LEADING UNDERGRADUATES ALONG STRUCTURED PATHS TO THE BUILDING OF GOOD PROCESS MODELS

Alan S. Foss[1], George Stephanopoulos[2]
[1]Department of Chemical Engineering, University of California, Berkeley, CA 94720

[2]Department of Chemical Engineering
Massachusetts Institute of Technology, Cambridge, MA 02139

ABSTRACT

Students are led to crafting a process model before writing any equations. This is accomplished by leading them through a structured modeling methodology with which the physics and phenomena of the process are identified and engineering science concepts placed into a model structure simply by declaration. Such declarations are made through use of our new software that assembles the phenomena declared, builds the equations, and solves the equations numerically. The software is novel and unique; ModelLA is its name. With a functioning model, students can examine its characteristics and use such quantitative information to solve the engineering problem posed.

Following such an encounter with the cause and effect among variables, students are much better prepared than they were at the outset to write equations for the model. Through Q and A in a workshop session, the instructor leads the students again through the modeling methodology but this time challenging the students to formulate the equations. The opportunity is present at this stage to elaborate on the structure of the equations (e.g. linear or nonlinear algebraic, ODE, PDE) and effective numerical methods for their solution.

Through such an organization and software assistance, students are propelled quickly to a solution of the engineering problem, bypassing for the moment what often is viewed as insurmountable hurdles in writing and solving equations.

Proclaiming our approach

How would you like your students to craft quantitative models of the turbine, the condenser, and the cooling tower of Figure 1 *without writing any equations?* And then have them link the models together to investigate how to make the most electric power from Hades'supply of geothermal steam?

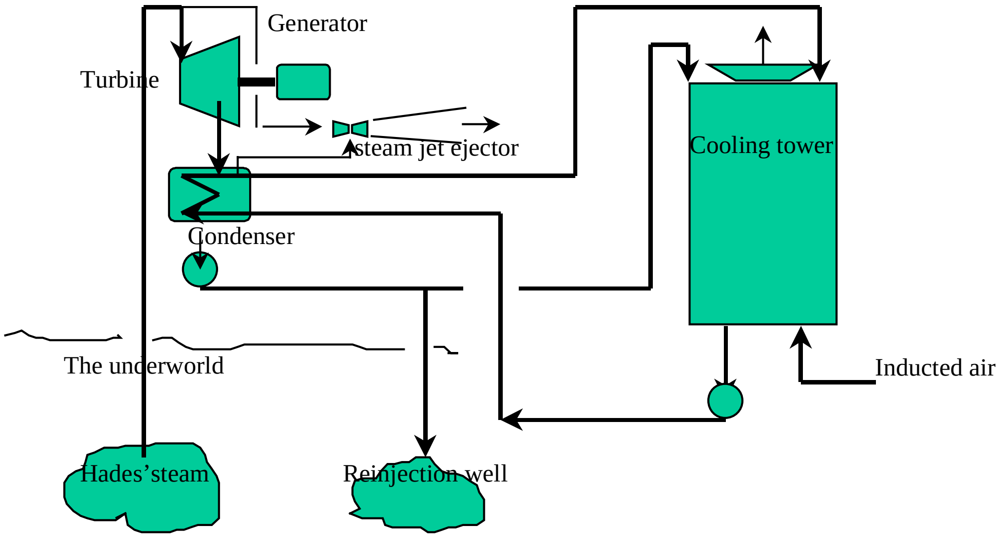

Figure 1 Generating electric power from geothermal steam

Our purpose here is to describe how to do this and the pedagogical value of doing it this way.

The essence of our message about modeling instruction is:

- Make sure that the students understand the physics and chemistry.
- Skip the equations (for now).
- Have the students develop a good model following a methodological modeling framework.
- Have the students use the model to solve the engineering problem.
- Write the equations later.

How do we do this?

The crafting of a model is accomplished through the use of ModelLA[1] , unique modeling software of our own design with which engineering science concepts can be incorporated in a model simply by declaring them in the language of chemical engineering. The software presents a methodological framework for model building. No equations are demanded of the students at this stage. Once the declarations are complete, the software writes the equations and solves them. The focus of the software is *modeling*, not process simulation as in Aspen or Chemcad. We describe the software□s contribution to teaching modeling in a later section.

