INTERACTIVE COMPUTER-AIDED INSTRUCTION

Susan Montgomery and H. Scott Fogler
Department of Chemical Engineering
University of Michigan
Ann Arbor, MI 48109

Abstract

This paper first describes how interactive computing can address the needs of both traditional and non-traditional learners through Soloman's Inventory of Learning Styles and also focuses on the higher level skills in Bloom's taxonomy. Next, the types of interactive computing are described and classified: Presentation, Assessment, Exploration, and Simulation. After a brief historical overview, CACHE's participation in a number of interactive computing projects and the resulting products are presented. The CACHE products described are POLYMATH, PICLES, the Michigan Modules, Purdue-Industry Computer Simulations and the Washington Chemical Reactor Design Tool. The paper closes with a discussion of applications using multimedia, hypertext, and virtual reality.

Introduction

Students learn a course's content best when exposed to the subject matter using a variety of teaching styles. A majority of undergraduate students learn best through experimentation and active involvement in the subject matter (Felder and Silverman, 1988). The standard textbook-lecture-homework triad allows for little of this type of learning. Consequently, it is necessary to enhance the curriculum by supplementing the standard teaching methods to increase the number of learning modes available to the student. This enhancement can be achieved through interactive computing that can provide students with supplementary exposure to the fundamental concepts in chemical engineering, as well as to give them an opportunity to apply and further explore these concepts. We will start this chapter by addressing the pedagogical needs and learning styles of engineering students, and discussing the types of interactive computer instruction that can address these needs and learning styles. We then provide a historical overview of the development of interactive computer materials in chemical engineering education, and a look to the future of interactive computing.

Focus on Learning Styles

Engineering education often focuses on disseminating technical information, without helping students make the connections between this information and both their own life experiences and the processes they will encounter in their careers (Griskey, 1991). However, extensive research in cognitive psychology has shown that effective learning of new principles requires explicit presentation of situations where these abstract principles are relevant (Vander-Stoep and Seifert, 1993).

Traditional teaching also ignores the needs of non-traditional learners and often results in students perceiving the material covered in these courses as foreign to any experience they may have had or will have as engineers. The need to address non-traditional learning styles is particularly important in efforts to attract and retain women and underrepresented minorities, who typically do not conform to traditional learning styles. Multimedia and interactive computing materials address these needs, by allowing students to interact with information using a variety of active mechanisms, rather than being passively exposed to information without goals (Qasem and Mohamadian, 1992). Kulik and Kulik (1986, 1987) reported that most studies found that computer-based instruction had positive effects on students. Specifically, students:

- learn more.
- learn faster (the average reduction in instructional time in 23 studies was 32%).
- like classes more when they receive computer help.
- develop more positive attitudes toward computers when they receive help from them in school.

One of the key factors to successfully developing and using interactive computing in courses is to first identify the activities that cannot be accomplished by other means (e.g., pencil and paper, calculator). By activities, we mean those exercises that are used to practice certain skills, learn new material, or test comprehension of previously learned material. In deciding which activities to include in educational software, one can take advantage of Bloom's Taxonomy of Educational Objectives (Bloom, 1956), shown in Table 1,which classifies the intellectual skill levels of various activities. A typical undergraduate course usually focuses on the first three levels only. Computer-based materials are one way to allow students to exercise their higher level thinking skills.

Once one has identified the skills one wants the user to practice, one needs to determine how to best reach the student to ensure that these skills are indeed exercised. Myers-Briggs Type Indicators have been widely used to classify student learning styles (McCaulley et al., 1983, Felder et al., 1993a). They can be used to classify people's personalities according to four dimensions: Extroversion vs. Introversion, Sensing vs. Intuition, Thinking vs. Feeling and Judging vs. Perceptive. This test's focus on personality as a whole is too broad for an identification of learning styles, however. An assessment tool that combines the simplicity of the Myers Briggs inventory with an emphasis on teaching is the Inventory of Learning Styles, by Barbara Soloman (1992). After answering 28 simple questions, a student is classified along four learning-style dimensions, shown in Table 2.

In combination, Bloom's Taxonomy and Soloman's Inventory of Learning Styles provide a means of determining both the content of the computer package and the presentation to the student, as shown in Fig. 1.

Keeping these considerations in mind, we can address the types of interactive computer software packages that can be used to assist in the training of chemical engineering student.

Table 1. Bloom's Taxonomy of Educational Objectives (Bloom, 1956).

1. *Knowledge*: The remembering of previously learned material. Can the problem be solved simply by defining terms and by recalling specific facts, trends, criteria, sequences, or procedures? This is the lowest intellectual skill level. Examples of knowledge-level assignments and questions are: *Write* the equations for a batch reactor and *list* its characteristics. *Which* reactors operate at steady state?
Other words used in posing knowledge questions: *Who . . . , When . . . , Where . . . , Identify . . . , What formula . . . .*

2. *Comprehension*: This is the first level of understanding and skill level two. Given a familiar piece of information, such as a scientific principle, can the problem be solved by recalling the appropriate information and using it in conjunction with manipulation, translation, or interpretation? Can you *manipulate* the design equation to find the effluent concentration or extrapolate the results to find the reactor volume if the flow rate were doubled? *Compare* and *contrast* the advantages and uses of a CSTR and a PFR. *Construct* a plot of N_A as a function of t.
Other words: . . . Relate . . . , Show . . . , Distinguish . . . , Reconstruct . . . , Extrapolate . . . .

3. *Application*: The next higher level of understanding is recognizing *which set* of principles ideas, rules, equations, or methods should be applied, given all the pertinent data. Once the principle is identified, the necessary knowledge is recalled and the problem is solved as if it were a comprehension problem (skill level 2). An application level question might be: *Make use* of the mole balance to solve for the concentration exiting a PFR.
Other words: . . . Apply . . . , Demonstrate . . . , Determine . . . , Illustrate . . . .

4. *Analysis*: This is the process of breaking the problem into parts such that a hierarchy of sub problems or ideas is made clear and the relationships between these ideas are made explicit. In analysis, one identifies missing, redundant, *and* contradictory information. Once the analysis of a problem is completed, the various subproblems are then reduced to problems requiring the use of skill level 3 (application). An example of an analysis question is: What conclusions did you come to after reviewing the experimental data?
Other words: . . . Organize . . . , Arrange . . . , What are the causes . . . , What are the components . . . .

5. *Synthesis*: This is the putting together of parts to form a new whole. A synthesis problem would be one requiring the type, size, and arrangement of equipment necessary to make styrene from ethyl benzene. Given a fuzzy situation, synthesis is the ability to formulate (synthesize) a problem statement and/or the ability to propose a method of testing hypotheses. Once the various parts are synthesized, each part (problem) now uses the intellectual skill described in level 4 (analysis) to continue toward the complete solution. Examples of synthesis level questions are: *Find a way* to explain the unexpected results of your experiment. *Propose* a research program that will elucidate the reaction mechanism?
Other words: . . . Speculate . . . , Devise . . . , Design . . . , Develop . . . , What

alternative . . . , Suppose . . . , Create . . . , What would it be like . . . , Imagine . . . , What might you see . . . .

6. *Evaluation***:** Once the solution to the problem has been synthesized, the solution must be evaluated. Qualitative and quantitative judgments about the extent to which the materials and methods satisfy the external and internal criteria should be made. An example of an evaluation question is: *Is the author justified* in concluding that the reaction rate is the slowest step in the mechanism?
Other words: . . . Was it wrong . . . , Will it work . . . , Does it solve the ***real*** *problem . . . , Argue both sides . . . , Which do you like best . . . , Judge . . . ,*

Table 2. Dimensions of the Inventory of Learning Styles (Soloman, 1992).

Dimension	Range	Comments
Processing	Active/Reflective	Active learners learn best by doing something physical with the information, while reflective learners do the processing in their heads (Felder, 1994).
Perception	Sensing/Intuitive	Sensors prefer data and facts, intuitors prefer theories and interpretations of factual information (Felder, 1989).
Input	Visual/Verbal	Visual learners prefer charts, diagrams and pictures, while verbal learners prefer the spoken or written word.
Understanding	Sequential/Global	Sequential learners make linear connections between individual steps easily, while global learners must get the "big picture" before the individual pieces fall into place.

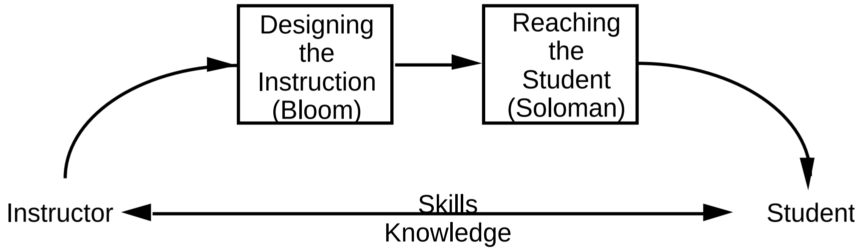


Figure 1. Incorporating Bloom's Taxonomy and Soloman's Inventory to impart skills and knowledge to students.

Types of Interactive Computing

Interactive computer-aided instruction in chemical engineering can be divided into four categories:

- Presentation
- Assessment
- Exploration
- Simulation

Presentation focuses on the delivery of technical material, which can occur in a number of ways. The following list is representative of these ways and the corresponding Soloman learning style dimension.

1. Display of text material (verbal).
2. Access to expanded explanation of text material through hot keys (active, sequential).
3. Visual and graphical representation of material (visual).
4. Use of animation to display phenomena (global) and manipulate equations (active).
5. Use of video clips to display industrial situations and situations where motion is involved (global, visual).

In the *presentation* phase the primary focus is on the knowledge, comprehension, and application levels of Bloom's Taxonomy. In the *assessment* category the student is tested on mastery of the material. The use of multiple-choice questions coupled with interactive simulations which the student must run to answer questions is one of the most effective testing methods. The simulations are closed-ended and focus on the first four levels of Bloom's Taxonomy (knowledge, comprehension, application, and analysis). The correct solutions to the questions are displayed immediately after the student's solution is entered. These types of assessments are particularly suited to the active, sequential and sensing learners.

The third category, *exploration*, allows users to better understand the role of various parameters on the performance of a given process through exploration of the process. These are exploratory simulations within a confined parameter space (simulations where equations are given and last terms may be dropped and the parameters take on any value). Instructional modules can also provide for the planning of experiments by allowing students to choose experimental systems, to take simulated "real" data, to modify experiments to obtain data in different parameter ranges, to manipulate data so as to discriminate among mechanisms, and to design a piece of equipment or process. These interactive computer modules can provide students with a variety of problem definition alternatives and solution pathways to follow, thereby exercising their divergent-thinking skills. Active learners enjoy the chance to manipulate information, and sensors and global learners get to experience a real process, or at least a simulation of it. This module type focuses on levels 3 and 4 of Bloom's Taxonomy (application and analysis). We will limit our discussion to those materials used in the core technical content courses, not the complex process simulations available for use in process design or laboratory courses. Examples of these exploratory materials are shown in Table 3, and are discussed in more detail in the historical overview.

The *simulation* category includes those activities in which the equations and parameter values are not given but can be easily entered in an ODE solver, such as POLYMATH, Maple, Mathematica, MATLAB, MathCAD, and spreadsheets. These tools allow users to better understand the role of various parameters on the performance of a given process through exploration of the process. This type gives the student practice of the higher levels of Bloom's Taxonomy (synthesis and evaluation).

Table 3. Exploratory Computer-Aided Instruction Materials.

PLATO Materials	Various
UM Interactive Modules (Drug Patch, Heat Effects)	Scott Fogler, Susan Montgomery et al. (U. of Michigan)
Lab Modules	Robert Squires, G.V. Reklaitis, S. Jayakumar et al. (Purdue U.)
Reactor Tool Kit	Bruce Finlayson (U. of Washington)
PICLES^{TM}	Doug Cooper (U. of Connecticut)

Historical Overview of Interactive Computing

PLATO

The early materials for interactive computer-aided instruction were developed for mainframe computers. While these materials proved very useful within individual universities, they were not easily distributable to other universities. They also suffered from the lack of availability of the materials to the students: most universities did not have adequate computer facilities to allow interactive access to lessons by large numbers of students. With the advent of personal computers such as IBM-PCs, the opportunities for interactive computer-aided instruction grew enormously. Interactive computer materials benefited greatly from the development of Control Data Corporation's PLATO (Programming Logic for Automated Teaching Operations) educational computer system. This system, at the University of Illinois in 1959, featured the use of terminals with touch-sensitive screens, as well as a highly efficient management and recording system (Smith, 1970, Smith and Sherwood, 1976). With the PLATO system, each terminal had a screen and keyboard that the students used to carry out their self-paced instructional lessons. The first interactive lessons used yes/no and keyword responses (e.g., does the unknown react with C_6 H_5 COCl in pyridine?). Later lessons simulated actual laboratory experiments. For example, students could explore the reaction of an olefin with bromine in methanol by varying the initial concentrations. Using known kinetic data, the computer immediately calculated and displayed the product composition. By experimenting with several sets of reaction conditions, students could understand the degree of completion involved in the reaction. Other lessons included the construction of NMR spectra, synthesis pathways, and determination of the unknowns. PLATO opened up the possibilities for truly interactive instruction.

POLYMATH

One of the earliest applications available to chemical engineers was POLYMATH, whose developers, Mordechai Shacham and Michael Cutlip, took advantage of the features of PLATO (Shacham and Cutlip, 1981a, 1981b, 1982, 1983). Using POLYMATH, students are able easily to set up a system of equations, to obtain an intuitive feeling of the problem being studied. This feeling and understanding is obtained because the student is able to take a significant amount of time to explore complex problems by varying the systems parameters and operating conditions rather than spending tedious time writing programs to enter and to solve the model equations for the physical system, including the required numerical methods. As a result, the student not only learns through discovery from the results of parameter variation, he or she has the opportunity to be creative in the solution to the problem. Thanks to POLYMATH, virtually every problem or homework assignment in chemical engineering can be turned into an open-ended problem that provides students with the extra time to practice their creative and synthesis skills.

University of Michigan Interactive Computer Modules

In the early 1970s, a number of interactive computer modules on reaction engineering were developed by Scott Fogler at the University of Michigan for use on a mainframe time-sharing computer. These modules focused on problem solving, freeing the student from the hassle of computation and mere formula plugging, so that the focus could be on process exploration and analysis. One popular example from this set is the Columbo module, in which students had to use their knowledge of chemical kinetics to solve a murder mystery. These simulations were enthusiastically received at a national meeting. Unfortunately, the programs were not easily transportable to computing environments at other universities. In the early 1980s, most of the six original reaction engineering modules were translated to IBM-PC format, using BASICA. These modules, especially one involving a styrene micro-plant, were distributed to and used at a limited number of universities.

