Negev
P. O. Box 653
Beer Sheva 84105, Israel
Tel: (972) 7-6461481 Fax: (972) 7-6472916
E-mail: shacham@bgumail.bgu.ac.il

Michael B. Cutlip
Department of Chemical Engineering
University of Connecticut
Box U-222
Storrs, CT 06269-3222
Tel: (860)486-0321 Fax: (860)486-2959
E-mail: mcutlip@uconnvm.uconn.edu

INTRODUCTION

Until the early 1980's, computer use in Chemical Engineering Education involved mainly FORTRAN and less frequently CSMP programming. A typical computer assignment in that era would require the student to carry out the following tasks: 1.) Derive the model equations for the problem at hand, 2.) Find an appropriate numerical method to solve the model (mostly NLE's or ODE's), 3.) Write and debug a FORTRAN program to solve the problem using the selected numerical algorithm, and 4.) Analyze the results for validity and precision.

It was soon recognized that the second and third tasks of the solution were minor contributions to the learning of the subject material in most chemical engineering courses, but they were actually the most time consuming and frustrating parts of computer assignments. The computer indeed enabled the students to solve realistic problems, but the time spent on technical details which were of minor relevance to the subject matter was much too long.

In order to solve this difficulty, there was a tendency to provide the students with computer programs that could solve one particular type of a problem. Listings or even disks containing small size programs were included in textbooks, and large scale commercial simulation programs were made available to students. This approach had the disadvantage that the programs were used in a "black box" manner where students only provided the input data and observed the result. The very important step of converting a physical phenomena to a mathematical model was missing, thus the connection between the mathematical model and the problem was obscured.

In the early eighties interactive numerical software packages with graphic capabilities began to emerge as the major computer tool for engineering problem solving. For example, Shacham and Cutlip⁵ developed such a package for the PLATO educational computer system. It was soon transported to the personal computer (Shacham et al.⁶) and has been used since under the name of "POLYMATH."

When using an interactive software package, the student is required to input the mathematical model and the numerical data, but the programs carry out all of the technical steps of the solution. These advantages of mathematical packages have led to the replacement of calculators and spreadsheets as the main calculational tool in many courses (for example, see Fogler²).

However, after some fifteen years since the availability of interactive software packages, the level of integration into undergraduate engineering courses is still disappointingly low. A recent survey by Jones⁴ has indicated that "Across the country, computers are usually not used effectively in undergraduate engineering science course. Often they are not used at all. Problem solving approaches and calculation methods are little influenced by the availability of computers." What are the reasons for this low level of computer use? Part of the answer is that educators are not familiar enough with these relatively new computational tools that integration into their courses is challenging. Another possible aspect is that the packages are not as user friendly and easy to use as they claim to be.

Many chemical engineering departments now select one or several interactive computational packages to be used by their students. The selection of a particular package for use is accomplished mostly based on an instructor's personal preference. The identification of a package for general departmental use is typically done by the departmental computer expert. Often the expert will select the package which is the most powerful, most flexible, and gives the user the widest range of options. Typically some research usage has strong emphasis. However, for educational use of undergraduates, other attributes of the package may be more important. The package must have enough capabilities to solve most problems encountered in undergraduate education but it must be user friendly, must have a short learning curve, and must require only minimal user intervention in the technical details of the solution process.

In order to familiarize engineering educators with the most widely used packages and to assist in the selection of the most beneficial packages to be used in core courses in the chemical engineering curriculum, we have assembled a set of ten benchmark problems. This problem set was utilized in conjunction with an ASEE Summer School for Chemical Engineering Faculty which was held in Snowbird, UT in August of 1997.

This paper will give a general overview of the benchmark problems that have been formulated and will identify the mathematical software packages that were used in the problem solution comparisons. Various criteria for comparison will be described, and ratings for the software packages will be made according to the developed criteria. This will be followed by conclusions based on the comparison.

PROBLEM SET DEVELOPED FOR THE COMPARISON

Table (1) summarizes the complete problem set which was developed for comparison purposes. Within this set there are representative problems from almost every required course in a typical chemical engineering curriculum. The mathematical models required for defining the various problems include:

1. Single and several simultaneous nonlinear algebraic equations (NLE's).
2. Simultaneous linear equations.
3. Simultaneous ordinary differential equations (ODE's) with known initial conditions and with split boundary values.
4. Simultaneous ordinary differential and nonlinear algebraic equations (DAE's).
5. Stiff ODE's.
6. Polynomial curves fitting
7. Linear, multiple linear and general nonlinear regression.