Why build a model?

Models are at the heart of problem solving in engineering and the sciences, as forcefully stated by Middleman [2] in the preface of his recent textbook. We need models to inform us about the characteristics or behavior of a process, information needed in making decisions about designs and operations. For example, one has to decide on the turbine discharge pressure, whether it can be achieved with the cooling water from the cooling tower, whether a high relative humidity of the air supplied to the cooling tower will have a serious impact on the power production, and whether the degree of condensation in the turbine discharge is within the turbine design specifications. To answer these questions, a model has to be interrogated from several different directions, and the modeler has to craft a model with enough capability and flexibility to meet the demands of such an interrogation. Thus, a process model assumes a central position in problem solving, bringing to that activity a structured approach to achieving specifications and satisfaction of constraints.

What is a good model?

Good models contain just enough detail to represent the important physical and chemical phenomena in a process. To be efficient, models should be directed to delivering information needed for the resolution of a particular problem. We seek to guide students in incorporating engineering science concepts in their models, and we thus consider a model good if those concepts are incorporated where they should be and are soundly represented in terms of the variables of the process. The model of the turbine, for example, need not be concerned with details of the steam flow over the turbine blades; it simply needs to represent turbine power delivered as a function of the steam supply rate , the pressure difference, the degree of condensation of the steam, and the effect of an increase of entropy of the steam in passing through the turbine. A condenser model need only represent the temperature of the coolant, the phase equilibrium in the shell as a function of temperature or pressure, and the rate of heat exchange with the coolant stream. The cooling tower model, however, needs a bit more detail, incorporating the fundamental processes of heat and mass exchange at finite rates between phases.

What do we consider to be a structured path?

There is a logical and intuitive hierarchy of engineering science concepts that engineers place one by one in creating the framework and flesh of a process model, concepts such as the need to define a model envelope (or control volume), the conservation principles, fluxes of energy and mass between phases, chemical reaction phenomena, phase equilibrium phenomena. These phenomena and concepts may be introduced at all levels of modeling detail, from the least detailed to the deepest and highly refined representation of elementary phenomena. A global structured path mirrors that hierarchy and directs the modeler in the logic of incorporating the engineering science concepts into the various levels of the model framework. At each level there are finger-like paths off in several directions along which the student needs to be guided to define in finer and finer detail such matters as an appropriate equations of state, the representation of the activity of species in solution, the character of heat exchange between parts of the process. The route through the maze of these paths often eludes the neophyte (and sometimes old-timers), and it is here where a structured environment is very useful. The software offers to the user a set of structured dialogs along these fingers with which declarations of phenomena can be made.

Why do we need to lead undergraduates along structured paths?

In the display of their logic in modeling, many undergraduates reveal that *they are pathless*. In the first place, few recognize the need for a model in solving an engineering problem. Control volumes are illconsidered, the nature of the model (dynamic or steady state) is undecided, the need to use conservation principles is frequently overlooked. When students are alerted that an energy balance is needed, the balance written does not acknowledge a dynamic process or a spatially distributed process. And when a balance is finally hammered out, the solution of the model equation is full of errors.

Such incompetent performance shows up at all levels and is pandemic, as reported by the extensive research of Woods and coworkers [3]. They and we in our observations see that the students lack a useful framework and a set of modeling and problem solving skills. That has implications for us as instructors and is our motivation for developing the ModelLA software. The software provides a modeling framework that has the potential to banish such dismal performance.

How we make it possible for students to solve problems before writing equations.

(a) The place of the ModelLA software

ModelLA immerses the user in a phenomena - oriented environment. The software accepts user declarations of the physics and phenomena in terms of the concepts of the discipline and in the language of chemical engineering. Its structure is expressly designed to mirror the hierarchically organized set of assertions a modeler would make in crafting a model. The functional elements of the software□s hierarchy corresponding to such assertions are arranged here in a cascade reflecting the modeler□s technical knowledge of engineering science that he or she declares to be operative as the model is defined in more and more detail.

