Chemical Reaction Engineering (CRE) Education: From the Era of Slide Rule to the Digital Age

H. Scott Fogler, University of Michigan
Michael B. Cutlip, University of Connecticut

Abstract

The complexity level of CRE problems that students work and study in both undergraduate and graduate courses has increased very dramatically in the last 40 years. In the era of slides rules, the major focus was on reactors operated isothermally where charts, nomographs and linear differential equations were used to describe very simple situations. However, today's students are routinely solving complex problems involving multiple reactions with heat effects in CSTRs and PFRs that are either operated adiabatically or have co-current or countercurrent heat exchangers. The ability to solve such complex high-level problems has been enabled by user-friendly software packages, such as POLYMATH, that easily allow solution of systems of ODEs. Numerical solutions of CRE problems can easily be incorporated into example problems and special simulation where many “what if” scenarios can be examined very quickly. These provide for not only a greater understanding of CRE but also serve as a vehicle for students to develop and practice their creative thinking skills.

The application of CRE fundamentals has been extended to many new and emerging areas such as biological systems (fermentation and enzymatic reactors), electrochemical systems (fuel cells), combustion processes, pharmacokinetics, and environmental engineering, to mention only a few. As new technologies emerge, it is clear that the fundamentals of CRE will continue to be applied and will make enabling and substantial contributions.

1. Typical Course Content in CRE

Chemical Reaction Engineering (CRE) is a core course in most every chemical engineering department in the U.S. and perhaps the world. CRE along with Separation Processes are the two primary courses that help to distinguish chemical engineering from other engineering disciplines. The evolution of CRE to its present level of importance and popularity started in the 1940s when CRE was barely mentioned in any chemical engineering courses.

In the early days, CRE was known and referred to as chemical kinetics and catalysis. As the number of different chemical products steadily increased, so did the production rates of these chemicals. It became apparent that the equipment and processing that worked for making one chemical did not come close to meeting the demands of making a different chemical. Everyone knew they could not just use the same type reactor feed, and temperature for a different reaction but very few chemical engineers had the CRE knowledge to scale the reactions. Pilot plants were used to determine the processing conditions for the full-scale production of the desired product. The experiments carried out on the pilot plants to find the right temperatures, processing times were costly and time consuming, and the need for the graduating chemical engineer to understand CRE principles became apparent.

Let's look at where chemical reaction engineering is now and how this was accomplished. Most universities have 15-week terms so consider the distribution of content material by the number of weeks devoted to a specific topic. Table 1 shows a typical distribution of the subject material at the University of Michigan. Discussions with colleagues reveal it is not too much different at other universities.

Table 1 - Typical Distribution of Material in an Undergraduate CRE Course.

1.	The Balance Equations	1 week
2.	Rate Laws, Activation Energy, and Stoichiometry	1 week
3.	Batch, Plug Flow, and CSTR Reactor Design	2 weeks
4.	Membrane and Semibatch Reactor Design	1 week
5.	Analysis of Rate Data	½ week
6.	Multiple Reactions in Batch, Plug Flow and CSTR Reactor Design	1 ½ weeks
7.	Reaction Kinetics and Mechanisms	½ week
8.	Enzyme Kinetics and Bio Reactor Designs	1 ½ weeks
9.	Heat Effects for Single Reactions, Multiple Steady States	2 ½ weeks
10.	Multiple Reactions with Heat Effects	½ week
11.	Gas-Solid Catalysis Laws, Mechanisms, and Decay	3 weeks
12.	Reactor for Catalyst Decay, STTR, Moving Bed	1 weeks
		16 weeks

There are various perturbations at different universities to the subjects and time allocations shown in Table 1. The most common topics that replace two or three weeks of material in Table 1 are shown in Table 2.

Table 2 - Variations of Topics Covered in CRE

A.	Diffusion and Reaction in Porous Catalyst	1 week
B.	Reactors for Catalyst Poisoning	1 week
C.	Mixing and Residence Time Distribution, Modeling, Non Ideal Reactors	2 weeks
D.	Chain and Polymer Reactions	2 weeks

For example, Topic 7 in Table 1 (Enzymes) would be replaced by Topic B in Table 2 (Poisoning).

A review of the material that was available in textbooks over 60 years ago surprisingly indicated that most of the subject matter in Tables 1 and 2 was discussed in volume three of Hougen, Watson and Ragaatz, *Chemical Process Principles*. The topics in Tables 1 and 2 that were missing or not covered in sufficient detail for an undergraduate course are shown in Table 3.

Table 3 - Topics Missing in the 1940s

1. Derivation and Similarities of the Balance Equation
2. Grater Understanding of Activation Energy
3. Continuous Stirred Tank Reactors (CSTRs)
4. Semibatch Reactors
5. Membrane Reactors
6. Moving Bed Reactors
7. Straight Through Transport Reactors
8. Algorithm for Handling Complex Reactions
9. Rigorous Applications of the Energy Balance
10. Multiple Steady States
11. Enzyme Kinetics and Bio Reactors
12. Multiple Reactions with Heat Effects
13. Modeling Non-Ideal Reactors
14. Gas Phase Reactions with Change in Total Number of Moles and Pressure Drop

2. Calculations in CRE

A. In the 1940s - Numerical Analysis and Slide Rules

The material was taught through involved example problems, many of which did not show how the end results were achieved. There were many times where gross approximations were made to facilitate slide rule calculations which used an Euler type integration scheme for differential equations,

$$X_{i+1} = X_i + f(X_1)\Delta Z$$

or perhaps a predictor corrector method. These calculations were all carried out with slide rules. It would take days to calculate temperature profiles for two or three different setting of the parameter or operating variables.