The computing scene has changed tremendously from the early days described above. Students now have access to rooms full of computers at their universities, including IBM personal computers, Macintoshes, and UNIX workstations. In addition, many students have computers of their own, often connected to their university's computing environments, and through them, to world-wide networks such as the Internet. With the advent of sophisticated LCD panels, professors can display the contents of the computer screen using an overhead projector, easily bringing computers directly into the classroom. In the early 1990's, twenty-four interactive computer modules were developed at the University of Michigan and distributed by CACHE to every chemical engineering department in the United Sates and Canada. They were developed by Fogler, Montgomery, and Zipp and are described in detail elsewhere (Fogler, Montgomery and Zipp, 1992).

These modules employ presentation, assessment and exploration to enhance the students' mastery of the material. In addition, the sophistication required to satisfy a more computer-oriented student body has resulted in the need for more "polished" modules than in the past. The resulting interactive computer module should provide a learning experience that supplements the typical classroom and standard homework activities, reaching those students, particularly active, sensing, and global learners, who are not reached through traditional activities.

There are many advantages to using computer-based learning tools. There are also some pitfalls that one must be aware of and avoid. In addition to ensuring the technical accuracy of the material and simulations in the module, there are other considerations we have become aware of through our student testing and the comments of the external faculty testers. Some of the aspects of the use of interactive computer modules in engineering education that should be considered by all computer module developers are:

- Ease of use
- Introduction of new technologies
- Maintaining the focus on the concepts
- Eliminating tediousness!
- Promoting learning
- Individual guidance

The interactive computer modules for chemical engineering instruction developed at the University of Michigan typically consist of the following components:

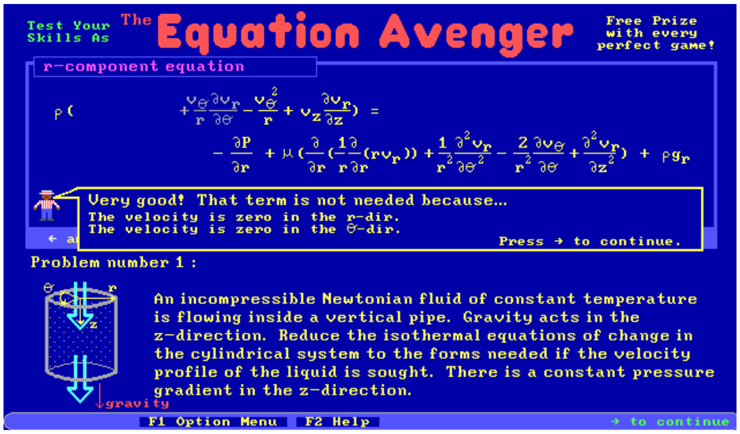
Introduction
Review of pertinent fundamentals
Demonstration
Interactive exercises
A branching component
Solution to the exercise
Evaluation

The review section makes extensive use of animation in the derivation of equations. The problem-solving session often includes a scenario that captures the student's interest. For example, in the SHOOT module (see Fig. 2), the objective is to master simplification of the equations of motion. The student must determine which terms in the equation should be dropped for a given situation. To make the learning more interesting, these decisions are made in an amusement park setting.

Students greatly appreciate this type of interaction. While they must still master the technical material, the experience itself is made more pleasant by the use of these external motivators.

The problem-solving session often includes a scenario that captures the student's interest. For example, in Kinetics Challenge 2 (see Fig. 3), the objective is to master some of the basic principles of stoichiometry. The student must answer a large number of multiple-choice questions. To make the learning more interesting, these questions are asked in a setting similar to the quiz show Jeopardy.

These modules were developed under sponsorship of the National Science Foundation and the University of Michigan; they are available to chemical engineering departments through CACHE.



$$\rho\left(\quad + \frac{v_\theta}{r}\frac{\partial v_r}{\partial \theta} - \frac{v_\theta^2}{r} + v_z\frac{\partial v_r}{\partial z}\right) =$$

$$-\frac{\partial P}{\partial r} + \mu\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial}{\partial r}(r v_r)\right) + \frac{1}{r^2}\frac{\partial^2 v_r}{\partial \theta^2} - \frac{2}{r^2}\frac{\partial v_\theta}{\partial \theta} + \frac{\partial^2 v_r}{\partial z^2}\right) + \rho g_r$$

Very good! That term is not needed because...
The velocity is zero in the r-dir.
The velocity is zero in the θ-dir.
Press → to continue.

Problem number 1 :

An incompressible Newtonian fluid of constant temperature is flowing inside a vertical pipe. Gravity acts in the z-direction. Reduce the isothermal equations of change in the cylindrical system to the forms needed if the velocity profile of the liquid is sought. There is a constant pressure gradient in the z-direction.

Figure 2. Interaction in SHOOT module, University of Michigan interactive computer modules.

Which of the following is the concentration of B for the nonisothermal elementary, irreversible, gas phase reaction A + 2B → 2C + 3D?
The feed is equimolar in A and B.

Select an answer (1-4) or
(ESC) to forfeit (no gain or loss) : 3

Host: Correct.

Figure 3. Interaction in KINETICS CHALLENGE 2 module, University of Michigan interactive computer modules.

Chemical Reactor Design Tool - University of Washington

The Chemical Reactor Design Tool developed at the University of Washington is primarily a set of exploration tools. This textbook supplement allows students to vary parameters and observe trends in complex chemical reaction engineering problems, including CSTR's, batch reactors, and plug flow reactors with axial and radial dispersion. The exploration of parameters allows users to very easily make their own deductions about the importance of physical phenomena, as well as to make design decisions that may revolve around conflicting constraints, none of which can be easily handled if one writes the program oneself. The interface is written in X-windows, providing the user with 3D perspective views, 2D contour plots, and solution variables, making it easy for the student to make comparison studies. In addition, the program automatically uses the correct, robust tools to solve the problem, so that the focus is on experimentation. This is ideal for active and global learners. The programs were developed under sponsorship of the National Science Foundation and the University of Washington, and are available to chemical engineering departments through CACHE.

Purdue-Industry Chemical Engineering Computer Simulation Modules

The Purdue-Industry Chemical Engineering Computer Simulation Modules are examples of the educational benefits that can result from collaborations with industry (Squires et al., 1992, Jayakumar et al.,1993). Their materials combine videotaped tours of portions of chemical plants with computer simulations of the systems, and allow students to perform “real world” design experiments to solve open-ended problems. These simulations have seen extensive use both in undergraduate laboratories, and in reactor design and process control courses. The key to the success of these modules is two fold: students can (1) see the actual plant through the videotape, and (2) use the computer simulation to study the effects of changes of system parameters on the operation of the system. These two features make the modules ideal for global and active learners.

Each module is written as an industrial problem caused by a change of conditions in an existing process, requiring an experimental study to re-evaluate the characteristic constants of the process. These might include, for example, reaction rate constants, equilibrium constants, heat transfer and mass transfer coefficients, and phase equilibrium constants. The student teams are expected to design experiments that will enable them to evaluate the needed constants. This is referred to as the *measurements* section of the problem.

After the constants have been determined, the student must validate them by using an existing computer model of the process, and comparing the simulated and experimental results. When they are convinced that their constants are reliable, the students must use these constants to predict some other specific process performance characteristics in the *applications* sections.

The Purdue modules are meant to supplement, not to replace, traditional laboratory experiments. Computer-simulated experiments have a number of advantages over traditional experiments:

- Processes that are too large, complex, or hazardous for the university laboratory can be simulated with ease on the computer.
- Realistic time and budget constraints can be built into the simulation, giving the students a taste of “real world” engineering problems.
- The emphasis of the laboratory can be shifted from the details of operating a particular piece of laboratory equipment to more general considerations of proper experimental design and data analysis.
- Computer simulation is relatively inexpensive compared to the cost of building and maintaining complex experimental equipment.

- Simulated experiments take up no laboratory space and are able to serve large classes because the same computer can run many different simulations.

These modules were developed under sponsorship of the National Science Foundation, Amoco Chemicals Corporation, Dow Chemical Company, Mobile Corporation, Tennessee Eastman Corporation, and Air Products and Chemicals, Inc., and are made available to universities through CACHE.

PICLES

One of the great opportunities afforded by the use of computers is the chance to simulate a real-time interaction with process equipment. One such simulator, focusing on process dynamics and control, is PICLES^{TM} (Process Identification and Control Laboratory Experiment Simulator), developed by Doug Cooper at the University of Connecticut (Cooper, 1993). PICLES is an IBM-PC training simulator that provides hands-on experience to students of process dynamics and control (see Fig. 4). It is used by over 60 chemical engineering departments around the world, and is very intuitive. Students using PICLES get experience in real-time use of P, PI, and PID control as well as using the Smith predictor, feed forward and cascade control, decouplers, and digital control of various systems, including: fluid level in gravity draining tanks, exit temperature in a heat exchanger, liquid level in a surge tank, and distillate and bottom composition in a distillation column (see Fig. 4).

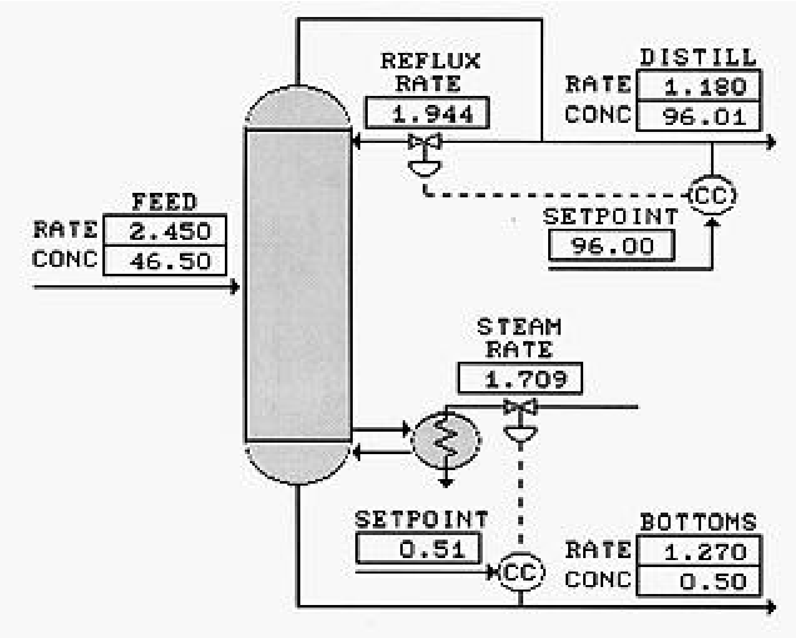


Figure 4. Process control of distillation columns using PICLES.

Current Practice and Future Directions

With the onset of CD-ROM technology, increasing computing speed and expanding computer memory, as well as more robust world-wide computer networks, we are at the threshold of a new revolution in interactive computer-aided instruction. Presentation of materials, which until now has been limited to the type of theoretical material presented in textbooks, is starting to incorporate investigation of “real world” processes, making the theoretical material come alive. Investigation, limited in the past to numerical simulations of processes, now is starting to include visual explorations into the processes themselves. Some examples are shown below.

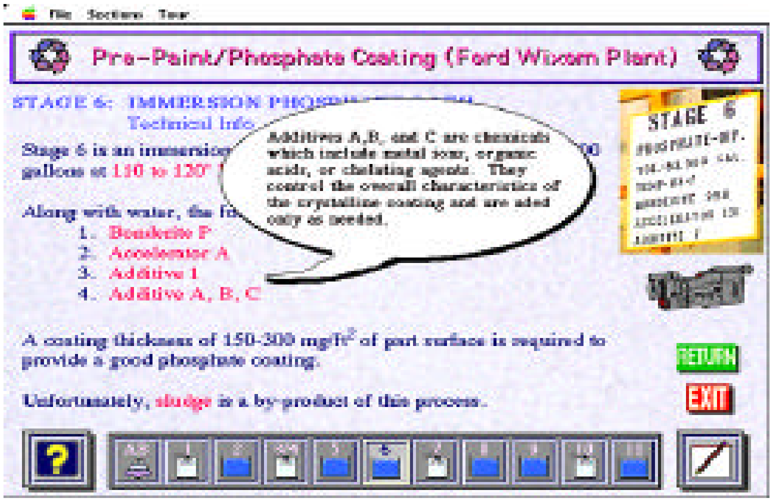
Multimedia and Hypertext

Multimedia presentations allow students to interact with information in a variety of ways. Qasem and Mohamadian (1992) found that multimedia allows the student to take an active role in the educational process, freeing the student from being a passive information target. Pohjola and Myllyla (1990) discuss an object-oriented hypertext approach to organizing educational chemical engineering information, setting the groundwork for future efforts, and highlighting the use of animation and other techniques to assist students in creating mind pictures of the steps occurring at the molecular level in chemical engineering processes. The next level of sophistication is the integration of still and moving images into these educational projects. Coburn et al. (1992), for example, includes video images, animation, sound and full-motion video in their modules for introductory thermodynamics. The materials are well presented, but incorporation of video images and full-motion video appear to be restricted to the motivators in the derivations of the concepts and in the historical perspective.

Computer-based educational materials that take full advantage of multimedia are starting to emerge. Susan Montgomery at the University of Michigan has developed some prototype multimedia materials for use in the material and energy balances course as well as in the chemical engineering undergraduate laboratory. These computer-based instructional materials integrate graphics, animation, video images and video clips into multimedia packages that allow students to learn the basic concepts in chemical engineering through exploration of actual situations ranging in scope from simple bench-scale experiments and day-to-day experiences to industrial chemical plants. For example, for an open-ended problem on mass balances, a multimedia module allows students to tour the phosphate coating system (see Fig. 5) of Ford Motor Company’s Wixom Assembly Plant. The module includes a description of each stage in the system, chemical usage, tank size and dump schedule information, and a short video clip of each stage.

In a module on multiphase systems (see Fig. 6), students can apply their expertise using T-xy diagrams to actual industrial equipment, a valuable experience for global learners

CACHE’s role in the implementation of multimedia materials for chemical engineering has been invaluable. The successful completion and delivery of the 25th anniversary CD-ROM was the culmination of uncounted hours of work for Peter Rony of Virginia Tech, head of the CACHE CD-ROM Task Force, and made possible by the vision of Michael Cutlip of the University of Connecticut, current past-president of the CACHE Corporation. The exploration of CD-ROM as an avenue for distribution before most universities had CD-ROM drives available is an example of the pro-active role CACHE has taken in fostering the development and implementation of interactive computer-based materials for chemical engineering courses.