- Control volumes for process unit
  - Control volumes for subunits.
    - Conservation principles
      - Declaration of Species
        - Fluxes among units and subunits.
          - Phases in units and subunits.
            - Equilibrium between phases
              - Reaction stoichiometry.
                - Reaction phenomena in phases.
                  - Satisfying the degrees of freedom

↓
Direction of
increasing detail

A model for the geothermal steam process, for example, would be crafted through multiple calls on these hierarchical elements by:

- Creating on the screen separate control volumes for the turbine, condenser, and cooling tower (and sub-control volumes for the latter two).
- Identifying the global species to be air, water, and carbon dioxide (which comes along with the steam) through access to a chemicals data base.
- Invoking mass and energy balances.
- Placing fluxes between units and subunits and identifying their driving forces.
- Locating regions where phases are in equilibrium and the equations of state for those phases, and
- Selecting some two dozen design variables to satisfy the degrees of freedom.

A significant portion of the relationships among variables and process units can be declared in a graphical manner, an example of which is shown in Figure 2 where graphical declarations have been made of the fluxes of heat and vaporized water between the water and air streams in the cooling tower. In this instance, these constructions are made in a disaggregation of the cooling tower into its

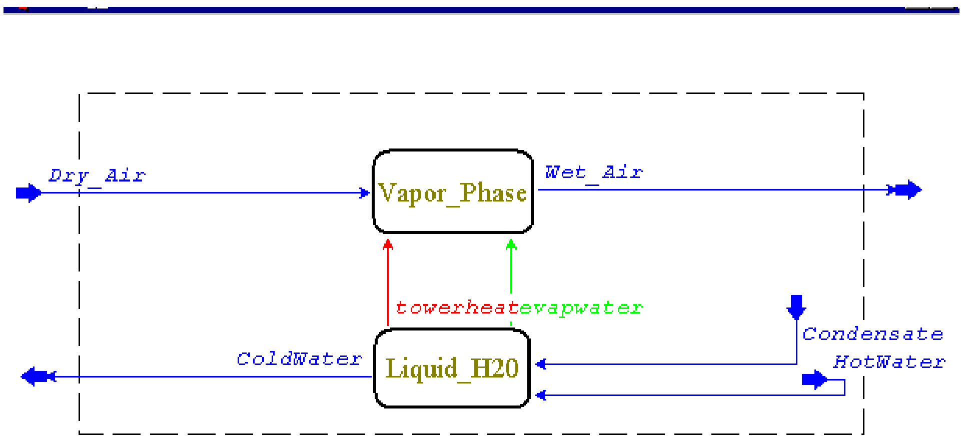

Figure 2 The graphical method of declaring fluxes between subunits of the cooling tower

constituent parts, the air stream and the water stream. The disaggregation capability of modelLA is essential to a modeler’s ability to declare the nature of the physical and chemical phenomena present in the various subparts of process units.

The elements of this hierarchy are not “pre-wired”. Each element is freely accessible at any time. The modeler’s path through them is fully adaptable to the style of the modeler. Forgotten definitions of model attributes may be incorporated by calling up the appropriate level of the hierarchy at any time. Thus, the set of these elements form a *framework* rather than a sequence for defining the attributes of a model.

The consistency of the user’s set of declarations of phenomena, phases, and reactions and a set of design variables satisfying the degrees of freedom is examined and checked by internal procedures and feedback given to the user when inconsistencies are found. Such a check is particularly helpful in satisfying the degrees of freedom because that is where the student’s qualitative understanding of cause and effect in a process is thoroughly challenged and frequently requires rethinking. Through such devices, the software protects the student from accumulating mistakes, shortens model development time, and reduces the need to run to the instructor for help.

At each one of the hierarchy levels, the student faces several decisions about the representation of phenomena, such as the activity of species in solution, the equation of state for vapor and liquid, the driving forces for transport of heat and species between phases as just mentioned, and the reaction rate dependence on temperature and concentration. It is the student’s responsibility to recognize that such decisions must be made. Declarations of the phenomena are made through dialogs at each level. This assemblage of ModelLA’s phenomena-oriented capabilities and its hierarchical organization constitutes the methodological modeling framework we have been developing.

The placing of the responsibility for perceptiveness on the student’s shoulders reflects our view that the most impressionable and enduring learning experience comes about through the student’s proposing and building the model using his or her own insights. Our software thus reflects the constructionist philosophy of learning (as articulated by Hestenes [4]) in which knowledge and the ability to use it are not merely transmitted from instructor to student but must be *reconstructed* by each student individually.