B. The 1950s

In the 1950s, many solutions to reaction engineering homework problems usually involved analytical solutions. However, a small number of universities assigned problems that could take advantage of mechanical type business calculators.

At the close of the 1950s, the tools for solving CRE problems are shown in Figure 1. The slide rule was the main calculational tool. The pencil and paper were used to record the calculations. When physical properties were needed, they were often obtained from nomographs. Rectangular, semi-log and log-log graph papers were used for plotting, and “French” curves used to “regress data” to smooth curves.

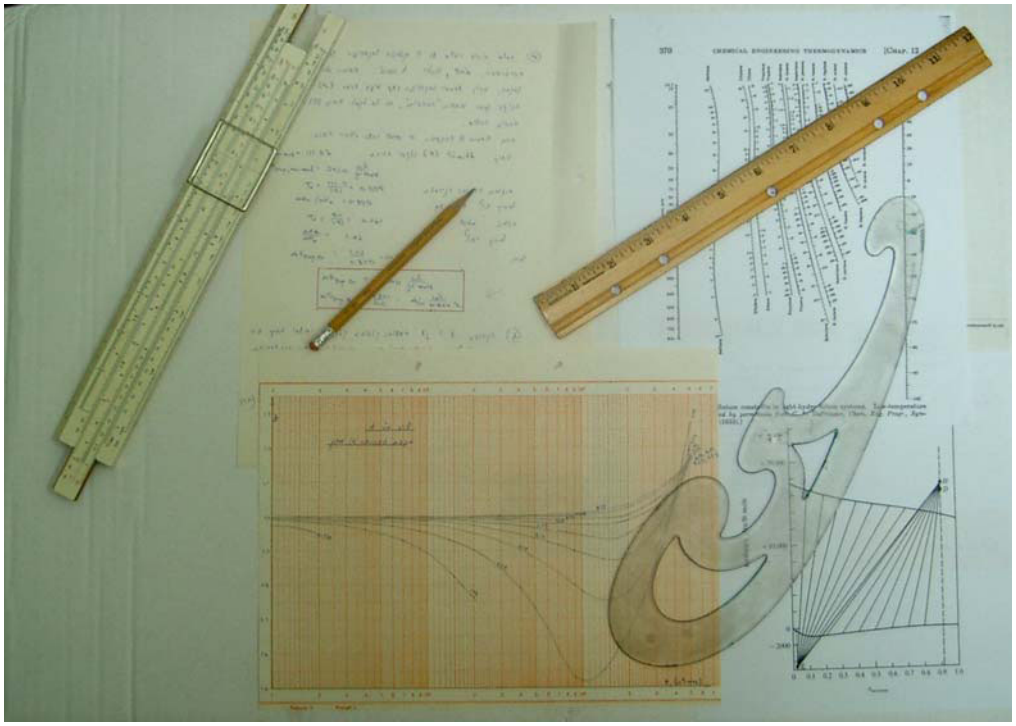


Figure 1 – CRE Student Tools in 1950s and Early 1960s

C. The 1960s

Analog and Hybrid Computers

The hybrid computers of this era used electronic circuits to perform calculations and were particularly useful for differential equations. The entire problem needed to be formulated into an electronic circuit using power supplies, function generators, initial condition power supplies, etc. The output was given to an oscilloscope or strip chart recorder. Figure 2 shows a Heath electronic analog computer kit which was used by some academic departments.

Figure 2 - Heath Electronic Analog Computer ~ 1960

Solutions of the problems involving series reactions, for example, required students to first wire up the problem and adjust the needed amplifiers and function generators to electronically simulate the problem and then "scale" the results back to the concentration variables of the reaction system. For example, in some exams or homework assignments, wiring sheets were used and the students were asked to show how to connect the wires to solve the a problem in a tubular flow reactor for reaction systems such as

$$A \rightarrow B \rightarrow C$$

Hybrid computers combined the analog computer with a small digital computer so that the analog problems could be under the control of a rudimentary digital computer. Some solid state electronics allows more accurate amplifiers so that more accurate results were obtained. All results were scaled to either ± 10 or ± 100 volts dc.

Digital Computers

Digital computers that were used in CRE at many universities started with the IBM 1401 and the much larger 360 series which followed. The programming language was usually FORTRAN that required programming for numerical solutions. Considerable effort was required to create a working program and make it adequate at solving CRE type problems. Users in the early 60s will remember when one run a day was the norm for developing and using FORTRAN programs.

Punched Cards

In the mid 60s, computer input was made with punched cards with some 72 columns of program information. Columns 72-80 were used to number the order of the cards so programs could be reassembled when the decks of cards were accidentally dropped. User of this era will remember that card punch machines that were used for hours to type up or correct the programs. Undergraduate CRE students of this period were using FORTRAN and other languages (such as MAD – Michigan Algorithmic Decoder) to write programs to solve reaction engineering problems with heat effects using a Runga-Kutta-Gill numerical scheme for differential equations. Hours and hours were spent as the students developed and keypunched his/her program and then waited in line for the output (if the program worked) to get the printout and his/her punched cards back. Most of the hours went into debugging the program thereby leaving little time to ***explore*** the problem. With the exception of the term projects, home problems that required only analytical solutions were assigned.

