1981

Computer-aided design tools in chemical engineering process design

Arthur W. Westerberg
Carnegie Mellon University

COMPUTER-AIDED DESIGN TOOLS IN CHEMICAL ENGINEERING PROCESS DESIGN

by

Arthur W. Westerberg

DRC-06-23-81
March 1981

COMPUTER-AIDED DESIGN TOOLS IN CHEMICAL ENGINEERING PROCESS DESIGN

by

Arthur W. Westerberg

Invited paper for Special Issue of Proceedings of IEEE on Computer-Aided Design.

Department of Chemical Engineering
Carnegie-Mellon University
Pittsburgh, PA 15213

March 1981

Abstract

This paper reviews the design activity for chemical engineering process design, starting from the earliest step of selecting which products to manufacture and ending with designing the operating procedures for a process plant. At each step we discuss computer-aided design tools which have been or are being developed. Throughout synthesis aids which help make discrete design decisions are contrasted with analysis aids which help to select the proper values for continuous variables.

Computer aids are abundant to aid in process design. The future holds promise for an integrated design tool which will aid the engineer from start to finish in his task.

Introduction

The purpose of this paper is to review the development and use of computer-aids in chemical engineering design, covering both industrial practice and current research activities. The aids to be considered will be for the design of complete chemical or petroleum process systems, each comprising a number of arbitrarily interconnected units. The aids for solving single units will not be stressed.

The earliest stages of design in the chemical industry, where computer aids have been discussed in the literature, is market forecasting. Models of the total basic chemicals industry are being developed as an aid for this step.

Having decided which product to manufacture or biproduct to dispose of, the next step is to select the appropriate chemical reaction routes around which to develop a process design. Aids which can generate alternative chemical reaction routes are well established in the area of generally exotic organic chemistry, with some becoming available which involve the more mundane chemistry needed for the production of basic chemicals that support the chemical industry.

Each plausible reaction route requires one to develop an industrial process which can implement the necessary reactions, separations, heating, cooling, pressure changes etc., to effect the chemical route economically. Here the largest number of aids currently exist or are being developed. "Synthesis" aids exist for helping to invent the structure of the process. These range from mixed integer linear programming aids used extensively by the oil industry for selecting the complex subprocesses to include in a refinery design, to the aids for suggesting the particular equipment needed for a more detailed chemical process design.

The analysis aids which allow one to do simulation and design calculations for a fixed structure are well developed and extensively used, particularly for steady-state (DC. analysis in electrical engineering jargon) calculations. Called "flowsheeting systems," most are based on a single program architecture, one not well suited for many important design calculations. We will discuss the variety of architecture used for these aids to expose their individual advantages and disadvantages. Dynamic simulation, the bread and butter of electrical circuit analysis, is much less used in process design. We will consider some of the efforts here and indicate reasons for the slow development.

The process design resulting from the above design activities describes each piece of equipment functionally — e.g. a pump is needed or a heat exchanger is needed. A crude cost estimate at this time is developed and used by management to help decide whether to continue the design. The next step is a major one and is to develop the list of actual equipment which must be purchased to implement the design. The approach is to develop the list by developing a two-dimensional diagram called a Piping and Instrumentation Diagram (PID) spread over some 40 to 60 large sheets of paper and showing the connectivity of all equipment. In parallel, major equipment items are ordered and detailed designs are initiated for them* Aids here are heavily supported by standard catalogue oriented data bases and by interactive graphics.

The design activity remaining prior to plant construction is to develop a three-dimensional model of the plant, either in plastic or within the computer. This step establishes the relative placement of equipment and the piping needed. It is then used as a blueprint for construction.

The last design step to be considered is the development of operating procedures to run the plant - from start-up to normal operation, from normal shut-down to emergency operation.

Throughout the rest of this paper we will examine the aids known to the author which exist to help in each of the above design steps. A pattern should evolve. Aids are labeled synthesis aids if they help to make discrete decisions such as which equipment items are needed and how they are to be interconnected. Aids are labeled analysis aids if they help to make decisions on the values for continuous variables.

Establishing the Market

The first problem we shall consider is selecting what to produce* This problem is quite different for a firm producing specialty chemicals than for a firm producing large quantities of basic chemicals. In the former case much research and development is usually needed to find safe and different chemicals which do not as yet have a market but will likely have if produced. We will not consider this (type of firm. The firm which makes its living from producing basic chemicals in large quantities has a different question to answer. Its question is to see if new or refined technology might allow it to produce a cl.amical already produced elsewhere but for less. Examples would be for petroleum companies or large chemical companies.

Rudd and coworkers (see Rudd (1975), Stadherr and Rudd (1976, 1978), and Stadherr (1976)) have been actively developing a large linear programming model of the entire United States basic chemicals industry. Each major chemical is included, along with the various possible chemical routes by which technology exists to produce it. Differences among the routes are the production of byproduct chemicals - which may also be basic chemicals and thus have a large market, the use of raw materials - which are often other basic chemicals, the use of energy and so forth. The first use of such a model is to see if the U.S. industry is responding to the marketplace - e.g. are there any obvious errors being made in choice of routes. Also, is the total manufacture of basic chemicals making efficient use of available raw materials?