When the declarations of the physics are complete and consistent, ModelLA writes the equations of the model and solves them numerically using gPROMS [5,6], a state-of-the-art numerical solver. Thus, equation writing and solution by the student is bypassed (for the moment).

With a functioning model at hand, students can then investigate process characteristics and extract quantitative information to help them make decisions about process operations such as the effect of an increase in the day’s relative humidity on power production and the several other questions posed at the outset of this paper. With those questions resolved, the engineering problem is considered solved.

However, the pedagogical “loop” about modeling still has to be closed. We must eventually challenge students to write the equations. Otherwise, the students are left without confirmation and confidence that they can execute an analysis and a mathematical model of the process, a capability they will need in their career as an engineer. We describe equation-writing sessions in the next section. In working with a small group of students, we have found that they enthusiastically wanted to write their own model equations and to test their model against ModelLA's results.

b) Iteractive workshop on writing equations

Interactive Q&A is the best way to get the equations written. Even though the students have crafted a working model with ModelLA, they often still need instructor assistance in clarifying process behavior and translating that into equations. The cause and effect relationships that students observed with the use of their model provide concrete examples of the phenomena that ought to be incorporated in the model. Those observations of cause and effect further arm the instructor in illustrating the logic of the steps of their inclusion. The problem solving techniques of Woods *et.al* [3] can be used profitably here when coordinated with that background. The instructor is essential at every stage of these workshops because there are often loose ends about equations that have to be clarified before the students are fully confident of their model.

The equations eventually get written by the students. Numerical solutions can be obtained by general purpose numerical solvers such as Matlab, Mathcad, Polymath, or by paper and pencil and the results compared with the ModelLA solution.

In such an environment, the instructor is further presented with a rare opportunity to describe the equation structure of the model. Nearly all of the material balance problems in the popular texts result in a set of linear algebraic equations, thus offering the instructor an authentically motivated reason to describe the process of Gaussian elimination in the solution of such problems. Numerical methods such as the Newton-Raphson method for nonlinear algebraic equations and the Euler and Runge-Kutta numerical integration methods may also be added to the student's store of equation solving methods as the opportunities arise. In advanced courses, one may delve even deeper into the underlying structure of linear dynamic models, pointing out the presence of dynamic modes and their relation to the eigenvalues and eigenvectors of the coefficient matrix of the state vector.

A further benefit of the workshop is the opportunity for the instructor to illustrate how the use of a process model contributes to an orderly framework for the resolution of engineering problems.

Experiments with the software and students

We have made some experimental evaluations of the effectiveness of the use of the software in supporting our approach to modeling. With a group of four junior-year students at Berkeley, we have observed that the rapidity with which they could craft a workable model amazed even themselves. The ease with which engineering science concepts could be identified and incorporated in their model was a major contribution to their speediness. They were, however, brought up short at times by model-structure consistency checks of their declarations relating to thermodynamics and energy balances. That, of course, we consider a strength of the software because misapprehensions about physical phenomena are common and need to be caught before unfruitful effort is expended. The students also encountered misapprehensions about satisfying the degrees of freedom with design variables, particularly when mass and energy flows interacted. That also is valuable feedback to the student.

This group, without prompting from the instructor, volunteered to write equations for their models just to check whether they understood the model they had built with ModelLA. They solved their equations using Matlab or Mathcad and sometimes paper and pencil. Usually their computed results matched those of ModelLA, but occasionally they did not, at which point the students had to reexamine their understanding of the physics and their representation of it. Their performance in this experiment confirms the effectiveness of the discipline of the model building methodology captured in the software. It also confirms our contention that writing equations can profitably be deferred to follow realization of the model.

A class of 38 senior-year students at MIT used the software in a project involving the synthesis and design of a moderately large chemical process. The process was decomposed through use of ModelLA□s disaggregation capability into a feed preparation section, a reaction section, and a separation section. The subunits needed to accomplish the processing tasks of each section were then the subject of synthesis and design by the students,. Modeling of process units was involved at this level. The students found that the multilevel modeling capabilities of the software permitted an efficient means of distributing modeling sub tasks among members of a team. Student evaluation of the software□s modeling capabilities was overwhelmingly positive. All appreciated being released from equation writing, thus freeing them to focus on synthesis matters. They valued being provided with directed paths to the creation of a model.