Stage 6 is an immersion
gallons at 110 to 120°

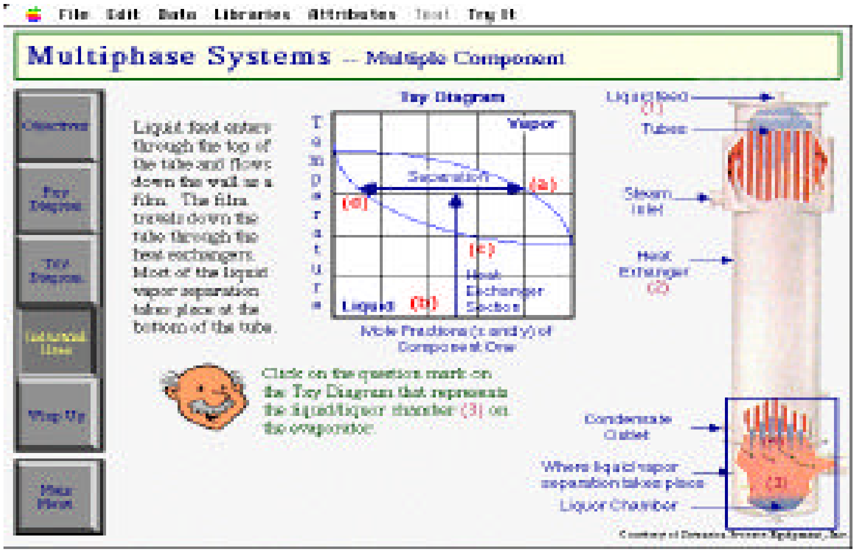
Along with water, the fo
1. Bonderite P
2. Accelerator A
3. Additive 1
4. Additive A, B, C

Additives A,B, and C are chemicals which include metal ions, organic acids, or chelating agents. They control the overall characteristics of the crystalline coating and are aded only as needed.

A coating thickness of 150-300 mg/ft² of part surface is required to provide a good phosphate coating.

Unfortunately, sludge is a by-product of this process.

Figure 5. Analysis of phosphate coating system.



Liquid feed enters through the top of the tube and flows down the wall as a film. The film travels down the tube through the heat exchangers. Most of the liquid vapor separation takes place at the bottom of the tube.

Click on the question mark on the Txy Diagram that represents the liquid/liquor chamber (3) on the evaporator.

Figure 6. Real world applications of liquid-vapor separation principles.

Virtual Reality

Another exciting area on the horizon is virtual reality (VR). CACHE has formed a Virtual Reality Task Force and within the next five years we should see a significant number of modules developed using virtual reality. One module currently under development at the University of Michigan by John Bell and Scott Fogler is the prototype of a chemical plant that uses a straight-through transport reactor with a coking catalyst (see Fig. 7). Here, the student uses VR to enter the plant lobby where he or she is given an overview of the process and is free to explore various parts of the room and video tapes at will, simply by moving a joy stick. After this introduction, the student enters the reactor room where he or she can change the operating parameters and see their effect on the reaction variables such as degree of coking and conversion. The student can travel *inside* the reactor to observe the coking and catalyst transport, and can even enter the catalyst pellet to view the pore space inside the pellet and see molecules reacting on the surface.

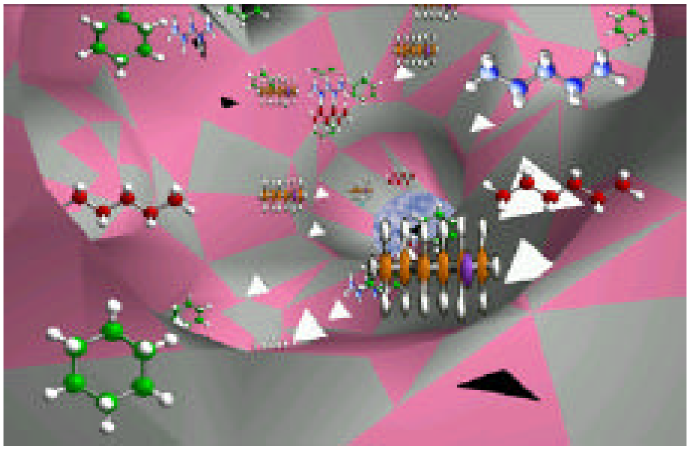


Figure 7. Exploration of the reactions taking place within a catalyst pellet - University of Michigan.

This visualization of the process and reaction mechanisms will greatly enhance the students' understanding and appreciation of this reaction engineering process. In general, the advent of virtual reality tools opens the door wide for all types of exploration.

Postscript

In surveying the literature prior to writing this article, the authors were surprised to see how few articles there are in the literature about instructional software that we know to exist. Our guess is that most early software developers were lone pioneers, with little time to devote to writing articles about their work, given the lack on importance placed on the development of educational materials by tenure and review committees. Recently, however, research focusing on the development of interactive computer-based educational materials seems to have garnered increasing respect in the academic community, as shown by the increasing number of peer-reviewed journals that can serve as outlets for dissemination about computer-aided instruction. These now include, among others, *Computer Applications in Engineering Education, The International Journal of Engineering Education, Computer Applications in Chemical Engineering, the Journal of Engineering Education,* and *Computers and Chemical Engineering.*

References

Bloom, Benjamin S. (1956). *Taxonomy of Educational Objectives; the Classification of Educational Goals, Handbook I: Cognitive Domain*. New York: David McKay Company.

Coburn, W.G. G.C. Lindauer, R.L. Collins, T.E. Mullin, and W.P. Hnat (1992). Development of multifaceted instructional modules for introductory thermodynamics. *Proceedings of IEEE Conference on Frontiers in Education*. 150-155.

Cooper, D.J. (1993). PICLES: The process identification and control laboratory experiment simulator. *CACHE News* **37**, 6-12.

Felder, R.M. (1989). Meet your students. I. Stan and Nathan. *Chemical Engineering Education,* Spring, 68-69.

Felder, R.M. (1994) Meet your students: V. Edward and Irving. *Chemical Engineering Education*, Winter, 36-37.

Felder, R.M., K.D. Forrest, L. Baker-Ward, E.J. Dietz, and P.H. Mohr (1993). A longitudinal study of engineering student performance and retention: I. Success and failure in the introductory course. *Journal of Engineering Education*, Jan, 15-21.

Felder, R.M. and L.K. Silverman (1988). Learning and teaching styles. *Engineering Education*, 674-681.

Fogler, H.S., S.M. Montgomery, and R.P. Zipp (1992). Interactive computer modules for chemical engineering instruction. *Computer Applications in Engineering Education,* **1**(1), 11-24.

Griskey, R.G. (1991). Undergraduate education: Where do we go from here? *Chemical Engineering Education*, *Spring*, 96-97.

Jayakumar, S., R.G. Squires, G.V. Reklaitis, P.K. Andersen, K.R. Graziani, B.C. Choi (1993). The use of computer simulations in engineering capstone courses: A chemical engineering example - the Mobil catalytic reforming process simulation. *International Journal of Engineering Education,* **9**(3), 243-50.

Kolb, D.A. (1984). *Experiential Learning: Experience as the Source of Learning and Developmen*t. Prentice-Hall, Englewood Cliffs, N.J.

Kulik, C.C. and J. A. Kulik (1986). *AEDS Journal*, **19**, 81.

Kulik, J.A. and C. C. Kulik (1987). *Contemporary Education Psychology*, **12**, 222.

McCaulley, M.H., E.S. Godleski, C.F. Yokomoto, L. Harrisberger and E.D. Sloan (1983). Applications of psychological type in engineering education. *Engineering Education*, 394-400.

Pohjola, V.J., and I. Myllyla (1990). Object-oriented hypermedia as a teaching aid in chemical engineering education. In TH. Bussemaker and P.D. Iedema (Eds.). *Computer Applications in Chemical Engineering*, Elsevier, Amsterdam. 199-202.

Qasem, I. and H. Mohamadian (1992). Multimedia technology in engineering education. *Proceedings. IEEE Southeastcon'92 (Cat. No. 92CH3094-0) vol. 1*. IEEE, New York. 46-49.

Shacham, M. and M.B. Cutlip (1981a). Educational Utilization of PLATO in chemical reaction engineering. *Computers & Chemical Engineering*, **5**, 215-224.

Shacham, M. and M.B. Cutlip (1981b). Computer-based instruction: is there a future in ChE education? *Chemical Engineering Education*, 78.

Shacham, M. and M.B. Cutlip (1982). A simulation package for the PLATO educational computer system. *Computers & Chemical Engineering*, **6**, 209-218.

Shacham, M. and M.B. Cutlip (1983). Chemical reactor simulation and analysis at an interactive graphical terminal. *Modeling and Simulation in Engineering*, 27.

Smith, S.G. (1970). The use of computers in teaching organic chemistry. *J. of Chemical Educatio*n, **47**, 608.

Smith, S.G. and D.A. Sherwood (1976). Educational uses of the PLATO computer system. *Science*, **192**, 344.

Soloman, B.S. (1992). *Inventory of Learning Styles*. North Carolina State University.

Squires, R.G., P.K. Andersen, G.V. Reklaitis, S. Jayakumar, and D.S. Carmichael (1992). Multimedia-based applications of computer simulations of chemical engineering processes. *Computer Applications in Engineering Education,* **1**(1), 25-30.

VanderStoep, S.W. and C.M. Seifert (1993). Learning 'How' vs. learning 'When': improving transfer of problem-solving principles. *Journal of the Learning Sciences,* **3**(1), 93-111.

GENERAL-PURPOSE SOFTWARE FOR EQUATION SOLVING AND MODELING OF DATA

Mordechai Shacham
Ben-Gurion University of the Negev
Beer-Sheva 84105, Israel

Michael B. Cutlip
University of Connecticut
Storrs, CT 06269

N. Brauner
Tel-Aviv University
Tel-Aviv 69978, Israel

Abstract

Much of the educational emphasis in numerical computing has shifted from FORTRAN or other source code programming languages to the use of general-purpose packages which typically solve Nonlinear Algebraic Equations (NLE), Ordinary Differential Equations (ODE), and carry out the computations required for data modeling and correlation. The educational advantage in using these programs is that they require the user to develop and input the model equations, but carry out the technical numerical details of the solution method without user intervention. In this paper comparisons of four such packages (MAPLE, MATLAB, MATHEMATICA and POLYMATH) are made for solving NLE's, ODE's, and for regression of data. References for test problems are identified, and the performances of the packages with the test problems are compared. Improvements are suggested in the solution and result storage algorithms, as well as in some "user friendly" features.

Introduction

The role of computers for numerical solution of chemical engineering problems was recognized early on over thirty years ago. The first textbook to address this subject was that by Lapidus (1962). This textbook included chapters on polynomial approximation, solution of ODE's, partial differential equations (PDE's), linear equations and NLE's, and the least squares error approach. There was also a chapter on optimization and control.

Seven years after the publication of the textbook by Lapidus (1962), the CACHE Corporation was founded. This paper concentrates on the solution of NLE's, ODE's and on data modeling and correlation. Optimization and control are covered in two additional separate papers in this monograph. The numerical solution of PDE's is still considered too difficult to be included in the undergraduate curriculum and is not covered here.

The history of the use of computers in chemical engineering education has been documented by Seader (1989). The publication of the textbook by Carnahan, Luther and Wilkes (1969) on numerical methods and the textbook by Henley and Rosen (1969) on material and energy balances formulated for digital computer use are mentioned as important developments in computer applications in chemical engineering. The CACHE Corporation published a seven volume set of books entitled *Computer Programs for Chemical Engineering Education* in 1972 with examples in seven curriculum areas. The examples included problem statements with solutions that were mostly via programs in FORTRAN. A few CSMP (Continuous System Modeling Package, available on mainframe computers) programs were also included. 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 matter in most chemical engineering courses, but they were actually the most time consuming and frustrating parts of a computer assignment. The computer indeed enabled the students to solve realistic problems, but the time spent on technical details which were of minor relevancy to the subject matter was much too long.

In order to solve this dilemma there was a growing tendency to provide the students with computer programs that can solve one particular type of problem. Listings, or even disks containing small programs, were included in textbooks, and large scale process simulators, such as FLOWTRAN (Seader, Seider and Pauls, 1979) were made available to departments through CACHE or from commercial vendors. The provision of a complete program to students for a particular problem has the disadvantage that the connection between the mathematical model and the problem is obscured. Students can only provide the input data, and then can only observe the results. The very important step of converting physical phenomena to a mathematical model is missing. Furthermore, each class of problems requires learning use of a different software tool and the input format of a particular program does not contribute to the learning of the subject matter.

The latest approach in chemical engineering education is to use general-purpose packages for problem solving. These packages require the student to input the mathematical model and the numerical data, but the programs carry out all the technical steps of the numerical solution. This approach is demonstrated by Fogler (1993), who discusses his method of teaching the chemical reaction engineering course. For example, his proposed approach for isothermal reactor design is to use mole balance, rate laws and stoichiometry in setting up the mathematical model and then to use a user friendly ODE solver such as POLYMATH (Shacham and Cutlip, 1994a) for combining and solving the equations. In this way, the use of the computer becomes just a routine, efficient and natural step in problem solution.

A general-purpose program for educational use must be versatile enough so that it can be used in most courses and laboratories during a four year educational program. It must be inexpensive and capable of running on inexpensive computers so that students can install and run it on their own computers. It must be user friendly so that it is easy to learn and easy to relearn. The technical details of the solution should take up only a minimum amount of the student's time. While most programs claim to be user friendly, there are actually objective measures to compare user friendliness of numerical software for equation solving and regression programs which will be presented later in this paper.