D. The 1970s

Punch cards continued to be used until the mid 1970s when FORTRAN programs could be written from computer work stations. Slide rules use disappeared completely as electron pocket calculators came into use in the mid 1970s for manual calculations. The first electronic calculator, the HP-35, is shown in Figure 3. This revolutionary calculator and others that followed allowed students to make desktop calculations such as log, ln, exponentials, trig functions, etc., and this allowed students and professionals to solve many problems much more rapidly.

Figure 3 – The HP-35 Electronic Calculator

Some CRE courses began using the IBM CSMP software (Continuous Systems Modeling Program) in the early 70s. This was a higher-level programming language that was suitable for integrating systems of ODEs on IBM System 360 mainframes. It also provided for easy control of initial conditions, repetitive integrations, inclusion of data from tables, and other convenient features. Output was either tables or very simple plots created from a standard alpha-numeric printer.

E. The 1980s

Computer programming, mainly with FORTRAN on large mainframe computers made by IBM, Control Data Corporation and others, continued to be used through the 1980s into the 1990s to solve complex CRE problems. Additionally, smaller computers such as the Digital Equipment Corporation (DEC) PDP series, were providing FORTRAN and other languages for making CRE calculations.

In 1981, IBM introduced the personal computer that was responsible for a dramatic increase in the availability of making CRE calculations on the desktop of faculty and students. Apple Corporation also pioneered the use of small personal computers. Again, FORTRAN compilers were initially used for programming of CRE problem solutions. Programs such as MATLAB[1] began to emerge that allowed for solution of numerical problems with increasingly less effort on the part of the user.

F. The 1990s

The next big advancement or paradigm shift in teaching undergraduate chemical reaction engineering came in the early 1990s with the introduction of inexpensive site licenses for POLYMATH[2] software by the CACHE Corporation[3] (Computer Aids for Chemical Engineering). This software was equation-orientated and provided interactive capabilities for the numerical solutions of CRE problems summarized in Table 4. Perhaps the most important was the ease in which systems of ODEs could be solved. Previous to POLYMATH, the students would spend 12-16 hours developing and testing FORTRAN codes to solve problems with heat effects and nonlinear reaction kinetics. After spending the time to defray the code, little time was left to explore the different parameters and operating conditions. With POLYMATH, one simply typed the ordinary differential equations (ODEs) as they were written on the paper, and they are solved immediately. Usually the programming took 15 to 30 minutes, after which the student was more than sufficient time to explore the problem and ask "What if…" questions. With POLYMATH it was also easy to address more complex problems such as multiple reactions with heat effects where some reactions are exothermic and some endothermic such as the production of styrene from ethyl benzene. Other situations that can be explored with POLYMATH in CRE are co-current and counter current heat exchange, membrane reactors, and bioreactors to mention a few.

Table 4 - Topics Easily Solved with POLYMATH

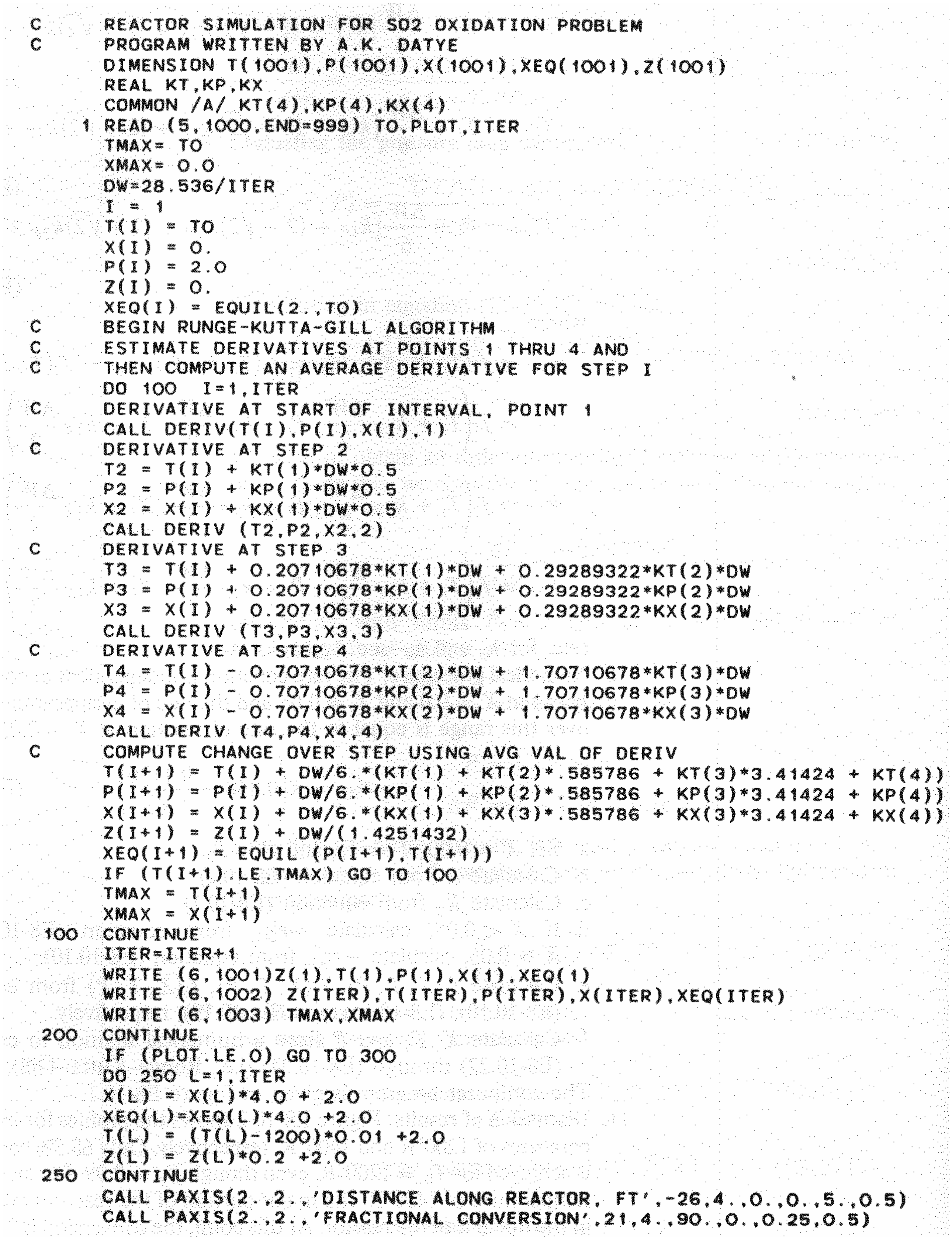
Multiple Reactions
Co and Counter Current Heat Exchanger
Membrane Reactors
Bioreactors
Multiple Reactions with Heat Effects
Nonlinear Regression of Rate
Solutions to Nonlinear Algebraic Balances on CSTRs