A second use is predictive in nature. Alternate models of expected availability of raw materials can be tried to see how the industry might shift to accommodate it most efficiently. One could consider the design of new processes where major shifts are indicated. (Not to be overlooked is a

diabolical use which could be made of such a program. A company which is a major supplier of a basic chemical could assess the impact on a competitor if it chose to stop selling to them* The question would be to see who suffers more.)

Recently Sophos et al. (1980) used the approach to investigate how industry should structure itself to meet three competing objectives: maximize "availability[11] (a thermodynamic concept), minimize lost work and minimize use of materials.

Such models can be used to assess the effects of various pricing strategies and so forth, but, for our purposes here, it is of interest when they suggest the development of a new or modified process and thus trigger the design activity.

The above type of modeling very crudely characterizes complete chemical processes indicating their behavior only in terms of the use of raw materials and energy and the creation of desirable or not desirable other products. It can only suggest the particular raw materials and products to start looking at for a design.

Given the probable raw materials and the desired products, the next step is to develop the alternate reaction sequences which could form the basis of a design. Here one attempts to enumerate and select among what can be an enormous number of alternatives. This step is labeled "reaction path synthesis[11] and is an attempt to [fl]do[M] chemistry on the computer.

The principal developments for this task have occurred in organic chemical synthesis, particularly for rather complex chemical molecules (see, for example, Corey and Jorgensen (1976), Wipte et al. (1977), Hendrickson (1976), Gasteiger et al. (1974), Gelernter £t al. (1973)). The ideas have been adapted to do chemistry more relevant to the manufacture of major industrial chemicals by Govind and Powers (1977) and by Agnihatri and Motard (1980).

A most difficult aspect of automated chemistry is establishing a criterion by which to rank order the alternatives. Usually one can assess the thermodynamic feasibility which determines that the reactions proposed can occur and proceed to an acceptable extent. However, no general approach is available to say if they will go at an acceptable rate (3 months is too long). A second difficult aspect is the enormous number of alternatives one can generate. These problems are solved (only in part) by incorporating rules based on experience with similar reactions to help sort out the better possibilities.

By whatever approach, one finally must settle onto a limited number of routes on which to base the design activity to follow, hopefully only one.

Scoping Out the Design

If the design activity is to construct a major facility, such as a refinery, then a set of aids based on mixed integer linear models is again frequently used. A refinery comprises several large subprocesses such as hydrocrackers, reformers, cat crackers, etc. These processes are standard technology for which rather detailed integer/linear input/output models have been developed.

The design question is to select how many of each, their interconnection, and their sizes to include in a given refinery. The input information is the types of crude oils which are likely to be processed and the various product mixes desired. By setting up a "super" mixed integer linear programming (MILP) model incorporating within it all the likely configurations, a design can be selected using existing MILP codes. The * optimization will eliminate unneeded subprocesses and unneeded interconnections. Integer variables allow concave cost functions and many other nonlinearities to be modeled.

This activity is a synthesis activity and one which has been used for years by industry. A recent reference exploring this approach is by Grossmann and Santibanez (1979).

Detailed Synthesis of a Process

However one reaches this step, one finally has selected the likely reaction path(s), the needed input raw materials and the desired products and must now develop the details of the process to effect the needed separation, mixing, heating, cooling, pressure changing, etc., tasks. This is the most commonly reported and taught design task and is called "process design."

The first step in this design activity is to conjecture the structure of the solution, a synthesis activity (see Westerberg (1980) and Nishida, Stephanopoulos and Westerberg (1981) for two reviews of the extensive literature related to this step). Research by Siirola, Powers and Rudd (1971) and by Mahalec and Motard (1977a,b) has led to programs which create entire process configurations. These programs are not successful enough to challenge a skilled design engineer but will outperform the novice.

Much more successful have been the synthesis tools developed for well defined subproblems within the process design activity. In particular synthesis results are industrially significant for the design of the heat integration system comprising a network of heat exchangers. A heat exchanger is a device which permits heat to be transferred through a (metal) wall from a hot stream which is to be cooled to a cold stream which is to be heated. The results available for this problem allow a process design engineer to predict the minimum amount and the type of utility streams needed to heat integrate a process. Utilities are external sources of heating or cooling such as steam or cooling water, and they cost money so one wants to use the minimum amount of them which is possible. One can also predict from topological arguments how many heat exchangers should be in a good design and even make a reasonable estimate

of the investment cost for them — all without developing the structure of the network. With these targets one can compare alternative process designs which are heat integrated without designing the heat integration system. One need design the details of the heat integration system for only the few designs finally selected as candidates. With the targets already known, one can usually discover an excellent heat exchanger network design quickly — and know he has one too. Over 40 articles have appeared on this synthesis problem alone. A detailed review appears in Nishida _et al_ (1981).