A recent survey by CACHE (Davis, Blau and Reklaitis, 1994) has indicated that only a disappointingly small percentage of engineers use numerical methods. Only 3% of the engineers use numerical libraries frequently and only 1% use mathematical packages. In comparison, 74% use spreadsheet programs frequently and 23% use them occasionally. There is a growing gap between the specialized users who, according to an article by Boston, Britt and Tayyabkhan (1993), will be able to solve "systems involving several hundreds of thousands of equations with several hundreds of degrees of freedom" by the year 2000 and the more typical engineers whose most sophisticated computational tool will remain the spreadsheet. The reasons for the limited use of numerical methods can be best described by observations made by Seader (1989) and deNevers and Seader (1992). Seader (1989) notes: "One problem is the lack of chemical engineering textbooks which include examples and exercises that presume that the student has a working knowledge of numerical techniques for solving sparse and dense systems of linear and nonlinear equations. Even if the textbooks were available, they might not be widely used because of the second problem, which is that many chemical engineering educators are either not familiar with or not comfortable with the use of computers to solve equations." A similar observation is made by deNevers and Seader (1992): "Since the advent of digital computers, textbooks have slowly migrated toward computer solutions of examples and homework problems, but in many cases the nature of the examples and problems has been retained so that they can be solved with or without a computer."

In short, the problem is that the available software in recent years has not been "user friendly" enough to entice most chemical engineering educators to use numerical methods. Many faculty have practiced "computer avoidance" and consequently many efficient new design techniques have not been introduced or used in classes. Furthermore, most examples in newer textbooks remained the same as those used before the advent of the computer. Often, the easiest faculty approach is to "computerize" by using spreadsheets alone, while avoiding the more realistic educational problem solving which numerical methods can provide.

Available Software

Extensive listings of the available chemical engineering software can be found in the annual CEP software directory which is published by the AIChE as a supplement to the December issue of *Chemical Engineering Progress*. The relevant software is listed under the category of "Mathematics and Statistics." This software directory gives information on what computer configuration is needed to run the software and often also the price of the software.

The software for solving equations and modeling data can be divided into three categories: (1) subroutine libraries, (2) single- or double-purpose programs, and (3) multi-purpose mathematical packages.

The 1995-96 edition of the CEP software directory lists several subroutine libraries written in FORTRAN, BASIC, Assembler and C. Our work has extensively used the mainframe version of the IMSL (1982) library as well as many of the subroutines from the library associated with the book entitled *Numerical Recipes* (Press, et al., 1986). These subroutines are all very well debugged and are dependable. The use of subroutine libraries is a must for someone who does his or her own programming, but these libraries lack the friendly user interface which is needed for a program to be effectively utilized in undergraduate engineering education.

Single-purpose programs are available for solving NLE's (TK Solver, Slaughter et al, 1990), for solving ODE's (ACSL, SimuSolv (Steiner, Rey and McGroskey, 1993), Tutsim (p. 532 in Coughanour, 1991)) and many programs for modeling of data. These programs are usually targeted toward specialized users, and they are typically too expensive and overly complicated for use in undergraduate chemical engineering education. Use of a single-purpose program also means that the student must learn many different programs to deal with a variety of problems.

The multi-purpose general mathematical packages such as POLYMATH (Shacham and Cutlip, 1994a), MAPLE (Ellis et al., 1992), MATLAB (MathWorks, 1991), and MATHEMATICA (Wolfram, 1991) are the most appropriate general equation solving and data modeling tools in chemical engineering education. These packages have inexpensive educational site licenses or student versions for most personal computers. User interfaces are provided to minimize programming; however, the effort required by the user to learn to use the program and to prepare the input data differs from one package to another. These differences among the packages will be discussed in more detail in the next two sections.

Spreadsheet programs for equation solving are not considered in this paper. They can be used for solving NLE's, ODE's and even PDE's, as was shown by Rosen and Adams (1987), but such use does require programming. In this respect, spreadsheets do not have advantages over programming languages. They can however be very helpful in simple calculations such as required in a "Mass and Energy Balance" course (Carnahan, 1993; Misovich and Biasca, 1990). Spreadsheets can also be used for data modeling and analysis. This aspect will be further discussed in a later section. In our opinion, the undergraduate chemical engineering computational toolkit for students must contain both a spreadsheet program and a package for equation solving and data modeling.

Solution of Nonlinear Algebraic Equations

Solution of NLE's are required in all chemical engineering courses. Typical examples include calculation of compressibility factor using an equation of state, adiabatic flame temperature, phase equilibrium calculations in ideal and nonideal mixtures, chemical equilibrium calculations, flowrate and pressure drop in a pipeline and pipeline networks, and steady state material and energy balances in chemical reactors. Many such examples are discussed in detail by Shacham and Cutlip (1994b). Equations of state, adiabatic flame temperature, and phase equilibrium calculations for ideal mixtures are representative of problems that are easily solved. That is, available programs will converge to the solution from reasonable initial estimates. Chemical equilibrium calculations, steady state material and energy balances on chemical reactors, and nonideal phase equilibrium problems are often very difficult to solve.

Problems in this category require accurate initial estimates that are very close to the solution, or some manipulation of the equations in order to enable the NLE solver to achieve a solution.

Recently, Shacham, Brauner and Pozin (1994) have compared four general-purpose packages (MAPLE, MATLAB, MATHEMATICA and POLYMATH) for their ability to solve sets of test problems taken from chemical engineering applications. The test problems were taken from Shacham (1984, 1989) and Shacham and Cutlip (1994b). None of the NLE packages could solve all the test problems from all sets of initial estimates. Thus it can be expected that any one of the packages will fail occasionally; therefore, it is important that the package should clearly signal when no solution is achieved. Some of the packages converged to a local minimum without indicating that solution had not been reached, and one package went into an infinite loop necessitating the rebooting of the computer.

This study concluded that the convergence intervals of both MAPLE and POLYMATH for difficult single nonlinear equations are the widest, and that multiple solutions are usually detected. For systems of NLE's, the largest number of cases were solved by MATHEMATICA and POLYMATH. All the programs except POLYMATH converged occasionally to a local minimum without giving a warning message. The largest number of such incidents were observed with MATLAB.

Shacham, Brauner and Pozin (1994) have recommended several "user-friendly" features to be included in an interactive NLE solver in relation to the four packages:

> *1. Menu-based program control instead of command-based control.*
>
> The program is easiest to use when all the available options are presented on the screen. The advantage of the menu based control becomes less obvious for complicated programs because of the need to search through several menu levels. POLYMATH was the only package which was completely menu-based.
>
> *2. Notation and format used in equation entry.*
>
> For the user it is most convenient when the equations can be entered into the package using almost the same notation and format as used in the mathematical model. Obviously there must be some rules for equation input, but there were major differences between the very flexible structure used by MAPLE and POLYMATH and the subroutine like structure required by MATLAB.
>
> *3. Equation ordering and detection of implicit relationships among variables.*
>
> When formulating a mathematical model, the general rule is usually considered first and the connections among variables are defined afterwards. The order of calculation is often the opposite, and it is much more convenient for the user when the reordering is done by the package. Often implicit relationships exist among variables that are difficult to detect by inspection. The package should find such relationships and inform the user. POLYMATH, MAPLE and MATHEMATICA reorder equations in the correct computational order; MATLAB does not perform any reordering. Only POLYMATH detects and issues appropriate error messages for existence of implicit relationship among variables.

> *4. Debugging aids.*
>
> All the packages can detect syntax errors after an equation has been entered. Only POLYMATH keeps a list of undefined variables. Such a list can be very helpful in detecting the misspelling of the name of a variable or in reminding the user of constants or variables that have not yet been defined.
>
> *5. Root verification and multiple solutions.*
>
> Depending on the solution method used, programs may converge to a local minimum instead of the root of the system of equations. The user must be made aware that the true solution has not been found. POLYMATH always displays the norm of the function values at the solution, whereas, the user must explicitly request calculation and display of the function values in the other packages. If there are several roots for a system of equations, all packages will find one solution when started from one set of initial estimates. The only exception is that POLYMATH finds up to five roots for a single equation inside an interval specified by the user.
>
> *6. Importance of initial estimates.*
>
> All the packages require the user to provide initial estimates for the unknowns. Shacham, Brauner and Pozin (1994) have shown that selection of initial estimates can be critical for highly nonlinear problems (such as chemical equilibrium, nonideal phase equilibrium, etc.) regardless of the program used. Often, transformations can be used to reduce the nonlinearity of the problem.

Solution of Nonlinear Ordinary Differential Equations

Numerical solution of ODE’s is extensively utilized in Process Dynamics and Control and Chemical Reaction Engineering courses. In fact, the first documented use of a specific ODE solver language was in control courses (CSMP was used in the “Control” volume of “Computer Programs for Chemical Engineering Education”, published by CACHE in 1972). Solution of ODE’s is also useful in simulating batch processes, such as batch distillation and solving steady state heat and mass transfer problems. Many examples requiring numerical solution of ODE’s discussed in detail by Shacham and Cutlip (1994b).

From the numerical point of view, many types of ODE’s are relatively easy to solve. They can be solved with high accuracy with some variation of the 4th-order Runge-Kutta (RK4) algorithm with error estimation and step size control (Press et al., 1986, pp. 554-562). Unfortunately, there are certain classes of problems which are difficult to solve. Carnahan and Wilkes (1981) mention reaction kinetics and distillation column modeling as being of the more difficult type. One common source of difficulty is that the system of equations is “stiff.” Stiff system of equations contain variables that change (decay) in widely varying time scale. In such cases, an explicit integration method, such as the RK4 method, is forced to use extremely small step sizes; consequently, a very large number of steps is required to integrate over the desired time interval. There are implicit and semi-implicit methods that can handle stiff equations effectively as discussed by Carnahan and Wilkes (1981), but these methods are much more complex than the RK4 method.

Another source of difficulty is singular (or more precisely, almost singular) points in the region of the integration. If, for example, one of the variables approaches zero and there is division by this variable, reduction of the step size to infinitesimally small values is required. The step adjusting algorithm may often "overstep" the singular point, causing the zero-bounded variable to become negative. From this point on, the integration is continued without any difficulty, giving incorrect results.

Equations with periodic solution may present difficulties, not with the precision of the solution itself, but in the presentation of the solution. Graphical or tabular presentation of the results is done in discrete time intervals. If the time intervals used are not small enough, details of the solution may be omitted, and instead of a continuous curve, broken lines are obtained.

The consequences of all these difficulties are that the user often cannot be certain of the accuracy or even validity of the numerical solution obtained. Unlike the solution of nonlinear algebraic equations, there is no simple way to verify the accuracy of the solution. Test problems that can be handled with RK4 algorithms can be found, for example, in Shacham and Cutlip (1994), while a moderately difficult stirred tank reactor problem is presented by Shacham, Brauner and Cutlip (1994a). Very stiff test problems are listed by Enright and Hull (1976) and Johnson and Varney (1976).

A number of test problems from Brauner, Shacham and Cutlip(1994) and a few stiff systems for mainly reactor design problems from Fogler (1992) have been solved using POLYMATH, MATLAB, MATHEMATICA and MAPLE. The solution algorithms used by the different programs are shown in Table 1.

Table 1. Integration Algorithms in the Four Packages.

	Nonstiff Algorithms	Stiff Algorithm
POLYMATH	RK3-RK4	Semi-implicit Euler
MATLAB	RK2-RK3, RK4-RK5, Adams	Gear
MATHEMATICA	No documentation for algorithm used	
MAPLE	RK4-RK5	- - -

MATLAB stores information on all integration steps for variables for which information is requested. POLYMATH and MATHEMATICA store results at even time intervals, MAPLE does not store the results for plotting, but plots during integration using "even" time intervals for adding a new section to the plot. In MATHEMATICA the user can change the frequency of the data storage and in MAPLE the frequency of the plot update. During the solution, only POLYMATH provides a plot of step size and local truncation error history. The other packages do not interact with the user during solution time and do not provide any integration error or step size information.

The test problems were intended to detect and demonstrate weaknesses of the packages. For the nearly singular test problem where the concentration of one component approaches zero, all packages (except MATHEMATICA with a very small tolerance) “overstepped” the nearly singular point and continued with negative concentration of this component. The use of MATLAB with very high precision caused the program to stop the integration with error message “singularity likely.”

For the stiff test problems, POLYMATH usually detected that the problem was stiff and recommended the use of the implicit Euler method. With the implicit Euler method, the solution obtained was unstable, oscillatory and inaccurate. When MATLAB was used with error tolerance not small enough, it did not detect the existence of stiffness and simply returned with incorrect results. With smaller error tolerance, the solution obtained was fairly accurate, but small oscillations of the stiff variable were noticeable in the results. The behavior of MAPLE for stiff systems was inconsistent. The program often returned with error messages or without any results or messages. Other times, the solutions were very accurate. For stiff problems, the performance of MATHEMATICA was the best, as it gave correct results using the default error tolerance values.

The basic quality of the plots was best with MATLAB where the results of all the integration steps are stored and can be displayed. In POLYMATH and MAPLE where the results are stored or displayed at fixed intervals, details of the solution are sometimes poorly plotted. If the solution is periodic, only two cycles could be displayed in full details with POLYMATH or the default setting of MAPLE. Attempts to increase the final time (adding more cycles) caused the graph to be displayed as broken lines instead of a continuous curve. In MAPLE, the display interval could be changed by the user; in POLYMATH there was the need to divide the integration interval to smaller parts in order to get a continuous curve in such cases. The Unix version of MATHEMATICA was used in this work, and this version did not provide plots of comparable quality to the other packages. The results when transferred to MATLAB gave plots as detailed as did the MATLAB package alone.

Solution of stiff ODE’s may take considerable amount of time. For one particular example containing 12 differential equations the time required was: POLYMATH - 1 min. (including restart of the solution twice), MATHEMATICA - 1.5 min., MATLAB - 0.5 min. In MAPLE, the integration interval had to be shortened, because use of the full interval led to continued execution for 10 minutes and the graph was not plotted. Defining a shorter integration interval resulted in 5 minute run time and the results were plotted. In any case, the integration time for stiff problems may be long. Therefore, it is important to keep the user informed that calculations are proceeding and how far the integration has gone. Only POLYMATH provides such information.

The user-friendliness of the different packages was similar to that for NLE’s. The format required by MAPLE and MATHEMATICA for entering the ODE’s was even more rigid, and the rules for the modifications required are less obvious than for entering NLE’s.

Data Modeling and Correlation

Data modeling and correlation is useful in most chemical engineering courses and laboratories. Typical examples are correlation of physical and thermodynamic properties, phase equilibrium, heat transfer coefficients, and reaction rate for homogeneous and heterogeneous reactions. Often the need arises to integrate or differentiate tabular data, such as when calculating mean heat capacity, vapor liquid equilibrium from total pressure measurements or using the differential method of reaction rate data analysis. Many detailed examples are discussed by Shacham and Cutlip (1994b).