G. The 2000s until Today

Students and faculty in the current decade are routinely using interactive software, such as POLYMATH, Mathcad[4], and MATLAB, on their own personal desktop or notebook computers to solve problems in CRE that were not ever considered in the early years of this field of study. This allows faculty teaching CRE to focus on the development of CRE principles in many major new areas while the generation of the numerical solutions of the resulting mathematical models has become several orders of magnitude less than in the days of FORTRAN programming on mainframe computers. The important educational benefit is that the theory of CRE can continue to be developed while the numerical solution of the resulting mathematical problems becomes much less of a challenge.

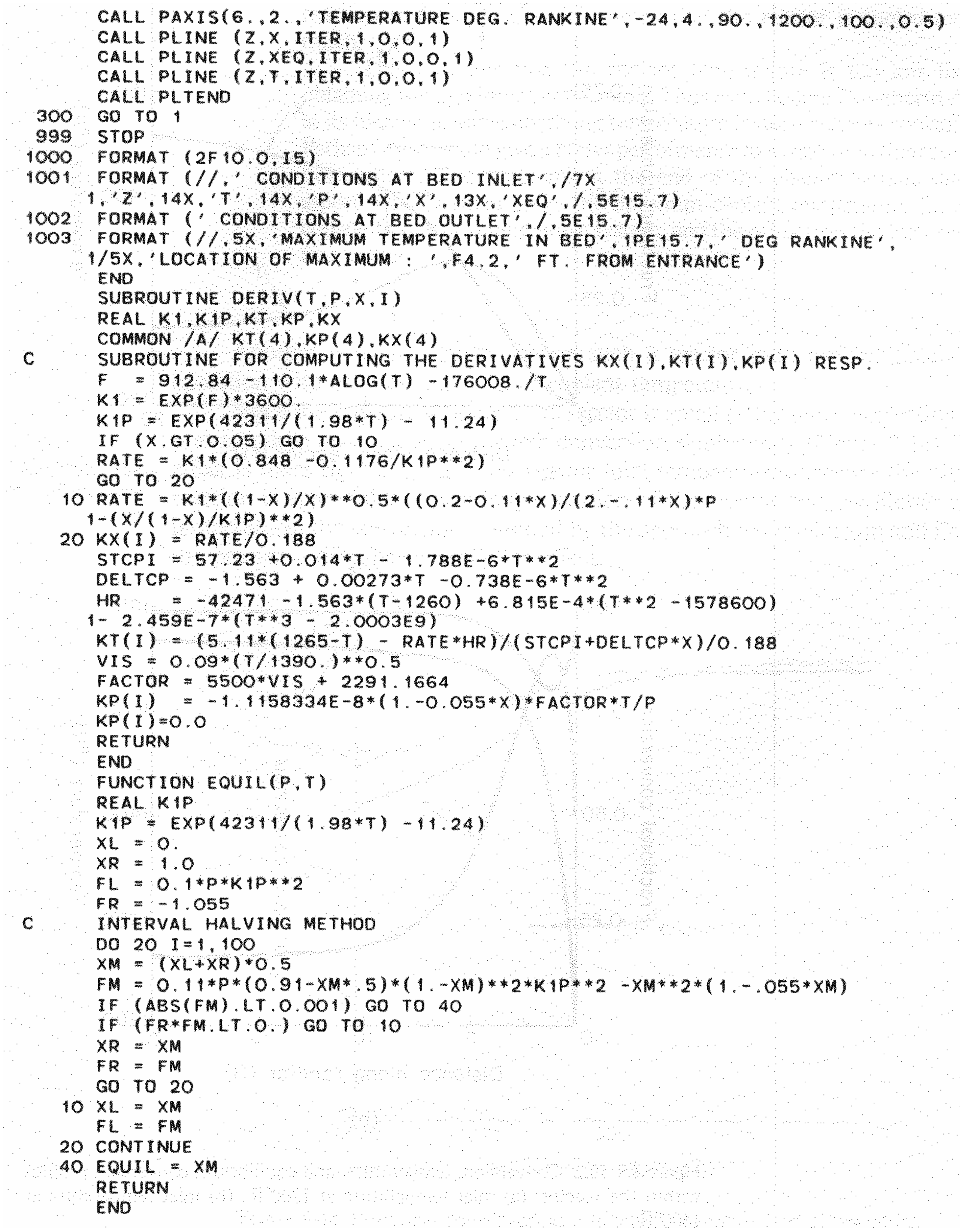
It is interesting to look back briefly some 20 years into the "then and now" solutions of a CRE problem that was first handled by a FORTRAN program and compare the levels of effort required to achieve a numerical solution. Consider the modeling of a non-adiabatic tubular flow reactor for the catalytic oxidation of SO_2 first discussed by Fogler in the second edition published in 1986. A partial listing of the FORTRAN program is reproduced in Table 5. Note that a number of essential subroutines necessary for this program and for output generation are not shown. This program is very complex and has many opportunities for errors to be made before an accurate solution is finally obtained.