A second area of major synthesis activity is to design separation systems. The design task is to separate multicomponent mixtures of chemical species into products, each of which often contain a single specie only. A process known as distillation is one of the most common technologies used — the still of the bootlegger separates ethyl alcohol from water using distillation. Here the useful results are the heuristic design rules which have evolved and which seem to be effective for distillation problems. Unfortunately the general separation system synthesis problem is far from being solved but is the center of much current research activity. A prediction by this author is that the effective tools will be again mixed integer linear programming based, supported with numerous heuristics to select the "super[11] process and an evolutionary strategy to add alternatives where the current solution suggests they might prove effective. Again, the review article by Nishida _et al_. (1981) should be consulted for additional information.

Analysis Tools for Process Design

Analysis tools are defined as those which permit one more easily to do the calculations associated with a process of fixed structure, generally to discover its performance. These tools are in fact needed to establish the equipment sizes and/or operating levels required to reach a specified performance. These aids cover a range of model complexity, ranging from the use of simple linear models in early design calculations to the very highly nonlinear and large models used for final calculations.

The early steps in process design are supported by programs like SYMBOL (Hutchison (1974)), where each element in the process is characterized by a very simple linear model. With this level of detail, one can discover the impact of design decisions such as the degree of conversion for a reaction in a reactor or the sharpness of a separation in a separation step. These are decisions whose values may be uncertain or whose values may be realized through later equipment design decisions. One will discover the variables having major impact and those of apparently minor consequence, using only simple approximate calculations. Also using simple models aids the designer to determine correctly the number of degrees of freedom for his problem, a nontrivial exercise when modeling a system with complex topology.

Later design steps require more rigorous models. The analysis step is almost always done nov using so-called "flowsheeting systems" (see Westerberg et aj^. (1979)). These are large industrial computer programs which allow an engineer to set up and solve an arbitrarily configured process model quickly. These programs contain libraries of equipment models, of supporting physical property evaluation routines, and of cost estimation routines. Such a model comprises a few hundred to 10's of thousands of simultaneous nonlinear algebraic equations. It is what

chemical engineers term a "steady-state model" which corresponds to a nontime varying "DC model" in electrical engineering.

The major contributors of equations to these models are the routines which evaluate physical properties such as equilibrium phase distribution coefficients, mixture heat capacities, entropies and so forth. These may represent 80% of the needed equations, and they are usually unpleasantly nonlinear and poorly behaved.

Unlike electrical engineering, the approach to find a steady-state solution is not usually done by moving from a known "zero state" dynamically to the steady state, although this approach has been advocated several times. Rather the equations are solved directly as simultaneous algebraic equations starting from an initial guess based on user input.

Almost all industrial flowsheeting systems have been based on a single program architecture termed "sequential modular." One has to understand a major difference between chemical engineering building blocks and the building blocks of such electrical simulators as SPICE or ECAP. A building block in a chemical process corresponds to a physical unit, and it may be modeled using hundreds to a few thousand equations. These equations will almost always require simultaneous solution procedures. These "dense" units are sparsely coupled. An electrical device on the other hand can often be modeled by very few equations but it is likely in a circuit which contains hundreds of them. The "macromodel" is more analogous to a typical chemical unit.

A model in a sequential modular simulator is thus in the form of an expertly written subroutine which is tailored to solving the equations for the unit. Assumed, so a solution procedure can be devised, are that all physical inputs to the unit (input streams) are known when the subroutine is called, that all equipment sizes are fixed, that all temperature and

pressure levels are fixed, and that all output stream values are to be calculated by the subroutine. This model is rather like assuming a resistor model will calculate the "outlet" current and voltage given the "inlet" current and voltage. Note that inlet and outlet are meaningful for process units and are of dubious interpretation for a circuit component.

Solving a complex structure of units is done by calling the unit routines in an appropriate sequence. When the physical flows recycle in the process (i.e., form a loop), the computation requires a first guess for the recycle stream(s) at the point where they are inputs to a unit. A sequence of unit calls follows until the recycling stream is calculated as an output of its source unit. The calculated output value is compared to the guessed input value and iterated until adequate agreement is found. Topologies considerably more complicated than a single loop usually exist so the ideas must be (and have been) generalized in the obvious manner (see Chapter 6 in Westerberg et al. (1979) for references and a detailed description of algorithms to find which streams to guess and iterate).

The program architecture for these flowsheeting systems has significant advantages and disadvantages over others possible. It is architecturally very simple and thus easy to implement. It is so thoroughly structured that a design engineer can establish which variables he must specify to permit a calculation. Adding new models is a well defined exercise. Unfortunately it gives the engineer the ability to simulate when his task is to design. He would usually like to specify values for streams leaving the process or interior to the process. He often wants to calculate equipment sizes to get these specifications. He is forced to guess sizes, calculate outputs and reguess sizes as a result.