Most of the statistical techniques for analyzing the accuracy of correlations have been known for several decades (Himmelblau, 1969), but these were rarely used on mainframe systems because of the limitations of the computational and graphical tools when using batch computations. Today, many interactive statistical analysis programs are available which can model and correlate data. The best known among these programs are SPSS (1983) and SAS (Cary, 1988). These and other similar programs are intended to be used by statisticians; they are unnecessarily complicated for routine use by engineers and engineering students to correlate engineering data. Fortunately, there are programs which focus on data modeling and analysis, and many of them are listed in the CEP software directory.

In order to select the most appropriate program for data modeling and analysis, the program requirements must be clearly defined. Some of the requirements are discussed in the book by Noggle (1993); however, the discussion is limited to models containing a single independent variable. The results of our investigations in this area are discussed in more detail in Shacham, Brauner and Cutlip (1994b).

A regression package appropriate for undergraduate engineering students or for practicing engineers must be able to perform linear, polynomial, multiple linear and multiple nonlinear regressions. It also should be able to determine cubic splines for relationships that cannot be adequately represented by known models. The package should provide options to extrapolate, differentiate and integrate tabulated data using the regression curve. Tools for transforming the original data (for models containing expressions such as log(P) or 1/T, etc.) should also be provided.

All regression programs provide some statistical information. In our experience, the reliance on one, or even several statistical variables indicating the quality of the fit (such as the variance or linear correlation coefficient) can be misleading, especially when the underlying assumptions in calculating these variables are not well understood (Shacham, Wisniak and Brauner, 1993; Shacham, Brauner and Cutlip, 1994b).

The quality of the experimental data can be checked by plotting the independent variables one versus another. This way, linear dependence between the independent variables can be detected. If the model is comprised of sums of terms of different order of magnitude (as in rate expressions for reversible reactions), plots can be derived which can indicate whether the data are precise enough to represent the contributions of the less significant terms.

When there are several candidate models for representing the same data, models that contain an insufficient number of parameters can be eliminated using residual plots. Models with excessive parameters can be eliminated using confidence intervals and the mean sum of squares to select from among candidate models. Practical selection of the best method for estimation of the parameters (i.e., linear versus nonlinear regression) should utilize error analysis of the transformation functions, residual plots and comparison of mean sum of squares. The regression package must provide options for plotting one variable versus another one, construction of residual plots, calculation of confidence intervals on the parameter values, and calculation of the mean sum of squares in order to facilitate the statistical analysis.

All of the above capabilities are available in the POLYMATH package, and we have used them successfully with undergraduate students. Most of the desirable options are also available or can be generated with little effort with the more advanced spreadsheets programs (an example is Quattro Pro 4.0). All the above capabilities (and some more) except for multiple linear and nonlinear regression, are available in the EZFIT program (Noggle, 1993).

Conclusions

There are several software packages that can be used as general-purpose tools for solving NLE's, ODE's and for modeling and correlating data that are useful within the undergraduate chemical engineering curriculum. These programs require the user to input the model and the data, but many of the technical details of the problem solution need not be considered during normal use. It is expected that the use of such packages will increase as educators realize their potential value and develop new design and calculation techniques that enable students to efficiently solve more realistic problems.

In general, existing packages can be improved by making the user interface more "friendly" by making the notation and the model entry format more flexible, carrying out equation ordering, detecting implicit connections among variables, and by providing the user with more information regarding the validity of the solution. The NLE solution algorithms should be improved to make them less sensitive to initial estimates in solving difficult problems, such as chemical or nonideal phase equilibrium problems. The ODE integration algorithms can be improved by making them monitor more closely the condition of the equations so that they can detect stiffness, singularity or other conditions which can make the solution inaccurate or incorrect and inform the user or change solution method accordingly. The ODE solvers also need an adaptive result storage scheme, where the results are stored at variable time intervals so as to closely follow the curvature of the solution.

In data modeling and correlation, the available software is adequate, but in order to use this software most effectively the study of statistical concepts must be reinforced, so that the chemical engineer can select and use the most significant statistical tests for a particular problem and not rely on tests when underlying assumptions are violated.

The use of the new interactive software tools will probably be the most beneficial for students and practicing engineers who did not learn to program or do not program well in languages such as FORTRAN. Over fifteen years of experience has reinforced our opinion that interactive programs on personal computers for the numerical solution of NLE's, ODE's, and data correlation will indeed become increasingly important tools in chemical engineering education.

References

Boston, J. F., H. I. Britt, and M. T. Tayyabkhan (1993). Software: Tackling tougher tasks. *Chem. Eng. Progr.*, **89**(11), 38.

Brauner, N., M. Shacham, and M. B. Cutlip (1994). Application of an interactive ODE simulation program in process control education. *Chem. Engr. Ed.*, **28**(2), 130-135.

Carnahan, B., H. A. Luther, and J. O. Wilkes (1969). *Applied Numerical Methods*. Wiley, New York.

Carnahan, B. and T. O Wilkes (1981). Numerical solution of differential equations - an overview. In R.S.H. Mah and W. D. Seider (Eds.), *Foundation of Computer Aided Chemical Process Design*, Engineering Foundation, New York, 225-341.

Carnahan, B. (1993). Personal Communication.

Cary, N. C. (1988). *SAS Language and Procedures Usage*, Version 6, SAS Institute.

Coughanowr, D. R. (1991). *Process Systems Analysis and Control*. 2nd ed. McGraw-Hill, New York.

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.*, submitted for publication.

Ellis, W. Jr., E. Johnson, E. Lodi, and D. Schwalbe (1992). *MAPLE V Flight Manual*. Brooks/Cole Pub. Co., Pacific Grove, California.

Enright, W. H. and T. E. Hull (1976). Comparing numerical methods for the solution of stiff systems of ODE's arising in chemistry. In Lapidus, L. and W. E. Schiesser (Eds.), *Numerical Methods for Differential Systems*, Academic Press, New York. 45-66.

Fogler, H. S. (1992). *The Elements of Chemical Reaction Engineering*, 2nd ed., Prentice-Hall, Englewood Cliffs, New Jersey.

Fogler, H. S. (1993). An appetizing structure of chemical reaction engineering for undergraduates. *Chem. Engr. Ed.*, **27**(2), 110.

Forsythe, G. E., M. A. Malcolm, and C. B. Moler (1977). *Computer Methods for Mathematical Computation*, Prentice-Hall, Englewood Cliffs, New Jersey.

Henley, E. J. and E. M. Rosen (1969). *Material and Energy Balance Computation*. Wiley, New York.

Himmelblau, D. M. (1970). *Process Analysis by Statistical Methods*. Wiley, New York.

IMSL Library Users Manual. 9th Ed., IMSL, Houston, Texas (1982).

Johnson, A. I. and J. R. Barney (1976). Numerical solution of large systems of stiff ordinary differential equations in a modular simulation framework. In Lapidus L. and W. E. Schiesser (Eds.), *Numerical Methods for Differential Systems*, Academic Press, New York. 97-124.

Lapidus, L. (1962). *Digital Computation for Chemical Engineers*. McGraw-Hill, New York.

Math Works, Inc. (1991). *MATLAB for Unix Computer's Users' Guide*. Natick, Massachusetts.

Misovich, M. and K. Biaslea (1991). The power of spreadsheets in a mass and energy balances course. *Chem. Engr. Ed.*, **25**(1), 46.

deNevers, N. and J. D. Seader (1992). Helping students to develop a critical attitude towards chemical process calculations. *Chem. Engr. Ed.*, **26**(2), 88-93.

Noggle, J. H. (1993). *Practical Curve Fitting and Data Analysis*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Press, W. H., B. P. Flannery, S. A. Teukolsky and W. T. Vetterling (1986). *Numerical Recipes*. Cambridge University Press, Cambridge, UK.

Rosen, E. M., and R. N. Adams (1987). A review of spreadsheet usage in chemical engineering calculations. *Comp. & Chem. Engr.*, **11**(6), 723-736.

Seader, J. D., W. D. Seider, and A. C. Pauls (1979). *FLOWTRAN Simulation - An Introduction*. CACHE, Ann Arbor, Michigan.

Seader, J. D. (1989). Education and training in chemical engineering related to the use of computers. *Comp. & Chem. Engr.*, **13**(4/5), 377-384.

Shacham, M. (1984). Recent developments in solution techniques for systems of nonlinear equations. In *Proc. of the Second International Conference on Foundations of Computer Aided Design*, (Ed.) A. W. Westerberg and H. H. Chien, pp. 891-923. CACHE Publications, Ann Arbor, Michigan.

Shacham, M. (1989). An improved memory method for solution of a nonlinear equation. *Chem. Eng. Sci.*, **44**, 7, 1495-1501.

Shacham, M., J. Wisniak, and N. Brauner (1993). Error analysis of linearization methods in regression of data for the Van Laar and Margules equations. *Ind. Eng. Chem. Res.*, **32**, 2820-2825.

Shacham, M., N. Brauner and M. Pozin (1994). Comparing software for interactive solution of systems of non-linear algebraic equations.

Shacham, M., N. Brauner, and M. B. Cutlip (1994a). Exothermic CSTR's: Just how stable are the multiple steady states? *Chem. Engr. Ed.*, **28**(1), 30-35.

Shacham, M., N. Brauner, and M. B. Cutlip (1994b). Critical analysis of experimental data, regression models and regressed coefficients in data correlations. Proceedings of the Conference on *Foundations of Computer-Aided Process Design*, 305-308, Snowmass, Colorado.

Shacham, M. and M. B. Cutlip (1994a). *POLYMATH 3.0 Users' Manual*. CACHE Corporation, Austin, TX.

Shacham, M. and M. B. Cutlip (1994b). *Numerical Solution of Chemical Engineering Problems Using POLYMATH*. Prentice-Hall, Englewood, New Jersey (to be published, 1997).

Slaughter, J. M., J. N. Peterson and R. L. Zollars (1991). Use of PC based mathematics software in the undergraduate curriculum. *Chem. Eng. Ed.*, **25**(1), 54-60.

SPSS User Guide, McGraw-Hill, New York (1983).

Steiner, E. C., T. D. Rey, and P. S. McGroskey (1993). *SimuSolv Reference Guide*, The Dow Chemical Company, Midland, Michigan.

Westerberg, A. W. (1993). Human aspects: Redefining the role of chemical engineers. *Chem. Eng. Prog.*, **89**(11), 60.

Wolfram, S. (1991). *MATHEMATICA, A System for Doing Mathematics by Computer*, 2nd ed. Addison Wesley, NY.

THERMODYNAMICS AND PROPERTY DATA BASES

Aage Fredenslund, Georgios M. Kontogeorgis and Rafiqul Gani
Chemical Engineering Department
Technical University of Denmark
2800 Lyngby, Denmark

Abstract

The formulation of the thermodynamic equilibrium problem and the methodologies used for its solution are presented in this chapter. The resulting equations need a number of pure component and mixture equilibrium properties, which, whenever not experimentally available at the conditions of interest, must be estimated. Several predictive techniques for preliminary estimation of a variety of properties are discussed. Special emphasis is given to the group-contribution methods (especially those related to the UNIFAC model) and the recent advances in mixing rules for cubic equations of state. Three application examples illustrating the applicability of some of the presented models to difficult systems (mixtures with polymers, electrolytes and multiphase equilibria) are given. Finally, some suggestions for future research in the area of applied thermodynamics are included.

Introduction

Phase equilibrium thermodynamics plays a particularly important role for process development in chemical, petroleum, and related industries. We mention here some examples: Separation of the products of organic synthesis and removal of by-products are often encountered in engineering practice. These separations are usually accomplished with conventional separation processes (e.g., distillation, extraction), which are based on thermodynamic equilibrium between two or more phases. However, some recent applications related to the food and biochemical industries (e.g., separation and fractionation of fish-oil related compounds) are more efficiently handled with non-conventional separation techniques, such as supercritical fluid extraction. Furthermore, since major products of interest to the chemical and petroleum industries are of complex (and often macro-) molecular structure, computer-aided product design will help to identify those materials which best satisfy specific requirements from an economic point of view (Zeck and Wolf, 1993). Satisfactory knowledge of thermodynamic, as well as physical, properties of the materials is needed for efficient product design.

Product development and process design require molecular-sound correlation models, and especially, predictive methods. Group-Contribution (GC) techniques have found wide applicability due to their successful compromise between simplicity and accuracy. The purpose here is not to give a complete review of all the successful GC methods, but rather to outline those methods and procedures which are of wide applicability and seem particularly promising for phase equilibrium computations in chemical engineering thermodynamics and thermodynamics instruction.

The rest of this chapter is organized as follows: first, the general formulation and the solution methods for thermodynamic equilibrium problem are given. Next, a number of predictive (mostly GC-based) methods for the estimation of pure component and mixture properties will be presented. Then, the recently proposed EoS/G^E mixing rules for cubic Equations of State (EoS) will be described in some detail, followed by a discussion of computational aspects in thermodynamic calculations.Three examples showing the capabilities of predictive models in the description of complex mixtures will be given. Finally, together with an overall assessment of existing GC-based models, some guidelines for future research in the area of phase equilibrium thermodynamics are provided.

Phase Equilibria: the Problem and the Solution

The type of problem that phase-equilibrium thermodynamics aims to solve can be formulated as follows: Suppose that two multicomponent phases, α and β, reach an equilibrium state and the temperature T as well as the mole fractions of phase α are given. The problem is then to determine the mole fractions of phase β and the pressure P of the system. Alternatively, the pressure may be known and then the temperature has to be calculated. The number of intensive properties that must be specified to fix the equilibrium state is given by the Gibbs phase rule (Number of independent intensive properties = number of components - number of phases + 2). For example, in the typical case of a two-component two-phase system, the number of intensive properties is two (four properties are involved and two must be specified before the other two are calculated).

The starting point for the solution of every phase-equilibrium problem is provided by classical thermodynamics which requires that when a system containing N components and two phases (α and β) reaches equilibrium (at constant T and P), the total Gibbs free energy assumes its minimum value. It may therefore be stated that the chemical potential or, equivalently, the fugacities of any component i in the two phases must be equal. Thus,

$$f_i^{\alpha} = f_i^{\beta} \qquad i = 1, \ldots, N \tag{1}$$

where f_i^{α} and f_i^{β} are the fugacities of component i in the two phases.

In order to solve Eq. (1), we need to relate the fugacities with the quantities which are directly measurable and computable (temperature, pressure, mole fractions) employing auxiliary functions closer to our physical senses instead of the rather abstract variables such as chemical potential or fugacity. In this section, we will demonstrate how Eq. (1) can be more conveniently transformed to the cases involving multicomponent Vapor-Liquid Equilibrium (VLE), Liquid-Liquid Equilibrium (LLE) and Solid-Liquid Equilibrium (SLE).