Table 5 – Partial FORTRAN Program for SO_2 Oxidation in a Non-adiabatic Tubular Flow Reactor



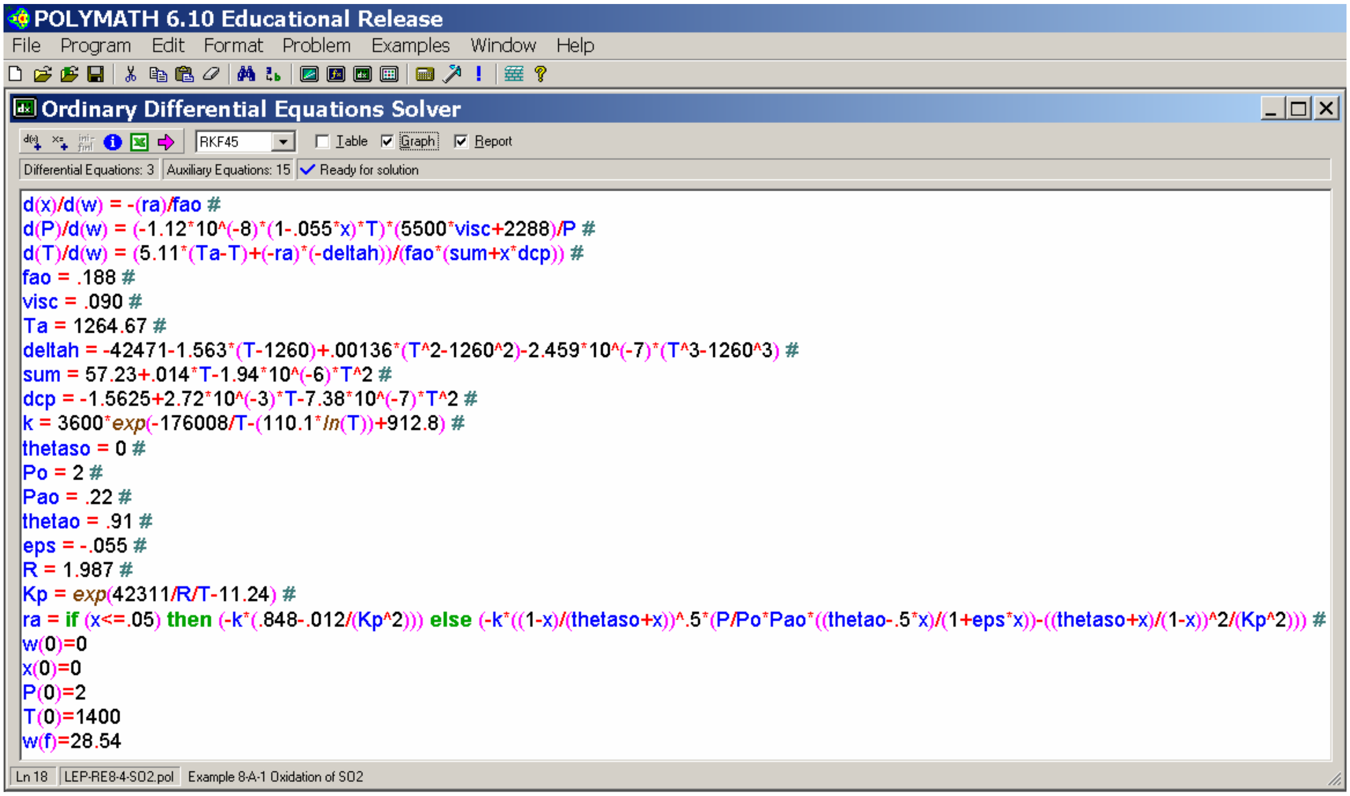
```
C     REACTOR SIMULATION FOR SO2 OXIDATION PROBLEM
C     PROGRAM WRITTEN BY A.K. DATYE
      DIMENSION T(1001),P(1001),X(1001),XEQ(1001),Z(1001)
      REAL KT,KP,KX
      COMMON /A/ KT(4),KP(4),KX(4)
    1 READ (5,1000,END=999) TO,PLOT,ITER
      TMAX= TO
      XMAX= 0.0
      DW=28.536/ITER
      I = 1
      T(I) = TO
      X(I) = 0.
      P(I) = 2.0
      Z(I) = 0.
      XEQ(I) = EQUIL(2.,TO)
C     BEGIN RUNGE-KUTTA-GILL ALGORITHM
C     ESTIMATE DERIVATIVES AT POINTS 1 THRU 4 AND
C     THEN COMPUTE AN AVERAGE DERIVATIVE FOR STEP I
      DO 100  I=1,ITER
C     DERIVATIVE AT START OF INTERVAL, POINT 1
      CALL DERIV(T(I),P(I),X(I),1)
C     DERIVATIVE AT STEP 2
      T2 = T(I) + KT(1)*DW*0.5
      P2 = P(I) + KP(1)*DW*0.5
      X2 = X(I) + KX(1)*DW*0.5
      CALL DERIV (T2,P2,X2,2)
C     DERIVATIVE AT STEP 3
      T3 = T(I) + 0.20710678*KT(1)*DW + 0.29289322*KT(2)*DW
      P3 = P(I) + 0.20710678*KP(1)*DW + 0.29289322*KP(2)*DW
      X3 = X(I) + 0.20710678*KX(1)*DW + 0.29289322*KX(2)*DW
      CALL DERIV (T3,P3,X3,3)
C     DERIVATIVE AT STEP 4
      T4 = T(I) - 0.70710678*KT(2)*DW + 1.70710678*KT(3)*DW
      P4 = P(I) - 0.70710678*KP(2)*DW + 1.70710678*KP(3)*DW
      X4 = X(I) - 0.70710678*KX(2)*DW + 1.70710678*KX(3)*DW
      CALL DERIV (T4,P4,X4,4)
C     COMPUTE CHANGE OVER STEP USING AVG VAL OF DERIV
      T(I+1) = T(I) + DW/6.*(KT(1) + KT(2)*.585786 + KT(3)*3.41424 + KT(4))
      P(I+1) = P(I) + DW/6.*(KP(1) + KP(2)*.585786 + KP(3)*3.41424 + KP(4))
      X(I+1) = X(I) + DW/6.*(KX(1) + KX(3)*.585786 + KX(3)*3.41424 + KX(4))
      Z(I+1) = Z(I) + DW/(1.4251432)
      XEQ(I+1) = EQUIL (P(I+1),T(I+1))
      IF (T(I+1).LE.TMAX) GO TO 100
      TMAX = T(I+1)
      XMAX = X(I+1)
  100 CONTINUE
      ITER=ITER+1
      WRITE (6,1001)Z(1),T(1),P(1),X(1),XEQ(1)
      WRITE (6,1002) Z(ITER),T(ITER),P(ITER),X(ITER),XEQ(ITER)
      WRITE (6,1003) TMAX,XMAX
  200 CONTINUE
      IF (PLOT.LE.0) GO TO 300
      DO 250 L=1,ITER
      X(L) = X(L)*4.0 + 2.0
      XEQ(L)=XEQ(L)*4.0 +2.0
      T(L) = (T(L)-1200)*0.01 +2.0
      Z(L) = Z(L)*0.2 +2.0
  250 CONTINUE
      CALL PAXIS(2.,2.,'DISTANCE ALONG REACTOR, FT',-26,4.,0.,0.,5.,0.5)
      CALL PAXIS(2.,2.,'FRACTIONAL CONVERSION',21,4.,90.,0.,0.25,0.5)
```