Two "fixes" exist. One is to write some unit models so an output stream value can be given and the unit will be sized to get it. The variety of combinations of output stream variables one may wish to specify and size variables one may wish to adjust suggests one may need to write a large number of alternate routines. The second "fix" is to provide a "computation control unit" which can be used to do the needed reguessing and iterations automatically. Both are common in industrial flowsheeting systems. Detailed descriptions of these ideas are in Westerberg *et al.* (1979).

Two other architectures exist but are not commonly used. One is a so-called simultaneous modular architecture and the other the equation solving architecture. The latter is of course the approach used most widely in electrical engineering circuit calculations. The simultaneous modular structure models each unit using the same input to output subroutines of the sequential modular approach. However, after executing each of these subroutines, a crude and usually linear model which characterizes the unit is developed. Its coefficients are adjusted to give the same local behavior as the detailed model. The simple model usually captures only the major variable interactions, ignoring the "off diagonal" ones entirely. Being linear however makes it simple to connect the units together in a complex structure and to solve the linear model quickly and directly using sparse matrix methods. Thus one has an exact solution for the complex structure but for approximate unit models. The streams from this solution become the input streams to the units for the next step which is to solve the exact models again. The coefficients for the linear models frequently become the iteration variables whose values are reguessed and iterated. Experience shows this approach often has better convergence behavior than the sequential modular approach.

The third architecture termed "equation solving" has had two approaches developed for it. The first is termed "tearing." Here one rearranges the equations such that, by guessing only a few *(say 50), values for the rest (say 950) can be determined one at a time using a forward substitution through an equal number (950) of (nonlinear) equations. The (50) tear variables are then equal in number to the (50) unused (nonlinear) equations which can be used as error functions for adjusting their values. Effectively the nontear variables are computationally eliminated from the problem, reducing the problem to one in only the number (50) of tear variables.

The approach mimics that used to develop ad hoc solution procedures for a unit model; it requires minimal space on the computer. It is very appealing. Attempts have been made to modify it to handle nested loops (Kevorkian (1980)) or more specifically linear equation subsets (Hernandez* and Sargent (1979); Stadherr *et al*. (1974)). They are bound to be troubled by the inside loop being singular even though the overall problem is not (Edie and Westerberg (1972)). It is this handicap for this approach which weakens but does not destroy it as a general approach for large problems corresponding to major units sparsely interconnected. Tearing algorithms tend to use structure only to find the tear variables, and one would have to be tempted to find tears at two levels for such a problem: the inner for the equations within a unit model and the outer for the variables connecting the units together.

To find tears may be done at several levels if the convergence algorithm employed does not attempt to imbed convergence loops one inside another but rather requires all tear variables to be converged at a single outer level. Westerberg *et aK* (1979) discusses this idea at length. Singularity of the problem as a whole then becomes the only question with this approach.

The second equation solving approach, and the one quickly becoming the dominant one, is the one always used; by electrical engineering – the Newton-Raphson, sparse matrix approach and examples of its use in chemical engineering include those in Goldstein and Stanfield (1970); Hutchison and Shewchuk (1974); Lin and Mah (1978); Locke et al. (1980). The ability to detect numerical singularity while developing the solution procedure is one very significant advantage. However, the computer memory required to handle the sparse matrices is orders of magnitude larger than required for the tearing approach.

The major problem faced by the equation solving system is the handling of the equations for physical properties. Companies have invested enormous amounts of time and money into developing the needed data, the subroutines to fit the data and the subroutines to use the fitted data to calculate physical properties given the temperature, pressure and composition of a stream. The operative word is "calculate."[11] The equations themselves are not accessible from these routines. Also as said earlier, these equations are perhaps 80% of the ones defining the problem. Thus commercially successful programs would seem to have to use these existing routines and try to incorporate them within a Newton-Raphson framework.

The obvious approach is to use numerical perturbation of the routines to build up numerical estimates for the Jacobian elements for the relationships implied by the routines. This approach has to be expensive in computer time. Another *Is* to develop approximate (nonlinear in general) models whose coefficients are set by matching their behavior to that of the exact physical property routines in the vicinity of the current guess for the solution. The approximate equations can then be part of the Newton-Raphson scheme. The last approach is to invent some ad hoc solution schemes which are in part similar to Newton-Raphson and in part similar to

sequential modular (see a description of MULTICOL (Hutchison and Shewchuk (1974)). These schemes work well generally, taking advantage of the known behavior of physical properties.

Our system (ASCEND-II (Locke et aJN (1980)) is under development and is including the physical property equations explicitly, in the face of all the arguments suggesting it may be a poor idea. Until the problem is faced, ways to solve it will not be forthcoming.