These two experiments in groups of different size and of different focus has given us evidence that there are definite benefits to complementing modeling instruction with software that offers a methodological modeling framework.

The essence of the pedagogical merits of our approach to modeling instruction

□ The focus is on physical and chemical phenomena and the use of engineering science concepts.

□ Models are crafted without writing equations, thus skirting a stumbling block that frequently thwarts student progress in modeling.

□ Understanding of cause and effect among process variables is easily acquired through investigation of model behavior.

□ Engineering problems are solved straight away with use of the quantitative relationships obtained from the model.

□ Equations are written after the problem has been resolved, the students thus benefiting from physical insight gained by modeling.

- Students are guided in a hierarchically structured modeling methodology through the use of the ModelLA program and reinforcement by the instructor.

Initiatives sought in approaches to instruction

The use of process models as a generalizing approach to the analysis of process behavior and the resolution of design and operating problems is not emphasized (and sometimes not even mentioned) in our present mode of teaching. Despite the extensive amount of equation writing we ask of our students there has been no thrust in our instruction or textbooks toward formulating from those equations an identifiable structural entity that represents the character of the process useable for process analysis. Instead, the equations are just a bundle of equations that students attempt to juggle, praying that the equations are complete and consistent. That lack of structure in process analysis is a major reason for student shortcomings in process modeling and our motivation in developing the ModelLA software.

There is a pressing need for placing a process model as a central element in process analysis and problem solving because it introduces structure into the method of incorporating information about the process and provides an organizing framework to the activities of problem solving.

Thus, we see that our discipline needs to advance two initiatives that can contribute to the coherency of the curriculum and that can better the capabilities of students:

- Bringing models into prominence when presenting representations of process behavior and showing how the use of models lends a clarifying logical structure to problem solving.

- Using at all levels of the curriculum a structured methodological approach to model formulation involving a high level of interaction and a constructionist approach to learning such as we have advanced here.

Acknowledgment

This work was supported by the National Science Foundation, Directorate for Education and Human Resources.

Availability of the software

Contingent on continued funding, the software and a set of some 30 course modules will be available to faculty members interested in helping us in evaluation of ModelLA□s contribution to teaching. Send expressions of interest on departmental letterhead to Professor George Stephanopoulos at MIT.

Literature Cited

[1] Foss, A.S., George Stephanopoulos, K.R. Geurts , P.J. Goodeve, D.K. Dahm, J. Bieszczad, A. Koulouris, □A phenomena-oriented environment for teaching process modeling: Novel modeling software and its use in problem solving□. Paper presented at the 1998 Annual AIChE Meeting, Miami Beach, November, 1998.

[2] Middleman, S., □An introduction to mass and heat transfer: principles of analysis and design□. John Wiley, New York 1998.

[3] Woods, D.R., *et.al.*, □Developing Problem Solving Skills: The McMaster Problem Solving Program□, J. Engineering Education, Vol. 86 (2), 75-91 (1997).

[4] Hestenes, D., □Modeling software for learning and doing physics□, in □Thinking Physics for Teaching□, Bernardini, C., editor, pps 25-65, 1995, Plenum Press, New York.

[5] Barton, P.I. and C.C. Pantelides, □Modeling of combined discrete/continuous processes□, AIChE J. 40, 966-979 (1994).

[6] Oh, M. and C.C. Pantelides, □A modeling and simulation language for combined lumped and distributed parameter systems.□ Computers and Chem. Eng. 20, 611-633 (1996)

Biographical Sketch

Alan Foss has conducted senior-year courses and the introductory course in chemical engineering at Berkeley for several years. He has been active in the development of instructional software over the past twenty years. He has conducted research in process control systems and has built software for teaching in that area. He is now Professor Emeritus and continues to work on the development of new software.

Biographical Sketch

George Stephanopoulos is internationally recognized for his research in process systems engineering, addressing process synthesis, design, and control concepts. In recent years he has directed software development for the design and operation of batch processing of chemical products. Prior to his tenure at MIT, he taught at the University of Minnesota and the National Technical University of Athens.