Table 5 (Cont.) – Partial FORTRAN Program for SO_2 Oxidation in a Non-adiabatic Tubular Flow Reactor



```
      CALL PAXIS(6.,2.,'TEMPERATURE DEG. RANKINE',-24,4.,90.,1200.,100.,0.5)
      CALL PLINE (Z,X,ITER,1,0,0,1)
      CALL PLINE (Z,XEQ,ITER,1,0,0,1)
      CALL PLINE (Z,T,ITER,1,0,0,1)
      CALL PLTEND
  300 GO TO 1
  999 STOP
 1000 FORMAT (2F10.0,I5)
 1001 FORMAT (//,' CONDITIONS AT BED INLET',/7X
     1,'Z',14X,'T',14X,'P',14X,'X',13X,'XEQ',/,5E15.7)
 1002 FORMAT (' CONDITIONS AT BED OUTLET',/,5E15.7)
 1003 FORMAT (//,5X,'MAXIMUM TEMPERATURE IN BED',1PE15.7,' DEG RANKINE',
     1/5X,'LOCATION OF MAXIMUM : ',F4.2,' FT. FROM ENTRANCE')
      END
      SUBROUTINE DERIV(T,P,X,I)
      REAL K1,K1P,KT,KP,KX
      COMMON /A/ KT(4),KP(4),KX(4)
C     SUBROUTINE FOR COMPUTING THE DERIVATIVES KX(I),KT(I),KP(I) RESP.
      F   = 912.84 -110.1*ALOG(T) -176008./T
      K1  = EXP(F)*3600.
      K1P = EXP(42311/(1.98*T) - 11.24)
      IF (X.GT.0.05) GO TO 10
      RATE = K1*(0.848 -0.1176/K1P**2)
      GO TO 20
   10 RATE = K1*((1-X)/X)**0.5*((0.2-0.11*X)/(2.-.11*X)*P
     1-(X/(1-X)/K1P)**2)
   20 KX(I) = RATE/0.188
      STCPI  = 57.23 +0.014*T - 1.788E-6*T**2
      DELTCP = -1.563 + 0.00273*T -0.738E-6*T**2
      HR     = -42471 -1.563*(T-1260) +6.815E-4*(T**2 -1578600)
     1- 2.459E-7*(T**3 - 2.0003E9)
      KT(I) = (5.11*(1265-T) - RATE*HR)/(STCPI+DELTCP*X)/0.188
      VIS = 0.09*(T/1390.)**0.5
      FACTOR = 5500*VIS + 2291.1664
      KP(I)  = -1.1158334E-8*(1.-0.055*X)*FACTOR*T/P
      KP(I)=0.0
      RETURN
      END
      FUNCTION EQUIL(P,T)
      REAL K1P
      K1P = EXP(42311/(1.98*T) -11.24)
      XL = 0.
      XR = 1.0
      FL = 0.1*P*K1P**2
      FR = -1.055
C     INTERVAL HALVING METHOD
      DO 20 I=1,100
      XM = (XL+XR)*0.5
      FM = 0.11*P*(0.91-XM*.5)*(1.-XM)**2*K1P**2 -XM**2*(1.-.055*XM)
      IF (ABS(FM).LT.0.001) GO TO 40
      IF (FR*FM.LT.0.) GO TO 10
      XR = XM
      FR = FM
      GO TO 20
   10 XL = XM
      FL = FM
   20 CONTINUE
   40 EQUIL = XM
      RETURN
      END
```