Little has been published on the development and use of dynamic simulators for general chemical engineering processes. This does not imply that no activity has occurred, but rather that little has occurred relative to the effort put into developing steady-state flowsheeting systems. In the literature several academically developed dynamic simulation systems are described (Motard (1970); Ham (1971); Franks (1972); Alfonso (1974); Barney et aK (1975); Patterson and Rozsa (1980)). Most of these are not designed to handle really large problems. Also, the handling of simultaneous nonlinear algebraic equations together with the differential equations is not conveniently done by most of these systems. One approach (see Franks (1972)) is like the sequential modular approach, where each unit is modeled by a subroutine which receives as inputs the input stream values to the unit and the "state variable values" for the unit. From these the subroutine develops the unit outputs and the right hand sides for the state equations. Units usually have "capacitance[11] which in principle computationally breaks the recycle loops appearing in a process — but not always. Therein lies a real problem: a recycle computation may exist among some of the units in a process suggesting the need for an iterative computation involving several units embedded inside the dynamic simulation.

Two systems (ASCEND-II (see Kuru (1981)) and SPEED-UP) take an approach based on Newton-Raphson not unlike the "tableau" approach of

electrical engineering, setting up and solving both the algebraic and discretized state equations simultaneously at each time step. ASCEND-II permits the structure of the problem (large units sparsely interconnected) to be used to decompose the the calculations (see Westerberg and Berna (1978)), so larger problems may be handled.

An industrially developed system (DSP-II, JUSE, Japan) uses an equation solving approach based on tearing to solve the algebraic and discretized state equations for a flowsheet. It preprocesses the model equations, developing a solution procedure for the equations. Because it preprocesses the equations all possible alternative discrete decisions must be anticipated, e.g., when a valve may be opened or closed and when closed that it causes a portion of the process to stop functioning.

The least well developed capability for analyzing arbitrarily configured process systems is for solving models which involve partial differential, ordinary differential and algebraic equations. Setting up and solving the models for single units (e.g., chemical reactors) is of course widely dealt with in the literature. Not really handled is a flowsheeting capability where some of the units may be reactors modeled by PDE's. Researchers would be well advised to follow the most recent work by Saito and Scriven (1980a, 1980b), where Newton-Raphson based approaches to handling free surfaces are presented. These ideas, extended, will underlie future developments in flowsheeting systems.

Optimization Aids for Design

As mentioned several times earlier, mixed integer linear programming (MILP) is frequently used as a design aid, particularly to help select among the number of alternative structures one might select for a design* A variety of other optimization strategies are published to solve problems with special structure, e.g., using dynamic programming, branch and bound, etc.

In this section we will concentrate on the use of optimization for finding the best temperatures, pressures, flows, equipment sizes, etc., for a design having a fixed topology. The problem is a nonlinear algebraic equality and inequality constrained, continuous variable optimization problem. It is typically one involving hundreds to thousands of variables, and of equality and inequality constraints. Only a few (1 to 50) degrees of freedom will typically exist for the problem. Westerberg (1980) has published a review of this area.

For problems of this type, the usual approach is to marry optimization to a steady-state flowsheeting system, and such work has been frequently reported (see Gaddy and Jinkerson (1978) for example). Flowsheeting systems are unfortunately ill suited for this marriage. They often make unreported (and sometimes ill-conceived) discrete decisions internally, changing the nature of the solution as the process moves from one type of behavior to another. An example is when the flowrate of a fluid increases sufficiently for the flow to change from laminar flow to turbulent flow* The flowsheeting system will automatically make such model changes, causing the outside optimization algorithm to experience discontinuities in the functions and their derivatives. The optimization scheme has had to be one which survives such problems. Pattern search methods such as the "complex[11] method (Umeda and Ichikawa (1971)) are frequently

advocated. Experience shows them to require a few hundred to a few thousand iterations to reach an answer, a very costly approach.

Handling inequality constraints and added equality constraints to those which are part of the flowsheeting model is also difficult. Usually they cannot be handled by using them directly but only by adding a penalty onto the objective function. Finally, gradients are seldom available, forcing one to use perturbation calculations if they are needed — a computationally expensive proposition.

The equation solving approaches to flowsheeting, particularly those based on Newton-Raphson methods, would appear to be much better suited for optimization. If correctly developed* these systems allow gradients to be had essentially for free using the adjoint network. Thus one can use second order methods for optimizing.

While everything sounds straightforward, a very real problem is not being addressed. Process systems, even in the steady-state, need to contain some general functions of the form

$$f(x) = \begin{cases} \hat{f}(x) & \text{if } g(x) * 0 \\ \hat{\hat{f}}(x) & \text{otherwise} \end{cases}$$

These general functions correspond to analyzing processes where streams might be vapor or liquid with the state not known a priori, flows might be laminar or turbulent, etc. The optimization algorithm, to be effective, must deal with this more general problem. The problem contains, unfortunately, discrete variables.

We can assume that one strategy is to guess values for the discrete variables, optimize a continuous variable problem, see if the optimum satisfies the discrete decisions, where it does not, change the discrete decision and solve again. This general description can be part of both a

rigorous branch and bound strategy or an heuristic one; it does suggest that we must still solve a continuous variable problem effectively as the inner loop which we will now discuss.

The experience in chemical engineering with continuous variable optimization methods suggests the following.