The above capabilities represent the various types of numerical analysis capabilities which are normally presented to undergraduate students in chemical engineering.

Table 1 Selection of Problems Solutions Illustrating Mathematical Software

COURSE	PROBLEM TITLE	MATHEMATICAL MODEL	PROBLEM
Introduction to Ch. E.	Molar Volume and Compressibility Factor from Van Der Waals Equation	Single Nonlinear Equation	1
Introduction to Ch. E.	Steady State Material Balances on a Separation Train*	Simultaneous Linear Equations	2
Mathematical Methods	Vapor Pressure Data Representation by Polynomials and Equations	Polynomial Fitting, Linear and Nonlinear Regression	3
Thermodynamics	Reaction Equilibrium for Multiple Gas Phase Reactions*	Simultaneous Nonlinear Equations	4
Fluid Dynamics	Terminal Velocity of Falling Particles	Single Nonlinear Equation	5
Heat Transfer	Unsteady State Heat Exchange in a Series of Agitated Tanks*	Simultaneous ODE's with known initial conditions.	6
Mass Transfer	Diffusion with Chemical Reaction in a One Dimensional Slab	Simultaneous ODE's with split boundary conditions.	7
Separation Processes	Binary Batch Distillation**	Simultaneous Differential and Nonlinear Algebraic Equations	8

Table 1 Selection of Problems Solutions Illustrating Mathematical Software

COURSE	PROBLEM TITLE	MATHEMATICAL MODEL	PROBLEM
Reaction Engineering	Reversible, Exothermic, Gas Phase Reaction in a Catalytic Reactor*	Simultaneous ODE's and Algebraic Equations	9
Process Dynamics and Control	Dynamics of a Heated Tank with PI Temperature Control**	Simultaneous Stiff ODE's	10

* Problem originally suggested by H. S. Fogler of the University of Michigan

** Problem preparation assistance by N. Brauner of Tel-Aviv University

MATHEMATICAL PACKAGES USED IN THE COMPARISON

Table 2 lists the various mathematical software packages that were used in the solutions to the problem set and the individuals who kindly provided the solution set. These individuals had considerable experience with the software package that they utilized. The major reference to these packages is given for the world wide web (WWW) so that information can easily be obtained. In many cases, the latest software is utilized, and capabilities beyond the student or academic versions are required. Many of these packages have academic site licenses which are relatively inexpensive for institutional computers; however, often the professional versions become quite expensive for individuals. Detailed cost information is available from each vendor. A major consideration for academic use is the license, if available, which allows students to place the software on their individually owned personal computers.

Table 2 Mathematical Software Packages Utilized

Mathematical Software Package	WWW Reference	Solutions Provided by
Excel 7.0 ^a	Microsoft Corporation (http://www.microsoft.com)	Edward M. Rosen, EMR Technological Group
Maple ^b	Waterloo Maple, Inc. (http://www.maplesoft.com)	Ross Taylor, Clarkson University
Mathcad ^c	Mathsoft, Inc. (http://www.mathsoft.com)	John J. Hwalek, University of Maine
Mathematica ^d	Wolfram Research, Inc. (http://www.wolfram.com)	H. Eric Nuttall, University of New Mexico
Matlab ^e	The Math Works, Inc. (http://www.mathworks.com)	Joseph Brule, John Widmann, Tae Han, and Bruce Finlayson, University of Washington
Polymath ^f	The CACHE Corporation (http://www.che.utexas/cache/)	Michael B. Cutlip, University of Connecticut Mordechai Shacham, Ben-Gurion University of the Negev, Israel

^aExcel is a trademark of Microsoft Corporation (<http://www.microsoft.com>)

^bMaple is a trademark of Waterloo Maple, Inc. (<http://www.maplesoft.com>)

^cMathcad is a trademark of Mathsoft, Inc. (<http://www.mathsoft.com>)

^dMathematica is a trademark of Wolfram Research, Inc. (<http://www.wolfram.com>)

^eMATLAB is a trademark of The Math Works, Inc. (<http://www.mathworks.com>)

^fPOLYMATH is copyrighted by M. B. Cutlip and M. Shacham (<http://www.che.utexas/cache/>)

MATERIALS AVAILABLE FROM THE INTERNET

This complete paper, the entire problem set, technical write-ups for each mathematical software package, and problem solution files for each package are available via FTP from **ftp.engr.uconn.edu** in directory **/pub/ASEE**. The written materials are only readable in Adobe Acrobat 3.0 format and higher; however, this software is free via the internet from www.adobe.com.