Vapor-Liquid Equilibrium

Two main approaches are often used in the solution of Eq. (1):

i. The activity coefficient or γ-φ approach (for low pressure calculations).

ii. The Equation of State (EoS) approach (for high pressure calculations).

The γ-φ approach: In the γ-φ approach which is employed at low pressures, the liquid phase is expressed through the activity coefficient (γ) and the standard state fugacity (f_i^o). It is possible to express the standard state fugacity in terms of the fugacity coefficient at saturation (φ^s), the vapor pressure at the system temperature (P^s) and the Poynting effect. Eq. (1) can thus be written as:

$$y_i \Phi_i^V P = x_i \gamma_i P_i^s \Phi_i^s \exp \frac{V_i (P - P_i^s)}{RT} \tag{2}$$

where φ_i^V is the fugacity coefficient of component *i* in the vapor phase, and x_i and y_i are the mole fractions in the liquid and vapor phases, respectively. The exponential term is the Poynting correction, which is often ignored at low to moderate pressures.

The key quantities are the vapor phase fugacity coefficient and the activity coefficient in the liquid phase (especially the latter) and this is why this approach is traditionally called the γ-φ or Low-Pressure approach to VLE calculations. γ is usually calculated using a model suitable only for the liquid phase. This can be a correlative model (e.g., UNIQUAC, NRTL) or a predictive model (e.g., UNIFAC, ASOG). φ_i^V must be calculated from another model (usually a cubic EoS or the virial equation truncated after the second term) or assumed to be equal to one (often a reasonable approximation at low to moderate pressures, except for strongly associating compounds, like organic acids). Provided that the liquid phase is adequately described, complex multicomponent equilibria can be modelled. However, due to the assumptions involved, the γ-φ approach is only applicable to mixtures at relatively low pressures (typically below 10-15 atm.).

The Equation of State approach: At high pressures, the EoS approach is used for VLE computations and Eq. (1) is written as:

$$y_i \Phi_i^V = x_i \Phi_i^L \tag{3}$$

The fugacity coefficients of the liquid and the vapor phases depend on temperature, pressure and composition and using established thermodynamic relationships, they may be calculated from any EoS applicable to both phases. Usually cubic EoS (e.g., SRK, PR) with classical mixing rules and one or two interaction parameters (obtained from fitting experimental VLE data or from generalized correlations) are used for this purpose. Contrary to the γ-φ approach, Eq. (3) applies to both low and high pressures but due to the limitations of most classical EoS and their mixing rules, especially in the description of the liquid phase, the EoS approach was (until recently) used only for mixtures with relatively nonpolar components (mainly systems with hydrocarbons and gases). The need for models which would combine the positive features of the γ-φ and the EoS approaches has led to the development of improved mixing rules for the cubic EoS often called combined EoS/G^E mixing rules, which are described later.

Liquid-Liquid Equilibrium

When two liquid phases α and β are in equilibrium with each other, Eq. (1) can be written as:

$$(x_i \Phi_i^L)^\alpha = (x_i \Phi_i^L)^\beta \quad i = 1, \ldots, N \tag{4}$$

or alternatively

$$(x_i \gamma_i)^\alpha = (x_i \gamma_i)^\beta \quad i = 1, \ldots, N \tag{5}$$

Equations (4) and (5) hold for every component in the mixture and are solved together with the mass balance equations, i.e., the sum of the mole fractions of all the components in each phase must be equal to unity. Equation (4) is used whenever an EoS suitable for the liquid phase is available. Since this is rarely the case, Eq. (5) is often used in conjunction with an activity coefficient model.

Solid-Liquid Equilibrium

In the case of a binary solvent/solute(solid) system and under certain simplifying assumptions, Eq. (1) yields an expression for the solubility (mole fraction) of the solute in the solvent as a function of γ (the liquid activity coefficient of the solute) and the ratio of the fugacities of pure solid and pure, sub-cooled liquid. The ratio of fugacities can be calculated as a function of several thermal properties of the solute and its melting point. In the case of nonideal solutions, the activity coefficient is usually calculated from the regular solution theory or an activity coefficient model. Cubic EoS have also been used (though not very often) for modelling SLE.

Multiphase Equilibria and Need for Properties

Multiphase equilibria problems such as VLLE can be formulated by solving Eq. (2) or (3) together with Eq. (4) or (5). Cubic EoS require accurate values for the critical temperature, the critical pressure and the acentric factor of each compound involved in the calculations. For many heavy and complex compounds, these properties are not known and/or cannot be measured experimentally and consequently, estimation methods need to be used. Other pure component properties (heat of fusion, solid and liquid heat capacities, solubility parameter, liquid volume, vapor pressure) are needed either directly in the calculations (e.g., in Eq. (2)) or indirectly through the models used (e.g., regular solution theory, free-volume models) and often they have to be estimated. Mixture properties are, of course, also needed. The fugacity coefficient (of each equilibrium phase) and, in particular, the liquid phase activity coefficient are key properties for phase equilibrium calculations.

Pure Component Property Estimation and Data Base

In computerized applications, either the experimental data in the form of property constants or parameters for regressed correlations or, parameters for appropriate estimation methods need to be supplied. Property data bases such as the Design Institute for Physical Property Data (DIPPR) data compilation system (Buck and Daubert, 1990), PPDS-2 (Scott, 1987) and the DORTMUND data-bank (Gmehling, 1985) represent one way of supplying such information. Another option is the use of property data bases in process simulators such as HYSIM

(1992), PRO-II (1991) and ASPEN (1982). Both types of data bases include property constants and, usually, temperature dependent parameters for regressed correlations for a very large number of properties and pure compounds. While in the former type of data bases, emphasis is given on information of data source and quality, in the later type of data bases, emphasis is given on estimation of properties (usually by group contribution methods). Estimation methods are needed as it is not always possible to find reliable experimental data for the required properties for the compounds of interest. Information on the data source and data quality however play an important role in the development of the estimation methods. Therefore, both types of databases have importance.

In the area of estimation methods, a recently developed group contribution approach (Constantinou and Gani, 1994 and Constantinou *et al*., 1994) for prediction of ten pure component properties deserves a brief description. With this method, the estimation is performed at two levels. In the first level, contributions from first-order groups (for example, groups from UNIFAC) are utilized while in the second more complex level, a small set of second-order groups are employed. The second-order groups use the first order groups as building blocks. Thus, the method provides both a first order approximation (first order group contributions) and a more accurate second-order prediction. Lists of first and second order groups, parameter tables and functions for ten properties (normal boiling point, normal melting point, critical pressure, critical temperature, critical volume, standard enthalpy of vaporization at 298 K, standard enthalpy of formation at 298 K, standard Gibbs energy, liquid molar volume at 298 K and the acentric factor) are given in the above mentioned publications. Several important features need to be noted. Significant improvement in the prediction accuracy has been achieved, it is possible to distinguish between some isomers, extrapolation is very reliable and finally, all the properties are estimated only from the structural information of the pure compound. Thus, those properties which are only dependent on the structural variables are called primary properties as opposed to secondary properties, which are dependent on other primary or secondary properties. The following simple functional expression is used for all ten properties:

$$f(x) = \sum_i N_i C_i + W \sum_j M_j D_j \tag{6}$$

where C_i and D_j are the first and second order group contributions respectively and N_i, M_j are the number of their occurrences in the compound. The constant W is set equal to unity for second-level estimation.

Models for Predicting Mixture Properties with Emphasis on the GC Approach

In this section, UNIFAC (Fredenslund *et al*., 1977) and related methods for predicting a variety of mixture properties are reviewed followed by a brief discussion on the achievements and the shortcomings of cubic EoS in calculating phase equilibria. Special emphasis is given to the recently developed combined EoS/G^E mixing rules. Finally, one of the most promising EoS/G^E models, the MHV2 model, is described and evaluated.

UNIFAC and Related Models

UNIFAC has been developed by Fredenslund *et al.* (1977) and is a GC activity-coefficient model based on UNIQUAC. It contains two contributions to the activity coefficient, a combinatorial (*comb*) and a residual (*res*) term:

$$\ln\gamma_i = \ln\gamma_i^{comb} + \ln\gamma_i^{res} \tag{7}$$

The combinatorial term, which is a function of the van der Waals volume (*R*), surface area (*Q*) and composition, takes into account the entropy effects arising from differences in size and shape between the components of the mixture. The *R* and *Q* values are estimated using the group-increments given by Bondi, and are readily available in the UNIFAC tables. The residual term is a function of the surface area, the composition, the temperature and the group-energy parameters (a_{mn}). It accounts for the energetic interactions between the functional groups in the mixture. The group parameters of the residual term are usually estimated from experimental low-pressure phase equilibrium (normally VLE) data. Occasionally, other types of experimental data are used in the parameter estimation (e.g., excess enthalpies h^E and heat capacities, infinite dilution activity coefficients, LLE, etc.). Three major UNIFAC models can be identified:

> Original UNIFAC: In the original UNIFAC, γ^{comb} was given by the Staverman-Guggenheim (SG) term, while the group-parameters were assumed to be temperature independent. For this reason, original UNIFAC, although successful for VLE calculations, does not yield quantitative predictions for excess enthalpies and extrapolations above 425 K should be avoided. The latest revision of this model is given by Hansen et al. (1991).
>
> Modified UNIFAC: In the modified UNIFAC model developed in Lyngby by Larsen *et al.* (1987), the empirical combinatorial formula originally suggested by Kikic *et al.* (1980) was adopted. This new combinatorial term performs better than the SG term for alkane mixtures. The residual term is the same as in original UNIFAC, but the group interaction parameters are assumed to be temperature dependent through a logarithmic temperature dependency. Both VLE and h^E data have been used in the parameter estimation.

Modified UNIFAC only offers a marginal improvement over original UNIFAC regarding VLE predictions. However, due to the temperature dependent parameters employed, the prediction of h^E, and hence, the temperature dependence of the activity coefficients is much improved. Thus, unlike the case with original UNIFAC, safe (large) temperature extrapolations (often up to 550-600 K) are possible with modified UNIFAC. An alternative similar modified UNIFAC model has been developed by the Dortmund group (Gmehling, 1986). VLE, h^E and infinite dilution activity coefficients have been used successfully in the parameter estimation, thus making the Dortmund version of modified UNIFAC a very attractive model (particularly suitable for infinite dilution activity coefficient calculations).

> New UNIFAC: This is the latest revision of the UNIFAC model, developed in Lyngby by Hansen *et al.* (1992). It has been found that the Kikic combinatorial formula employed in modified UNIFAC does not extrapolate well to mixtures with polyethylene, and hence it should not be used for mixtures with polymers. On the other hand, the original SG term provides a much better representation of athermal polymer solutions and is, therefore, retained in the new UNIFAC model. The residual term is that of UNIFAC using linear temperature-dependent parameters fitted exclusively to VLE data. The logarithmic term of the temperature dependence is dropped in the new model both because it is not

needed if only VLE data are to be fitted and because it leads, in some cases, to uncertain extrapolations. The performance of new UNIFAC is at least as accurate as the previous versions of UNIFAC for VLE calculations (and often better), while it retains the correct limit for polymer systems. Furthermore, it has been used as the basis for the extension of the UNIFAC model to polymers.

Application of UNIFAC to LLE/SLE Calculations

The modified UNIFAC models, with temperature dependent parameters, predict qualitatively well UCST and closed-loop phase envelopes for some systems over an extended temperature range. However, it has been found that, in general, UNIFAC with VLE-based parameters do not yield quantitative prediction of LLE. For this reason, Magnussen *et al.* (1981) developed a UNIFAC-LLE parameter table (parameters based exclusively on LLE data), which yield quantitative estimations of LLE phase compositions around 25 °C.

Gmehling *et al.* (1978) used UNIFAC for predicting SLE for some solids (e.g., phenanthrene, anthracene) in a variety of single and mixed solvents (alcohols, acids, hydrocarbons, etc.). Eutectic temperatures and compositions were also calculated for some binary systems (benzene/phenol, benzene/acetic acid). Good agreement with experiment is found. In all cases the predictions were much better than when the ideal solution (γ=1) assumption is used.

Use of γ in Transport Properties

Under certain assumptions, the surface tension of a liquid mixture (σ) is related to the surface tension of a pure component *i* (σ_i), the molecular surface area A_i of a pure component *i*, the compositions in the surface (*s*) and the bulk (*b*) liquid phases and the corresponding activity coefficients, $\gamma_{i,s}$ and $\gamma_{i,b}$. Suarez *et al.* (1988) used the modified UNIFAC model of Larsen *et al.* to predict $\gamma_{i,s}$ and $\gamma_{i,b}$, having determined σ_i and A_i from selected experimental surface tension data for binary mixtures. They obtained excellent agreement between experimental and calculated surface tensions.

Wu (1986) developed a widely applicable GC viscosity model by combining Eyring's theory with UNIFAC, but the group interaction parameters were obtained from experimental viscosity data. Recently Cao *et al.* (1993) developed a GC viscosity model with parameters taken directly from original UNIFAC-VLE (Hansen *et al*., 1991). The viscosity predictions for liquid mixtures can be thus carried out without any viscosity information for mixtures and very good agreement is obtained with experimental data. The average deviation is only 4.1% compared to about 13% with the method of Wu.

The GC Approach for Polymers and Electrolytes

Many GC models, both EoS and γ-models, have been proposed in the last ten years for modelling polymer solutions. Two EoS models are hereafter briefly described.

1. The GC version of the Flory EoS (GC-Flory) originally developed by Holten-Andersen *et al.* and recently revised by Bogdanic and Fredenslund (1994).
2. The Group-Contribution Lattice Fluid (GCLF) EoS proposed by High

> and Danner (1990), which is a GC version of the Lattice-EoS proposed earlier by Panayiotou and Vera. GCLF was recently revised (Danner, 1994) to include a GC-based interaction parameter. Both EoS are predictive models and they successfully represent polymer/solvent VLE, but their applicability is somewhat limited by their rather small (compared to UNIFAC) parameter table. Using the same VLE-based parameter table, GC-Flory has been recently shown to predict qualitatively well LLE for binary polymer solutions, including Upper and Lower Critical Solution (UCST, LCST) behavior. In several cases, especially for polar solutions (e.g., octanol/polyethylene), the predictions are quantitatively adequate for design purposes.