1. Successive linear approximation methods are often but not always very effective. In this approach the objective function and constraints are linearized around the current solution. Constraints are added to control the step size, and the approximate model is solved as a linear program. Often the optimum resides entirely on the constraint boundary for the problem, and the only trick is to find the active constraints. Once found, convergence is second order as the problem is like solving equality constraints only via the Newton-Raphson method.

2. The generalized reduced gradient (GRG) method (Abadie and Carpenter (1969)) is very effective.

3. The generalized penalty function approach is not usually as successful as the GRG method or the successive quadratic approximation approach.

4. The best method appears to be the successive quadratic approximation approach of Han (1975), as implemented by Powell (1977). To handle typical chemical process problems, some method is needed to decompose or reduce the problem size. Berna, Locke and Westerberg (1980) give a decomposition strategy. Using it, the linearized equality constraints are used to reduce the resulting approximate quadratic programming problem which is solved at each iteration. Biegler and Hughes (1980) and Jiraponghan ex jiK (1980) give a two tier approach much like the simultaneous modular approach for analyzing problems, where parameters

for simple-to-solve models are fitted based on the behavior of the exact models, and then the simple (but not necessarily linear) models are used during the optimization.

Alternate Design Criteria

Design criteria, other than economics, are the subject of several studies. Grossmann and Sargent (1979) and Grossmann and Halemane (1980) have done work on designing processes which have to be flexible in their operation. They have developed ideas on how to model a flexible process and then, because of its very special structure, how to solve the optimization problem which minimizes the process cost subject to it being flexible. A special feature of chemical processes is the existence of manipulable variables. These variables can have their values altered as the process faces a different set of inputs, giving the process design a chance to meet the desired output specification. It is assumed a control system, which is yet to be designed, will do the manipulating.

Related to flexibility is uncertainty in design. Here one attempts to account for the uncertainty within which one knows values for a number of the parameters for the design. For example, one may be uncertain of the value of a heat transfer coefficient, but one may want a design which is flexible enough to operate as long as the uncertain parameters are within prescribed ranges. Grossmann and Sargent (1979) and Grossmann and Halemane (1980) also consider this problem.

Morari et al. (1980) approach the problem of designing networks of heat exchangers which can survive a range of input flowrates and temperatures while still requiring the theoretical minimum use of utilities and while still meeting the target temperatures of the outlet streams. They call such a solution a "resilient" solution. They also indicate how one could develop and operate a control structure for such a system.

Safety is another design criterion. The traditional approaches have all been explored, e.g., the use of fault trees. Relevant publications include Powers and Tompkins (1974); Lapp and Powers (1976, 1977) and Kumamoto and Henley (1979).

Lapp and Powers (1976, 1977) have added a twist to the developing of a fault tree for a process. They describe a quite different process model called a "cause and effect model" from which one can automatically generate a fault tree. According to their approach, the engineer can develop a model of the process, which includes all the equipment items. Stored in a library are "cause and effect" models for each equipment type, which are then "wired" together to model the system the engineer has described. From these system models, a fault tree is automatically developed. In principle, considerable time can be saved using these ideas because the current approach to set up a fault tree requires enormous amounts of tedious work.

Kumamoto and Henley (1979) develop a method to find failure modes which can account for the sequence in which the failures occur; thus they handle control loops more naturally. The method does not develop a fault tree but rather depends on the construction and use of decision tables.

Process Design to Hardware Conversion

This step in the design is where the functional description (which can usually be presented on 2 engineering drawings) is converted into a set of 40 to 60 Piping and Instrumentation Diagrams (PID[f]s) which detail every actual piece of equipment to be purchased. These drawings are only two dimensional so they do not allow one to determine the exact piping needed, nor where connections to vessels are geometrically located, etc. A number of aids exist for this step* Drawing oriented dedicated computer systems, such as the Applicon System, are used to aid this step. The engineer sits at a CRT terminal and constructs, out of an extensive catalogue of parts, the PID[f]s directly. The system draws columns, valves, heat exchangers, interconnecting piping, etc., placing them where requested. By attaching information to these items (e.g., through a keyboard input), the drawings become one-to-one with a list of all equipment items called a "materials takeoff list."[11]

Until the actual equipment is known, many control and safety questions cannot be answered. One synthesis activity is to establish automatically the control structure for the process on a PID. Papers by Govind and Powers (1976, 1977, 1978), Morari _et al_. (1980a,b,c) and Arkun and Stephanopoulos (1980) are aimed at this problem. The control problem is first to select which variables to control. The second decision is to select which variables one should measure and the third which one should manipulate to effect the desired control. The fourth is to select the control structure connecting the measured and manipulated variables.