CRITERIA FOR COMPARISON

Educators who begin using one of the available numerical software packages tend to make comparisons with the use of a programming language for the solution of similar problems. The leap from a programming language to a particular software package usually leads to significant time-savings after the investment of effort in learning how to use the software package. Many educators become “addicted” to the one particular package that they first started using. They typically come to believe that the “selected” package is suitable for all of the needs of the undergraduate students, and that this “selected” package is probably not very different from other similar mathematical packages. Often the “selected” mathematical is chosen because of its use in graduate research. Unfortunately, this selection approach does not seem to lead to widespread acceptance of the numerical software packages by the engineering graduates (Davis et al.¹). In order to make their use more widespread, several actions should be taken. One of them is a comparison of the software packages based on objective criteria and the ease to which a package can allow significant improvements in the chemical engineering curriculum.

The following criteria are suggested for a mathematical software comparison:

1. Numerical Performance

1.1 Ability to solve all typical benchmark problems

2. User Friendliness

2.1 Menu versus command based program control

2.2 Notation and format used in equation entry

2.3 Debugging aids (syntax errors, undefined or not initialized variables, etc.)

2.4 Equation ordering and detection of implicit relationships

2.5 Solution verification

3. Technical effort required for:

3.1 Preparation of the model

3.2 Documentation of the model

- 3.3 Setting up the solution algorithm
- 3.4 Presentation of the results
- 3.5 Documentation of the results
- 3.6 Alteration of the model for parametric studies

Most of the above criteria are self explanatory. More detailed discussion of various criteria will be provided in the next section through reference to a particular example.

A COMPARISON OF THE PACKAGES

The general numerical performance of all of the packages was quite adequate. Some of the packages did require the latest professional version to accomplish all of the solutions. Significantly, there were major differences between the packages with regard to the other criteria. These differences will be demonstrated with reference to one of the benchmark problems.

Problem 9 deals with the “Reversible, Exothermic, Gas-Phase Reaction in a Catalytic Reactor,” which is an advanced level assignment in Chemical Reaction Engineering courses. The problem statement, the equations, and the numerical constants are given in Appendix A. The problem consists of three simultaneous ODE’s and several additional explicit algebraic equations. The initial values of all the variables are known.

The solutions via the different mathematical software packages will be presented in increasing level of complexity.

POLYMATH (Appendix B)

The POLYMATH solution given in Appendix B demonstrates that the differential equations (30), (37), and (39) of the Problem 9 needs only very minor modifications to be entered into the POLYMATH model input file. Similarly the algebraic equations (31) to (35) are entered with only minor modification. Note that POLYMATH automatically reorders the equations, placing equation (31) last to allow explicit expression of the algebraic variables. The POLYMATH entry screen also displays a list of undefined variable during problem entry which is extremely helpful in debugging the mathematical model.

POLYMATH is menu driven, and all the available options are presented on the screen. No technical information the integration method, stepsize, and error criteria needs to be provided by the user. The software has a default relative error tolerance value which halts the integration with an error message if a solution cannot be obtained with the desired accuracy.

The output shown in Appendix B indicates that the results can be displayed in graphical form of such quality that they can be directly included in student reports. The scaling and labeling is accomplished automatically but can be modified if desired.

The model is documented in a compact and clear form as shown in Appendix B.

The documentation includes all the equations, initial values of the dependent and independent variables, and final values of the independent variables. For parametric reruns, only the pertinent equations or constants must be altered.

MATHCAD (Appendix C)

The Mathcad solution is shown in Appendix C. The model definition requires more extensive changes to the original equation set that was required in POLY-MATH. Clear distinction must be made between variable calculated from differential and algebraic equations. The differential variables must be expressed as vectors and the differential equations as vectors of functions. The name of an “algebraic” variable must include the list of the “differential” variables that the “algebraic” variables utilize. The program is command based, and the user must specify the integration algorithm to be used and the number of data points to be stored for plotting.

The results are stored in a matrix, and the user must assign the matrix columns back to the variables in order to obtain a meaningful plot. Transformation of the variables and functions into a vector and transformation of the results back into the variables are unnecessary technical details that the user must handle using Mathcad.