The complexity of the partial FORTRAN program of Table 5 can be compared with the "Now" version of a complete POLYMATH program for this same problem as shown in Figure 4. Today's user can employ the interactive POLYMATH editor on a personal computer to easily enter the differential and algebraic equations. Syntax is automatically checked during entry, and the equations can be entered in any order as they are automatically ordered before solution. "If … then … else" logic can be utilized, for example, as is employed in the calculation of the reaction rate, **ra**, in Table 5. Robust integration algorithms with automatic step size adjustment and low error tolerances can be selected prior to problem solution. The user can select some combination of a Report, Graph, or Table to provide the results. For example, a graphical plot of the reactor temperature profile from POLYMATH is shown in Figure 5.



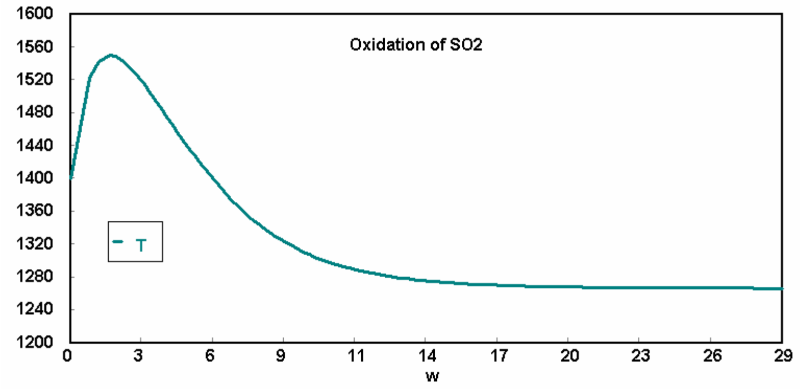
RKF45 ☐ Table ☑ Graph ☑ Report

Differential Equations: 3 | Auxiliary Equations: 15 | ✓ Ready for solution

```
d(x)/d(w) = -(ra)/fao #
d(P)/d(w) = (-1.12*10^(-8)*(1-.055*x)*T)*(5500*visc+2288)/P #
d(T)/d(w) = (5.11*(Ta-T)+(-ra)*(-deltah))/(fao*(sum+x*dcp)) #
fao = .188 #
visc = .090 #
Ta = 1264.67 #
deltah = -42471-1.563*(T-1260)+.00136*(T^2-1260^2)-2.459*10^(-7)*(T^3-1260^3) #
sum = 57.23+.014*T-1.94*10^(-6)*T^2 #
dcp = -1.5625+2.72*10^(-3)*T-7.38*10^(-7)*T^2 #
k = 3600*exp(-176008/T-(110.1*ln(T))+912.8) #
thetaso = 0 #
Po = 2 #
Pao = .22 #
thetao = .91 #
eps = -.055 #
R = 1.987 #
Kp = exp(42311/R/T-11.24) #
ra = if (x<=.05) then (-k*(.848-.012/(Kp^2))) else (-k*((1-x)/(thetaso+x))^.5*(P/Po*Pao*((thetao-.5*x)/(1+eps*x))-((thetaso+x)/(1-x))^2/(Kp^2))) #
w(0)=0
x(0)=0
P(0)=2
T(0)=1400
w(f)=28.54
```

Ln 18 | LEP-RE8-4-SO2.pol | Example 8-A-1 Oxidation of SO2

Figure 4 –POLYMATH Program for SO_2 Oxidation in a Non-adiabatic Tubular Flow Reactor



Note that POLYMATH can also automatically generate an Excel[5] spreadsheet solution for this problem and can also provide ordered MATLAB code for an m-file solution to this problem.