Original UNIFAC alone usually underestimates the activity coefficients in polymer solutions, because it ignores the significant free-volume differences between solvents and polymers, which often exist in polymer solutions. Three attempts to remedy this deficiency have been proposed. Oishi and Prausnitz (O-P, 1978) and Iwai and Arai (1989) added a Free-Volume (FV) term in the original UNIFAC model. Kontogeorgis *et al.* (1993) developed a simple γ-model (Entropic-FV) using the residual term of new UNIFAC (with linear temperature dependent parameters) and a combined comb/FV term recently proposed by Elbro *et al.* (1990). All three models make use of existing UNIFAC tables but require the component volumes at the system temperature. When experimental data are not available, both the solvent and polymer densities can be predicted with very good accuracy (1% for most solvents below the normal boiling point and 2.7% for amorphous polymers) using a recently developed GC method for the volume (Elbro *et al.*, 1991).

The O-P UNIFAC-FV and Entropic-FV models have been extensively tested and shown to provide very good predictions for VLE and infinite dilution activity coefficients for a great variety of homopolymer and co-polymer solutions. Entropic-FV is simpler than the O-P model and, unlike the latter, it can be easily extended to multicomponent solutions. Both UNIFAC and Entropic-FV models predict UCST behavior qualitatively well using the VLE-based UNIFAC parameter table. In addition, Entropic-FV is capable of predicting qualitatively well the FV-driven LCST behavior, as well as combined UCST and LCST phase diagrams. From a quantitative point of view, the LLE predictions with Entropic-FV are often rather good, particularly for relatively nonpolar solutions (e.g., polystyrene/cyclohexane).

Simple activity coefficient models have been proposed for predicting the activity coefficient of solvent in liquid mixtures containing ions. Two models have received much attention during the last years: the Chen *et al.* (1982) model and the model proposed by Sander *et al.* (1986). The latter has been simplified recently by Nikolaisen *et al.* (1993). Both models can treat aqueous as well as mixed solvent electrolyte solutions; they use a Debye-Hückel expression to describe the long-range ion-ion interactions. In the Chen model, short-range interactions are described with the NRTL equation, while in Sander's model an extended UNIQUAC equation (with concentration-dependent parameters) is used for the same purpose. In the modification of Sander's model by Nikolaisen *et al.*, only binary (linear temperature dependent) concentration independent parameters are used. Chen and Sander models give good representation of VLE for electrolyte solutions. In particular, the modification by Nikolaisen *et al.* has been found to give good predictions of mineral solubilities in ternary and multicomponent systems.

Use of Cubic EoS for Phase Equilibrium Calculations

Cubic EoS are often used for phase equilibrium calculations for nonpolar (hydrocarbon) or slightly polar (gas/hydrocarbon) systems. Usually the classical van der Waals one-fluid (vdW1f) mixing rules are used together with the geometric mean rule for the cross energy parameter and the arithmetic mean rule for the cross co-volume parameter. In most cases (especially when asymmetric systems are involved, e.g., methane/nC_{36} or ethane/nC_{44}) one or two interaction parameters (in the cross energy terms) are needed. Several generalized correlations have been reported for specific cases (e.g., the correlation by Kordas *et al.* (1994) covering the whole CO_2/hydrocarbon series). The applicability of cubic EoS to complex equilibria (e.g., for water/hydrocarbons) present many difficulties, even if adjustable parameters are used. In these cases, more sophisticated concentration-dependent or density-dependent local composition mixing rules are used, but at the cost of extra complexity in equilibrium calculations and significant increase in computing time. An interesting review on these rules is given by Heidemann and Fredenslund (1989). The need for complex mixing and combining rules stems from the recognition that many of the limitations of cubic EoS are due to the use of vdW1f mixing rules for the EoS parameters, especially for the mixture energy parameter.

In the last years the EoS/G^E approach has become very popular, since it incorporates G^E models (e.g., UNIFAC, NRTL) in the mixing rule for the energy term of cubic EoS in a way which appears to be both computationally efficient and relatively fast and successful in the description of complex equilibria. The EoS/G^E models which use Group-Contribution (GC) G^E models are particularly worth mentioning, since the resulting EoS/G^E models are purely predictive tools for phase equilibrium calculations. The basic principles of the EoS/G^E models and one of the most successful such models (MHV2) are hereafter described.

The EoS/G^E Mixing Rules for Cubic EoS

The basic principle (or starting point) in the development of most EoS/G^E models is to demand that the excess Gibbs energy from a cubic EoS be equal to the excess Gibbs energy from an activity coefficient model (e.g., UNIFAC, Wilson, etc.), i.e.,:

$$\left[\frac{G^E}{RT}\right]_P^{EoS} = \left[\frac{G^E}{RT}\right]_P^{*} \tag{8}$$

where the superscript * refers to the activity coefficient model used. The subscript P denotes that the equality of equation (8) is valid at a certain pressure, which is called "reference pressure". Vidal (1978) developed an EoS/G^E model assuming infinite reference pressure. This match, however, renders it impossible to use G^E-parameters based on low-pressure VLE data, like those in the GC tables of UNIFAC or ASOG or the NRTL (and other liquid solution models) parameters published by Dechema.

The first combination which led to simple and purely predictive EoS/G^E models was the MHV1 and MHV2 mixing rules (Modified Huron-Vidal first and second order; Michelsen, 1990). In these mixing rules, it is assumed that equation (8) is valid at zero pressurizes making it possible to use the existing UNIFAC (and related) tables. In the MHV2 model, as usually used, the SRK EoS is combined with the modified UNIFAC model (Larsen *et al.*, 1987), but the approach is readily applied to any EoS and G^E model.

An EoS/G^E model without specific reference pressure (LCVM model; Boukouvalas *et al.*, 1994), particularly successful for phase equilibrium calculations for asymmetric mixtures (i.e., those with components differing significantly in size), has been recently developed.

A theoretical limitation of the aforementioned models is that they do not obey the quadratic composition dependence of the mixture second virial coefficient, required by statistical mechanics. Although this limitation seems not to have serious impact on the accuracy of equilibrium calculations, Wong and Sandler (1992) recently proposed an alternative combination approach (where equation (8) is valid for the excess Helmholtz free energy instead of G^E) at infinite pressure, which provides the theoretically correct composition dependence for the mixture second virial coefficient.

The MHV2 Model

Let us assume the SRK EoS. Taking the limit of EoS at zero pressure in combination with the basic equation (8), the following mixing rule can be easily derived for the reduced dimensionless energy parameter (=*a/bRT*):

$$q^{ex}(\alpha) = \left[\frac{G^E}{RT}\right]^* + \sum_i x_i q^{ex}(\alpha_i) + \sum_i x_i \ln\left(\frac{b}{b_i}\right) \tag{9}$$

where $q^{ex}(\alpha)$ is the function:

$$\ln(fo/RT) + \ln b = -1 - \ln(u_o - 1) - a \ln\left(\frac{u_o + 1}{u_o}\right) = q^{ex}(a) \tag{10}$$

where f_o is the fugacity at zero pressure. u_o (=V_o/b) is the reduced volume at zero pressure, obtained by solving the SRK EoS. It is defined only when $\alpha > \alpha_{lim} = 3 + 2\sqrt{2}$ .Equation (9) is the basic equation for the exact modified Huron-Vidal mixing rule, as originally proposed by Michelsen (1990). In principle, any mixing rule can be used for *b*, although usually the linear rule is used. On the right-hand side of equation (9), UNIFAC or any other G^E model can be used. Equation (9) ensures that the cubic EoS reproduces exactly the G^E model it is combined with. Equation (9) has to be solved iteratively for the energy parameter. An additional problem of the exact model is that it is not valid for $\alpha < \alpha_{lim}$. Such low α values correspond to systems where the reduced temperature of one of the components is higher than approximately 0.9, for example, mixtures with a gas and a heavy hydrocarbon, where the gas is in supercritical condition. These problems of the exact model find a solution with Michelsen's observation that $q(\alpha)$ is almost a linear function of α, i.e., $q(\alpha) \approx q_o + q_1\alpha$. When this relation is inserted in equation (9), an explicit mixing rule for α, the so-called MHV1 mixing rule is obtained.

A much better reproduction of the G^E model is obtained with a quadratic $q(\alpha)$ function, i.e., $q(\alpha) \approx q_o + q_1\alpha + q_2\alpha^2$. The resulting mixing rule is called MHV2 (and is no more explicit with respect to α):

$$q_1\left(\alpha - \sum_i x_i \alpha_i\right) + q_2\left(\alpha^2 - \sum_i x_i \alpha_i^2\right) = \left(\frac{G^E}{RT}\right)^* + \sum_i x_i \ln\left(\frac{b}{b_i}\right) \tag{11}$$

q_1 and q_2 are constants depending on the EoS used and they are fitted in the interval $10 < \alpha < 13$.

Using equation (11), MHV2 model can also be used for systems with supercritical components. In order to do so in a predictive way, Dahl *et al.* (1991) added 13 new groups involving gases in the modified UNIFAC parameter table. The basic results with MHV2 can be summarized as follows:

1. Low pressure VLE is predicted as accurately as using UNIFAC and the γ-φ approach.
2. VLE at high pressures and temperatures is also predicted well, even for binary and ternary systems involving highly polar compounds (Dahl and Michelsen, 1990). For example, the following overall errors in Bubble Point Pressure calculations have been reported: acetone/water 0.8% up to 60 bar, water/ethanol 2.3% up to 186 bar, methanol/benzene 2.7% up to 58 bar, acetone/methanol/water 8% up to 82 bar.
3. Binary and ternary gas solubilities at high temperatures and pressures are predicted very well, even for systems containing methanol, water and gases. In particular, MHV2 predicts reasonably well the three phase equilibria for carbon dioxide/ethanol/water and water/2-butanol/butane, systems which are important in process design and for supercritical extraction processes. Furthermore, solubilities in multicomponent systems are predicted (based only on binary data) better than when more complex density-dependent mixing rules are used. This makes the EoS/G^E approach a very attractive and predictive alternative to these complex mixing rules.
4. The one limitation of MHV2 is that it cannot describe accurately phase equilibria for asymmetric systems, e.g., carbon dioxide or ethane with heavy hydrocarbons and carbon dioxide with fatty acids and esters. This deficiency may be partly explained by the fact that, in these cases, the quadratic $q(\alpha)$ function employed in MHV2 (or any other approximate $q(\alpha)$ function) cannot guarantee full reproduction of the G^E model implemented. Note that the Wong-Sandler mixing rule also fails to reproduce the G^E model used and, thus, it yields unsatisfactory results for asymmetric systems, like 1-hexanol/water (Coutsikos *et al.*, 1994). A quadratic composition dependence in the mixture co-volume parameter in conjunction with the MHV1 mixing rule (Coniglio *et al.*, 1994) or a different EoS/G^E approach, like the empirical LCVM model (Boukouvalas *et al.*, 1994), have been shown to represent successfully asymmetric systems, preserving at the same time the EoS/G^E solution to this problem. In particular, the LCVM model yields very good predictions for the activity coefficients in asymmetric alkane systems and reasonable estimates for the solubilities of solid aromatic hydrocarbons in supercritical carbon dioxide.

Computational Aspects and Examples of Application

Among the many different models and methods applicable to a particular thermodynamic property, it can be argued that some are more suitable than others (O’Connell, 1983). Even the most popular models have differing levels of accuracy, reliability, computational efficiency and parameter development. Also, purposes for which property values are used can vary.

Therefore, in this section, some aspects related to the appropriate selection of the thermodynamic model for a specific problem, the correct and efficient utilization of the selected model and finally, the validity of the model parameters are discussed.

Inaccuracies in design calculations can often be traced to inaccurate prediction of properties, which in turn, can be traced (in some cases) to the wrong selection of thermodynamic models. For example, the UNIFAC-LLE model should not be used for VLE calculations even though this model has the available group interaction parameters while the WILSON model cannot be used for liquid-liquid phase split computations. Also, the range of validity of the UNIFAC-LLE model parameters is from 282 K to 313 K. Cox (1993) has pointed out some of the dangers of using models without considering the implications (for example, some of the methods for predicting the behavior of a surfactant molecule will yield incorrect predictions if water is present). This indicates that the mixture (or species) for which the property is to be estimated, needs to be carefully analysed. From the models presented in the above sections, it is obvious that aqueous mixtures of alcohols, polymers and salts need different models for the estimation of the same properties. Also, the sensitivity of the model parameters may vary for the same property and mixture (or species). The solution of Eq. 3 may become very sensitive to numerical errors near the critical point.

It is also necessary to be aware of the range of validity of the model parameters and how the model parameters were determined. The extrapolation capability of the model needs to be carefully evaluated for untried systems and conditions outside the range of the experimental data (used for parameter estimation). If the confidence region of the parameters is very small, even interpolation of the model parameters can be risky.

Finally, the problem of unavailable parameters needs to be discussed. In many cases, the experimental phase equilibrium data are available at conditions which do not coincide with the conditions maintained in plant operation. Also, in many cases, a subset of the model parameters may not be available. Can the existing model parameters be used or should new parameters be estimated? If experimental data are not available, how can the new parameters be estimated? What happens if the mixture-related parameters are available but some pure component parameters are not available? Some questions users may need to ask are: How should the pure component property parameters be estimated? Using the mixture properties? General answers to these question unfortunately do not exist because different models behave differently even for the same property prediction problem (but for different mixtures). Thus, it is important to analyze the model parameter information and study the sensitivity of the model parameters before attempting to find a method of solution. Thermodynamic consistency should also be considered during the validation of computed property values.

Three application examples using some of the models described in the section on mixture properties are presented in the form of computed phase diagrams, since they play an important role in the design and synthesis of chemical and biochemical processes. The first example (Fig.1) concerns the prediction of UCST, LCST and hourglass phase diagrams with the Entropic-FV model for the system polystyrene/acetone. It can be seen that Entropic-FV predicts qualitatively well UCST and hourglass curves and quantitatively well LCST phase envelopes.

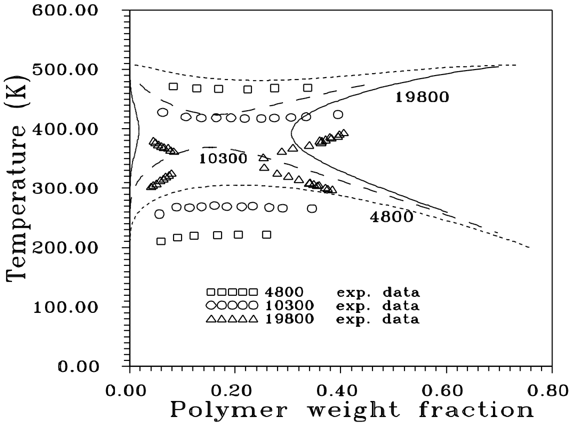


Figure 1. Experimental and predicted LLE phase diagram for polystyrene/acetone (at three polymer molecular weights) with Entropic-FV. The points are the experimental data and the lines are the model's predictions.