Govind (1978), in his Ph.D. thesis, develops control structures involving the more traditional controller instrumentation (PID controllers) arranged in standard and novel ways. Morari _et al_. (1980a,b) lay down a very interesting and useful framework within which to describe and solve

this problem. They work toward finding useful problem decompositions to reduce the number of schemes one would otherwise generate* Decomposition includes breaking processes apart where the economic coupling is weak, selecting variables which can simply be regulated, finding dynamic loops which can be successfully solved by "tearing" them, etc. Also they develop a number of tools to allow one to select automatically all possible candidates for measuring and manipulating, considering, for example, extensions to the ideas of controllability and observability.

At this point in the design, more detailed safety and reliability studies, discussed earlier, are performed.

One new major industrial system is now under development to attempt to create a rather all encompassing aid for this step at the Computer Aided Design Centre in Cambridge, England. The project.is known as PEGS. In its initial version, the drawing aids will likely be those emphasized.

Design of Major Equipment

/ In parallel with the effort to develop the detailed PID's for design /

is the activity associated with designing the major equipment items which are not off-the-shelf items* Examples are heat exchangers, storage vessels, columns, etc. This design activity is really in the realm of mechanical engineering and will not be discussed here at any length. The problem is to design this equipment to meet safety codes (which differ from country to country).

In chemical engineering, two organizations (at least) exist which develop design aids for heat exchangers. These are HTRI (USA) and HTFS (UK). They receive subscription fees from the companies they serve. Their **goal** is to develop computer programs to design all types of heat exchangers. They carry out extensive experimental programs to develop the correlations to permit them to estimate the effects of various heat transfer and fluid flow phenomena. Phase changes often complicate the behavior found in such devices. When designing a heat exchanger numerous discrete decisions must be made, a synthesis task. Examples are to select which fluid is to flow through the tubes and which over them, what the flow pattern of the two fluids relative to each other should be (ranging from countercurrent to cocurrent and everything in between), what the tube sheets should look like, etc. Nothing has been published on how these decisions are made automatically, but one must suspect it is at present done heuristically.

Analysis programs include the use of finite element models, etc. Again, this activity is more a mechanical engineering one and we leave it at this point.

3 Dimensional Visualization

A remaining major design activity before construction commences is to develop a three dimensional model for the process. This step is invariably done at present by actually building a model out of plastic. The alternative which is available is to develop a 3-dimensional model in a computer data base and to use sophisticated graphics techniques to display this model on a CRT. At least one such package of real consequence has been developed. It has been developed by the Computer Aided Design Centre in England and is called PDMS (Piping Data Management System). Drawings of all sorts are one of the outputs possible from this program. Closely related is the graphics technology behind design packages for buildings which also communicate with the architect and engineer through 3-dimensional visualizations.

A number of interesting research activities suggest themselves to support such a package. One design activity is to place the equipment on the plot of land in such a way that piping costs are minimized, that fires, if they were to occur, can be contained (e.g., keep the storage vessels away from other vessels and a long way from the reactors), that equipment can be conveniently maintained, etc.

A subsidary problem is to route pipes between equipment items already positioned -- sounds like laying out an integrated circuit? The problem is three dimensional because equipment is also placed in the vertical direction.

Plant Operating Procedures

The last major design activity we shall touch upon is the design of the operating procedures for a plant. The plants have to be started up. The startup activity can blossom into a many month activity if extreme care is not taken in the "design" of this step. Also the plants are frequently moved from one operating mode to another while still producing. They are also shut down for scheduled maintenance or because of an emergency.

An heuristic approach to solving the "design" of operating procedures must perforce exist in any computer driven batch process. A batch process is one which is not run continuously, but is constantly being moved from one operating mode to another. Industry has developed programs for this activity, one example being AUTRAN developed by Merck and Control Data Corporation.

A Ph.D. thesis has appeared in this area by Teague (1980) working for G. Powers. This thesis described a language for writing operating procedures. It is an attempt to create a language rich enough to permit the description of any activity a human operator might do to operate a plant, including the making of mistakes. A purpose of this language was to provide a model of an operator so a fault-tree analysis could be generated automatically for a process, including the operator actions. One can also envision a true synthesis activity in which the operating procedures are generated automatically; a study of this type has already appeared (Rivas et al. (1974a,b)).

On top of this last activity are the higher level decisions of scheduling the plant production to meet predicted, but varying market demands, new material availability, ambient conditions, etc.

Clearly there are no shortages of problems to solve in process design.

In Conclusion

We have reviewed the design process for chemical plant design, tracing it from its inception through, to the design of its operating procedures. Along the way we have described a number of computer aids which can exist to help in the activities being discussed. We certainly have omitted many for lack of space and knowledge of them. The penultimate system presumably will be to tie all these design aids together with several as yet to be developed aids, using a sophisticated data base as the repository for all partial and completed results. The ultimate will be for the whole process to occur automatically, or will it? One significant goal of a design aid is to train the engineer about the design so he can understand it and, very importantly, criticize it. Hopefully, until we really can do design automatically, anticipating all possible problems, we will not forget this requirement.

References

Abadie, J. and J. Carpenter, "Generalization of the Wolfe Reduced Gradient Method to the Case of Nonlinear Constraints," Chapt. 4 in Optimization, Ed. R. Fletcher, Academic Press, New York, NY, 37 (1969).