Figure 5 – Graph from POLYMATH Program for SO_2 Oxidation in a Non-adiabatic Tubular Flow Reactor

H. The Future

As CRE continues to develop, numerical solutions to partial differential equations that are used to model every more complex reactors and reacting systems will be needed. In this regard, software such and MATLAB and COMSOL[6] (formerly FEMLAB), for example, have developed software programs that can deal with PDEs. The current status is such that general use is coming but at present students can typically only change various reaction parameters in available solutions. Unfortunately, current students typically cannot easily change the model or system with the "canned" programs. However, if students learn how to program with software such as COMSOL, they can study any reaction in a tubular reactor. In the near future, it will probably be as easy to deal with PDEs as is now available for ODEs.

CRE appears to have a bright future in chemical engineering and related fields as it is well grounded in fundamentals that can be applied into the many developing areas of emerging and new technologies. It seems that the necessary computing and important software will continue to be developed that will allow students, faculty, and practicing engineers to solve ever more complex CRE related problems.

References

1. MATLAB is a registered trademark and software product of The Mathworks (http://www.mathworks.com/)
2. POLYMATH is a product of Polymath Software (http://www.polymath-software.com/)
3. CACHE Corporation (http://www.cache.org/)
4. Mathcad is a registered trademark and software product of the PTC Corporation (http://www.mathsoft.com/)
5. Excel is a registered trademark and software product of the Microsoft Corporation (http://www.microsoft.com/)
6. COMSOL is a registered trademark and software product of COMSOL, Inc. (http://www.comsol.com/)

Bibliography

Aris, R., and N.R. Amundson, *Mathematical Methods in Chemical Engineering Volume 2 First-Order Partial Differential Equations with Applications,* Prentice-Hall Inc., Englewood Cliffs, NJ 1973.

Aris, R., *Elementary Chemical Reactor Analysis*, Prentice-Hall Inc., Englewood Cliffs, NJ 1969.

Aris, R., *Introduction to the Analysis of Chemical Reactors*, Prentice-Hall Inc., Englewood Cliffs, NJ 1965.

Boudart, M., *Kinetics of Chemical Processes,* Butterworth-Heinemann Reprint Series 1991, originally published by Prentice-Hall in 1968.

Butt, J.B., *Reaction Kinetics and Reactor Design,* Prentice-Hall Inc., Englewood Cliffs, NJ 1980.

Butt, J.B., *Reaction Kinetics and Reactor Design, Second Edition, Revised and Expanded,* Marcel Dekker, Inc., New York 2000.

Carberry, J.J., *Chemical and Catalytic Reaction Engineering,* McGraw Hill, New York NY 1976.

Cooper, A. R. and G. V. Jeffreys, *Chemical Kinetics and Reactor Design*, Prentice-Hall Inc., Englewood Cliffs, NJ 1971.

Davis, M.E. and R. J. Davis, *Fundamentals of Chemical Reaction Engineering*, McGraw Hill, New York, NY 2003.

Denbigh, K.G., *Chemical Reactor Theory, An Introduction,* Cambridge University Press, 1965.

Fogler, H. F., *Elements of Chemical Reaction Engineering, Fourth Edition*, Prentice-Hall Inc., Upper Saddle River, NJ 2005.

Froment, G. F. and K. B. Bischoff, *Chemical Reactor Analysis and Design, Second Edition,* John Wiley & Sons, New York 1979, 1990.

Gray, P. and S. K. Scott, *Chemical Oscillations and Instabilities Non-linear Chemical Kinetics*, Clarendon Press, Oxford 1990.

Harriott, P., *Chemical Reactor Design,* Marcel Dekker, Inc., New York 2003.

Hayes, R.E., *Introduction to Chemical Reactor Analysis,* Gordon and Breach Science Publishers, 2001.

Hill Jr., Charles G., *Introduction to Chemical Engineering Kinetics & Reactor Design*, John Wiley & Sons, September 1977.

Holland, C.D. and R. G. Anthony, *Fundamentals of Chemical Reaction Engineering, Second Edition,* Prentice-Hall Inc., Englewood Cliffs, NJ 1989.

Hougen, O.A., and K.M. Watson, *Chemical Process Principles, Part Three: Kinetics and Catalysis*, John Wiley & Sons, New York 1964.

Levenspiel, O., *Chemical Reaction Engineering, Third Edition*, John Wiley & Sons, New York 1999.

Levenspiel, O., *Chemical Reactor Omnibook*, Oregon State University Book Stores, Inc. Corvallis, Oregon 1989.

Nauman. E.B., *Chemical Reactor Design,* John Wiley & Sons, New York 1987.

Nauman. E.B., *Chemical Reactor Design, Optimization, and Scaleup,* McGraw Hill, New York NY 2001.

Petersen, E. E., *Chemical Reaction Analysis*, Prentice-Hall Inc., Englewood Cliffs, NJ 1965.

Rase, H.F., *Chemical Reactor Design for Process Plants, Volume One: Principles and Techniques,* John Wiley & Sons, New York 1977.

Rase, H.F., *Chemical Reactor Design for Process Plants, Volume Two: Case Studies and Design Data,* John Wiley & Sons, New York 1977.

Rawlings, B. and J. G. Ekerdt, *Chemical Reactor Analysis and Design Fundamentals*, Nob Hill Publishing, Madison WI, 2002.

Riet, K. V. and J. Tramper, *Basic Bioreactor Design,* Marcel Dekker, Inc., New York 1991.

Smith, J. M., *Chemical Engineering Kinetics*, McGraw Hill, New York NY 1981.

Walas, S. M., *Reaction Kinetics for Chemical Engineers*, McGraw Hill, New York NY 1959.