MATLAB (Appendix D)

The Matlab solution is shown in Appendix D. The solution requires the user to write a program where Matlab provides the subprograms for numerical integration and plotting. Like Mathcad, the differential variables and functions must be transferred to vectors and the results assigned back to the variables. The user must select the integration method to be used. The setup for a plot of the output requires several lines of code. In order to write the Matlab program or to understand the model documentation, the user must be familiar with the Matlab syntax.

MATHEMATICA (Appendix E)

Appendix E shows the model as it is entered and solved using Mathematica. In this case “In[]” indicates that the input is provided by the user, and “Out[]” shows the program response. The user must select the integration algorithm. The integration and plotting involve subprogram calls with various parameters and complex syntax rules. Clearly the model preparation, setting up the solution algorithm, and presentation of the results require extensive technical effort and knowledge of Mathematica’s syntax rules. Even the documentation cannot be understood without some familiarity with these rules.

MAPLE (Appendix F)

The Maple solution is provided in Appendix F. The author of this solution has chosen to utilize his own routine for the numerical integration, but the solution could also possibly be obtained using Maple’s “dsolve” algorithm. The syntax for setting up the model for solution is fairly complex. The differential equations, the algebraic equations and the initial conditions must be collected into separate vectors which

are used consequently as arguments for the routine that carries out the numerical integration. The results of the integration are stored in a matrix, and it is the users responsibility to determine what variable is stored in a particular column. Labels for the axes of the plot must be provided explicitly. The disadvantage of this requirement is demonstrated in the writeup of the solution where two graphs appear without labels. The curve in the graph of T versus W is not a smooth curve, indicating that not enough data points have been stored to allow generation of a smooth curve. In summary, the technical effort required to set up, solve and document this example using Maple is rather extensive.

EXCEL (Appendix G)

Excel can be used to solve benchmark problem 9, but the solution involves extensive programming with Visual Basic for Applications which is part of Excel version 7.0. In order to solve this problem, a subprogram (public function) must be prepared for carrying out the numerical integration, and several additional subprograms are needed for calculating the derivative values. The programming involves extensive transfer of values between addresses in a vector of parameters and symbolic variable names.

The effort involved and the level of expertise required to solve this problem with Excel is much higher than required by any of the other packages. The documentation for the use of Visual Basic for Applications is difficult to find and hard to implement. Thus, the use of Excel to solve this particular problem involves a programming language, and the advantage that most mathematical software packages possess with built-in differential equation solution capability is lost.

CONCLUSIONS

All of the mathematical software packages were able to correctly solve the ten benchmark problems, but there were considerable differences in the “user friendliness” and the technical effort required to set up the model and document the model and the solution.

The detailed comparison of the various packages for problem 9 indicated that the POLYMATH package was superior to the others in terms of user friendliness and the amount of technical effort involved in the solution. The ranking of the other packages was as follows: 2) Mathcad, 3) Matlab, 4) Mathematica, 5) Maple and 6) Excel.

It is important to emphasize however, that this conclusion applies only to the type of problem which is represented by Problem 9. That is the simultaneous integration of several ODE's with known initial values and several explicit algebraic equations. Other types of problems alter the ranking significantly. For example, Excel has been found to be most appropriate for demonstrating single variable iterative processes and Matlab is excellent for demonstrating matrix operations and multivariable iterative processes.

Our work and experience with mathematical software has lead us to conclude that the most educational benefit can be gained by using several packages through-

out the curriculum. This is the same conclusion reached by Jones⁴ after he conducted a computer use survey. We suggest that the criteria developed in this work should be applicable in selection of a particular mathematical software package for a specific course. Several different mathematical software packages should be evaluated. Students are critical evaluators of educational software, and their input should be solicited. They will give strong preference to efficient software that will save them time and effort in solving the problem and in documentation of the solution. Once they become convinced of the value of a particular package, then they will continue to use it independently in their engineering coursework.

REFERENCES

1. Davis, J. F., G. E. Blau, and G. V. Reklaitis (1994). "Computers in Undergraduate Chemical Engineering Education: A Perspective on Training and Application", *Chem. Engr. Ed.* **28**.
2. Fogler, H. S. (1993). "An Appetizing Structure of Chemical Reaction Engineering for Undergraduates", *Chem. Engr. Ed.*, **27**(2), 110.
3. Fogler, H. S. (1992), **Elements of Chemical Reaction Engineering**, 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ.
4. Jones, J. B. (1998). "The Non-Use of Computers in Undergraduate Engineering Science Courses", *J. Engr. Ed.*, **87**(1), 11.
5. Shacham, M., and M. B. Cutlip (1982). "A Simulation Package for the PLATO Educational Computer System", *Comp. & Chem. Engr.*, **6**(3), 208.
6. Shacham, M., M. B. Cutlip and P. D. Babcock, "(1985). "A Microcomputer Simulation Package for Small Scale Systems", *Microprocessors and Microsystems*, **9**(2), 76-81.