The second example (Fig.2) concerns the application of the aqueous electrolyte model by Nicolaisen *et al*. for the determination of the phase diagram (salt precipitation curves) for the system SO_4^{2-}, Cl^-, Na^+, K^+/water. The model yields satisfactory predictions of multicomponent equilibria.

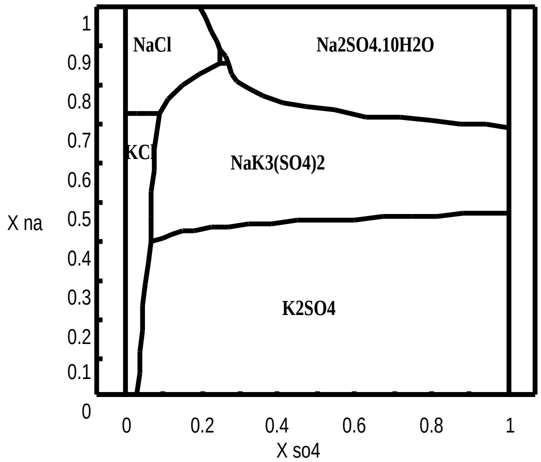


Figure 2. Quaternary SLE diagram on a water-free basis for the system SO_4^{2-}, Cl^-, Na^+, K^+ /water at 289.09 K calculated with the model of Nicolaisen et al. (The experimental data, not shown in the figure, almost coincide with model's predictions).

The third example (Fig. 3) concerns the use of the MHV2 model for the prediction of VLLE for the system propane-ethanol-water at P = 35 atm. It can be seen that the ethanol-water azeotrope has disappeared at this pressure. Also, that the vapor phase is very close to the liquid-liquid phase boundary.

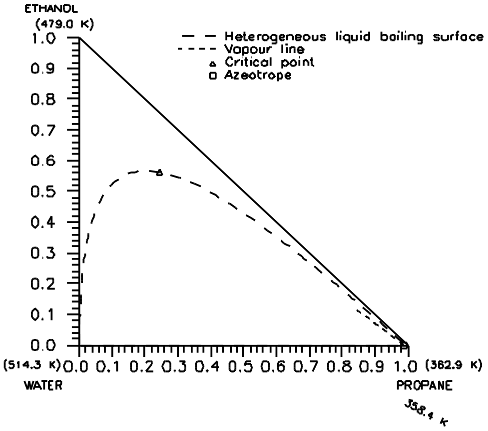


Figure 3. Prediction of VLLE for the ternary system propane-ethanol-water at P=35 atm. using the MHV2 model.

Conclusions and Future Directions

The need for reliable data and predictive methods for the estimation of pure component and mixture properties for phase equilibrium calculations is illustrated. Several models for solving the phase equilibrium equations have been briefly presented. Special attention is given to the GC approach and related models. In particular, the various versions of UNIFAC and the MHV2 mixing rule are discussed in detail and both their achievements and limitations are clearly stated.

It can be concluded that:

1. It seems difficult to use a single model for solving many different phase equilibrium problems and for a variety of mixtures.
2. The GC approach is a powerful technique for preliminary design calculations, but accurate or reliable predictions cannot be always obtained, while there are several additional problems (e.g., LLE calculations at high temperatures, mixtures with isomers).
3. Difficult systems (e.g., mixtures with polymers, electrolyte solutions, high pressure complex equilibria) can be described reasonably well with relatively simple models (similar to UNIQUAC/UNIFAC), often based on the principles of the GC approach.
4. The recent EoS/G^E mixing rules deserve special attention. These mixing rules allow cubic EoS to be used for describing highly non-ideal sys-

> tems, or equivalently extend the applicability of classical low pressure activity coefficient models like UNIFAC to high pressure applications including gas solubilities. Future work on these mixing rules is necessary and should focus on improved understanding of their capabilities as well as to extend their applicability to difficult systems (e.g., mixtures with associating fluids, electrolytes and polymers, including polymer blends, high pressure LLE, solid/supercritical fluid equilibria).

Future work should further focus on solving problems related to the biotechnology and food industries, environmental applications, new materials (e.g., co-polymer blends), as well as quantitative representation of condensed phase equilibria (e.g., LLE, solid solutions). In those cases where experimental data (needed in the design of processes and models) are not available, Molecular Simulation (MS) techniques might be particularly useful in the future. MS may provide guidelines for the development of better models or help in the estimation of parameters for equilibrium models using only pure component data (e.g., see Jonsdottir *et al.*, 1994). MS will also help in the investigation of the extrapolation performance of existing models in areas where experimental measurements are scare, very difficult or expensive (e.g., critical properties of heavy compounds) or even it may help in explaining discrepancies between different sets of experimental data (e.g., see the work of Siepmann *et al.*, 1993 on the critical density of heavy alkanes).

Finally, a note regarding the estimation of pure component properties, which are often of great importance for phase equilibrium calculations and product design, needs to be made. For many compounds related to industrially important applications these properties are not available experimentally and must be estimated, usually by GC methods. Novel estimation methods should focus not only on the description of classical critical properties, but also on temperature dependent properties (e.g., vapor pressures, second virial coefficient) and, if possible, their applicability should be extended in the estimation of a number of important non-equilibrium properties (e.g., surface tension, viscosity, diffusion coefficient, glass transition temperature). Finally, it will be important to see if these novel GC methods will cover all the molecular weight range for which experimental data are available, i.e., from low molecular weight solvents up to oligomers and polymers.

References

Aspen Plus (1982). Technical reference manual (Aspen Technology Inc, Cambridge, Massachusetts, USA).

Bogdanic, G. and Aa. Fredenslund (1994). Revision of the Group-Contribution Flory Equation of state for phase equilibria calculations in mixtures with polymers. 1. Prediction of Vapor-Liquid Equilibria for polymer solutions. *Ind. Eng. Chem. Res.*, **33**(5), 1331-1340.

Boukouvalas, C., N. Spiliotis, P. Coutsikos, N. Tzouvaras, and D.Tassios (1994). Prediction of vapor-liquid equilibrium with the LCVM model: a linear combination of the Vidal and Michelsen mixing rules coupled with the original UNIFAC and the t-mPR equation of state. *Fluid Phase Equilibria*, **92**, 75-106.

Buck, E., and T. E. Daubert (1990). Project 801: The DIPPR data compilation project In T. B. Selover (Ed.), *Design institute for physical property Data: Ten years of accomplishment, AIChE Sympo. series,* 275, **86**, 1-4.

Cox, K. R. (1993). Physical property needs in industry. *Fluid Phase Equilibria,* **82**, 15-26.

Cao, W., K. Knudsen, Aa. Fredenslund, and P. Rasmussen (1993). Group-Contribution viscosity predictions of liquid mixtures using UNIFAC-VLE parameters. *Ind. Eng. Chem. Res.*, **32**(9), 2088-2092.

Chen, C.C., H.I.Britt, J.F.Boston, and L.B. Evans (1982). Local composition model for excess Gibbs energy of electrolyte systems. *AIChE J.*, **28**(4), 588-596.

Coniglio, L., K. Knudsen and R. Gani (1994). Model prediction of supercritical fluid-liquid equilibria for carbon dioxide and fish oil related compounds. Submitted to *Ind. Eng. Chem. Res.*

Constantinou, L, and R. Gani (1994). A new group-contribution method for the estimation of properties of pure compounds. Accepted for publication in *AIChE J*.

Constantinou, L, R. Gani, and J. P. O'Connell (1994). Estimation of the acentric factor and the liquid molar volume at 298K through a new group contribution method. Submitted to *Fluid Phase Equilibria.*

Coutsikos, P., N.S. Kalospiros, and D. Tassios (1994). Capabilities and limitations of the Wong-Sandler mixing rules. Submitted for publication.

Dahl, S., Aa. Fredenslund, and P. Rasmussen (1991). The MHV2 model: a UNIFAC-based equation of state model for prediction of gas solubility and vapor-liquid equilibria at low and high pressures. *Ind. Eng. Chem. Res.*, **30**, 1936-1945.

Dahl, S., and M.L. Michelsen (1990). High-pressure vapor-liquid equilibrium with a UNIFAC-based equation of state. *AIChE J.*, **36**(12), 1829-1836.

Danner, R. (1994). Paper presented at the 14th European Seminar on Applied Thermodynamics, Athens, Greece (June 9-12).

Elbro, H.S., Fredenslund, Aa., and Rasmussen, P. (1990). A new simple equation for the prediction of solvent activities in polymer solutions. *Macromolecules*, **23**(1), 4707-4714.

Elbro, H.S., Aa. Fredenslund, and P. Rasmussen (1991). Group contribution method for the prediction of liquid densities as a function of temperature for solvents, oligomers and polymers. *Ind. Eng. Chem. Res.*, **30**(12), 2576-2582.

Fredenslund, Aa., J. Gmehling, and P. Rasmussen (1977). Elsevier (Ed.), *Vapor-Liquid Equilibria using UNIFAC*, Amsterdam pp. 1-380.

Gmehling, J. (1985). Dortmund Data Bank-Basis for the development of prediction methods, CODATA Bulletin 58.

Gmehling, J. (1986). Group contribution methods for the estimation of activity coefficients. *Fluid Phase Equilibria*, **30**, 119-134.

Gmehling, J., T. Anderson, and J.M. Prausnitz (1978). Solid-Liquid Equilibria using UNIFAC. *Ind. Eng. Chem. Fundam.*, **17**(4), 269-273.

Hansen, H.K., P. Rasmussen, Aa. Fredenslund, M. Schiller, and J. Gmehling (1991). Vapor-Liquid Equilibria by UNIFAC Group-Contribution. 5. Revision and Extension. *Ind. Eng. Chem. Res.*, **30** (10), 2352-2355.

Hansen, H.K., B. Coto, and B. Kuhlmann (1992). UNIFAC with linearly temperature dependent group-interaction parameters. *Internal publication of IVC-SEP center No. 9212*, Institut for Kemiteknik, The Technical University of Denmark.

Heidemann, R.A., and Aa. Fredenslund (1989). Vapor-liquid equilibria in complex mixtures. *Chem. Eng. Res. Des.*, **67**, 145-158.

High, M.S., and R.P. Danner (1990). Application of the group contribution lattice-fluid EoS to polymer solutions. *AIChE J.*, **36**(11), 1625-1632.

Hysim Manual (1992), Hyprotech Ltd., Canada.

Iwai, Y. and Arai, Y. (1989). Measurement and prediction of Solubilities of hydrocarbon vapors in molten polymers, *J. Chem. Eng. Japan*, **22**(2), 155-161.

Jonsdottir, SO., K. Reassumes, and AA. Fredenslund (1994). UNIQUAC-parameters determined by Molecular Mechanics. Accepted for publication by *Fluid Phase Equilibria.*

Kontogeorgis, G.M., Aa. Fredenslund, and D. Tassios (1993). Simple activity coefficient model for the prediction of solvent activities in polymer solutions. *Ind. Eng. Chem. Res.*, **32**, 362-372.

Kordas, A., K. Tsoutsouras, S. Stamataki, and D. Tassios (1994). A generalized correlation for the interaction coefficients of CO_2-hydrocarbon binary mixtures. *Fluid Phase Equilibria*, **93**, 141-166.

Kikic, I., Alessi, P., Fredenslund, Aa., and P. Rasmussen (1980). On the combinatorial part of the UNIFAC and UNIQUACK models, *Can. J. Chem. Eng.*, **58**, 253-258.

Larsen, B.L., P. Rasmussen, and Aa. Fredenslund (1987). A modified UNIFAC group-contribution model for prediction of phase equilibria and heats of mixing. *Ind. Eng. Chem. Re.*, **26**, 2274-2286.

Magnussen, T., P. Rasumssen, and Aa. Fredenslund (1981). UNIFAC Parameter table for prediction of liquid-liquid equilibria. *Ind. Eng. Chem. Proc. Des. Dev.*, **20**, 331-339.

Michelsen, M.L. (1990). A modified Huron-Vidal mixing rule for cubic equations of state. *Fluid Phase Equilibria*, **60**, 213-219.

Nicolaisen, H., P. Rasmussen, and J. M. Sørensen (1993). Correlation and prediction of mineral solubilities in the reprocal salt system $(Na^+,K^+)(Cl^-,SO_4^{2-})$-$H_2O$ at 0-100 °C. *Chem. Eng. Sci.*, **48**(18), 3149-3158.

O'Connell, J.P. (1983). Structure of thermodynamics in process calculations. Proc. 2nd Int. Conf. FOCAPD, CACHE, Snowmass, CO, June 19-24.

Oishi, T. and J.M. Prausnitz (1978). Estimation of solvent activities in polymer solutions using a group-contribution method. *Ind. Eng. Chem. Process Des. Dev.*, **17**, 333-339.

Pro/II Manual (1991), Simulation Sciences Inc., USA, version 3.02.

Sander, B., Aa. Fredenslund, and P. Rasmussen (1986). Calculation of vapor-liquid equilibria in mixed solvent/salt systems using an extended UNIQUAC equation. *Chem. Eng. Sci.*, **41**(5), 1171-1183.

Scott, A. C. (1987). PPDS, the U.K. Physical Properties Data Service for the chemical and process industries. In R. Jowitt (Ed.), *Physical properties of foods*, Vol. 2, Elsevier, London, pp. 503-521.

Siepmann, J.I., S. Karaborni, and B. Smit (1993). Simulating the critical behavior of complex fluids. *Nature*, **365**, 330-332.

Suarez, J.T., C.Torres-Marchal, and P. Rasmussen (1989). Prediction of surface tensions of nonelectrolyte solutions. *Chem. Eng. Sci.*, **44**(3), 782-786.

Vidal, J. (1978). Mixing rules and excess properties in cubic equations of state. *Chem. Eng. Sci.*, **33**, 787-791.

Wu, D.T. (1986). Prediction of viscosities of liquid mixtures by a group contribution method. *Fluid Phase Equilibria*, **30**, 149-156.

Zeck S., and D. Wolf (1993). Requirements of thermodynamic data in the chemical industry. *Fluid Phase Equilibria*, **82**, 27-38.