APPENDICES*

APPENDIX A - The Problem 9 definition (attached to this paper) is from pp. 16-18 in document* Tenprobs.pdf authored by Michael B. Cutlip, Department of Chemical Engineering, Box U-222, University of Connecticut, Storrs, CT 06269-3222 (mcutlip@uconnvm.uconn.edu).

APPENDIX B - The POLYMATH solution is located on pp. PM-19 to PM-20 in document* Polymath.pdf and authored by Michael B. Cutlip, Department of Chemical Engineering, Box U-222, University of Connecticut, Storrs, CT 06269-3222 (mcutlip@uconnvm.uconn.edu) and Mordechai Shacham, Department of Chemical Engineering, Ben-Gurion University of the Negev, Beer Sheva, Israel 84105 (shacham@bgumail.bgu.ac.il).

APPENDIX C - The Mathcad solution is located on pp. MC-20 to MC-22 in document* Mathcad.pdf and authored by John J. Hwalek, Department of Chemical Engineering, University of Maine, Orono, ME 04469 (hwalek@maine.maine.edu).

* This complete paper, the entire problem set, technical write-ups for each mathematical software package, and problem solution files for each package are available via FTP from <ftp.engr.uconn.edu> in directory /pub/ASEE.

APPENDIX D - The Matlab solution is located on pp. ML-23 to ML-24 in document* Matlab.pdf and authored by Joseph Brule, John Widmann, Tae Han, and Bruce Finlayson, Department of Chemical Engineering, University of Washington, Seattle, WA 98195-1750 (finlayson@cheme.washington.edu).

APPENDIX E - The Mathematica solution is located on pp. M41 to M45 in document* Mathemat.pdf and authored by H. Eric Nuttall, Department of Chemical and Nuclear Engineering, University of New Mexico, Albuquerque, NM 87134-1341 (nuttall@unm.edu).

APPENDIX F - The Maple solution is located on pp. 38-43 in document* Maple.pdf and authored by Ross Taylor, Department of Chemical Engineering, Clarkson University, Potsdam, New York 13699-5705 (taylor@sun.soe.clarkson.edu).

APPENDIX G - The Excel solution is located on pp. 22-25 in document* Excel.pdf and authored by Edward M. Rosen, EMR Technology Group, 13022 Musket Ct., St. Louis, MO 63146 (EMRose@Compuserve.com).

APPENDIX A - PROBLEM 9

9. REVERSIBLE, EXOTHERMIC, GAS PHASE REACTION IN A CATALYTIC REACTOR

9.1 Numerical Methods

Simultaneous ordinary differential equations with known initial conditions.

9.2 Concepts Utilized

Design of a gas phase catalytic reactor with pressure drop for a first order reversible gas phase reaction.

9.3 Course Usage

Reaction Engineering

9.4 Problem Statement

The elementary gas phase reaction $2A \rightleftharpoons C$ is carried out in a packed bed reactor. There is a heat exchanger surrounding the reactor, and there is a pressure drop along the length of the reactor.

The various parameters values for this reactor design problem are summarized in Table (3).

* This complete paper, the entire problem set, technical write-ups for each mathematical software package, and problem solution files for each package are available via FTP from **ftp.engr.uconn.edu** in directory **/pub/ASEE**.

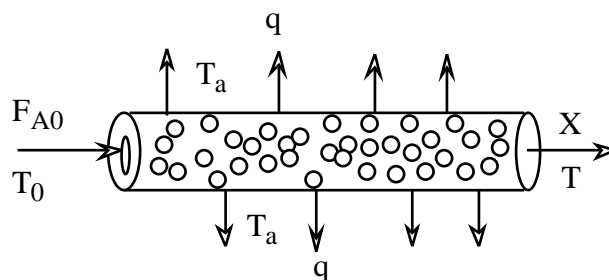


Figure 1 Packed Bed Catalytic Reactor

Table 3 Parameter Values for Problem 9.

$C_{PA} = 40.0 \text{ J/g}\cdot\text{mol}\cdot\text{K}$	$R = 8.314 \text{ J/g}\cdot\text{mol}\cdot\text{K}$
$C_{PC} = 80.0 \text{ J/g}\cdot\text{mol}\cdot\text{K}$	$F_{A0} = 5.0 \text{ g}\cdot\text{mol}/\text{min}$
$\Delta H_R = -40,000 \text{ J/g}\cdot\text{mol}$	$Ua = 0.8 \text{ J/kg}\cdot\text{min}\cdot\text{K}$
$E_A = 41,800 \text{ J/g}\cdot\text{mol}\cdot\text{K}$	$T_a = 500 \text{ K}$
$k = 0.5 \text{ dm}^6/\text{kg}\cdot\text{min}\cdot\text{mol} @ 450 \text{ K}$	$\alpha = 0.015 \text{ kg}^{-1}$
$K_C = 25,000 \text{ dm}^3/\text{g}\cdot\text{mol} @ 450 \text{ K}$	$P_0 = 10 \text{ atm}$
$C_{A0} = 0.271 \text{ g}\cdot\text{mol}/\text{dm}^3$	$y_{A0} = 1.0 \text{ (Pure A feed)}$
$T_0 = 450 \text{ K}$	

- Plot the conversion (X), reduced pressure (y) and temperature ($T \times 10^{-3}$) along the reactor from $W = 0 \text{ kg}$ up to $W = 20 \text{ kg}$.
- Around 16 kg of catalyst you will observe a “knee” in the conversion profile. Explain why this knee occurs and what parameters affect the knee.
- Plot the concentration profiles for reactant A and product C from $W = 0 \text{ kg}$ up to $W = 20 \text{ kg}$.

Addition Information

The notation used here and the following equations and relationships for this particular problem are adapted from the textbook by Fogler³. The problem is to be worked assuming plug flow with no radial gradients of concentrations and temperature at any location within the catalyst bed. The reactor design will use the conversion of A designated by X and the temperature T which are both functions of location within the catalyst bed specified by the catalyst weight W .

The general reactor design expression for a catalytic reaction in terms of conversion is a mole balance on reactant A given by

$$F_{A0} \frac{dX}{dW} = -r'_A \quad (30)$$

The simple catalytic reaction rate expression for this reversible reaction is

$$-r'_A = k \left[C_A^2 - \frac{C_C}{K_C} \right] \quad (31)$$

where the rate constant is based on reactant A and follows the Arrhenius expression

$$k = k(@ T=450^\circ K) \exp \frac{E_A}{R} \left[\frac{1}{450} - \frac{1}{T} \right] \quad (32)$$

and the equilibrium constant variation with temperature can be determined from van't Hoff's equation with $\Delta \tilde{C}_P = 0$

$$K_C = K_C(@ T=450^\circ K) \exp \frac{\Delta H_R}{R} \left[\frac{1}{450} - \frac{1}{T} \right] \quad (33)$$

The stoichiometry for $2A \rightleftharpoons C$ and the stoichiometric table for a gas allow the concentrations to be expressed as a function of conversion and temperature while allowing for volumetric changes due to decrease in moles during the reaction. Therefore

$$C_A = C_{A0} \left(\frac{1-X}{1+\varepsilon X} \right) \frac{P T_0}{P_0 T} = C_{A0} \left(\frac{1-X}{1-0.5X} \right) y \frac{T_0}{T} \quad (34)$$

and

$$y = \frac{P}{P_0} \quad (35)$$

$$C_C = \left(\frac{0.5 C_{A0} X}{1-0.5X} \right) y \frac{T_0}{T}$$

The pressure drop can be expressed as a differential equation (see Fogler³ for details)

$$\frac{d\left(\frac{P}{P_0}\right)}{dW} = \frac{-\alpha(1+\varepsilon X) P_0 T}{2 P T_0} \quad (36)$$

or

$$\frac{dy}{dW} = \frac{-\alpha(1-0.5X) T}{2y T_0} \quad (37)$$

The general energy balance may be written at

$$\frac{dT}{dW} = \frac{U_a(T_a - T) + r'_A(\Delta H_R)}{F_{A0}(\sum \theta_i C_{Pi} + X \Delta \tilde{C}_P)} \quad (38)$$

which for only reactant A in the reactor feed simplifies to

$$\frac{dT}{dW} = \frac{U_a(T_a - T) + r'_A(\Delta H_R)}{F_{A0}(C_{PA})} \quad (39)$$