Agnihotri, R.B. and R.L. Motard, "Reaction Path Synthesis in Industrial Chemistry," ACS Symp. Ser., Computer Applications to Chemical Process Design and Simulation, 124, 193 (1980).

Alfonso, L.L., "DYSCO: An Interactive Executive Program for Dynamic Simulation and Control of Chemical Processes," Ph.D. Thesis, Univ. of Michigan (1974).

Arkun, Y. and G. Stephanopoulos, "Studies in the Synthesis of Control Structures for Chemical Processes. Part IV: Design of Steady-State Optimizing Control Structures for Chemical Process Units," AIChE J., 26, 975 (1980).

Barney, J.R., R.S. Ahluwala and A.I. Johnson, DYNSYS Users Manual, Univ. of Western Ontario, London, Ontario (1975).

Biegler, L.T. and R.R. Hughes, "Application of Powell's Algorithm to Quadratic Approximation Programming," 73rd Annual Meeting, AIChE, Chicago, IL, Nov. 16-20 (1980).

Corey, E.J. and W.L. Jorgensen, "Computer-assisted Synthetic Analysis. Synthetic Strategies Based on Appendages and the Use of Reconnective Transforms," J. Am, Chem. Soc, 98, 189 (1976).

Edie, F.C. and A.W. Westerberg, "A Potpourri of Convergence and Tearing," Chemical Engng. Computing, 1, AIChE, Nev; York, 35 (1972).

Franks, R.G.E., Modeling and Simulation in Chemical Engineering, John Wiley and Sons (1972)-

Gaddy, J.L, and K.R. Jinkerson, "Ethylene Process Optimization — Constraining Relaxation and Bounds Adjustment," 71st Annual Meeting AIChE, Miami, FL, Nov. (1978).

Gasteiger, J., P.D. Gillespie, D. Marquarding and I. Ugi, "From van't Hoff to Unified Perspectives in Molecular Structure and Computer Oriented Representation," Topics in Curr. Chem., 41, 1 (1974).

Gelernter, H., N.S. Sridharan, A.J. Hart and S.C. Yen, "The Discovery of Organic Synthesis Routes by Computer," Topics in Curr. Chem., 41, 113 (1973).

Goldstein, R.P. and R.B. Stanfield, "Flexible Method for the Solution of Distillation Design Problems Using the Newton-Raphson Technique," IEC Process Design and Develop., ^9, 78 (1970).

Govind, R., "Control System Synthesis Strategies," Ph.D. Thesis, Carnegie-Mellon Univ., Pittsburgh, PA (1978).

Govind, R. and G.J. Powers, "Control System Synthesis Strategies," paper presented at 82nd National Meeting, AIChE, Atlantic City, NJ (1976).

Govind, R. and G.J. Powers, "Interaction Analysis in the Synthesis- of Control Structures," paper presented 70th Annual Meeting, AlChE, New York, NY, Nov. (1977a).

Govind, R. and G.J. Powers, "A Chemical Engineering View of Reaction Path Synthesis," in Computer Assisted Organic Synthesis, Ed. W.T. Wipke e£ al., 61^, 81, ACS Symp. Ser. (1977b).

Govind, R. and G.J. Powers, "Synthesis of Process Control Systems," IEEE Trans, on Sys. Man and Cyber, FMC-8, 792 (1978).

Grossraann, I.E. and K.P. Halemane, "A Decomposition Strategy for Designing Flexible Chemical Plants," 73rd Annual Meetling, AIChE, Chicago, IL, Nov. 16-20 (1980).

Grossmann, I.E. and J. Santibanez, "Applications of Mixed Integer Linear Programming in Process Synthesis," 72nd Annual Meeting, AIChE, Nov. (1979).

Grossmann, I.E. and R.W.H. Sargent, "Optimum Design of Multipurpose Chemical Plants," I&EC Process Design and Develop., 18, 343 (1979).

Ham, P.G., "REMUS The Transient Analysis of Integrated Chemical Processes," Ph.D. Thesis, Univ. of Pennsylvania (1971).

Han, S.P., "A Globally Convergent Method for Nonlinear Programming," Dept. of Comp. Sci., Cornell Univ., Report No. 75-257 (1975).

Hendrickson, J.B., "A General Protocol for Systematic Synthesis Design," Topics in Curr. Chem., 62, 49 (1976).

Herandez, R. and R.W.H. Sargent, "A New Algorithm for Process Flowsheeting," European Federation of Chemical Engng. 12th Symp. on Computer Applications in Chem. Engng., Montreux, Switzerland, April 8-11 (1979).

Hutchison, H.P., "Plant Simulation by Linear Methods," Trans. Inst. Chem. Eng., 52, 287 (1974).

Hutchison, H.P. and C.F. Shewchuk, "Computational Method for Multiple Distillation Towers," Trans. Inst. Chem. Eng., 52, 325 (1974).

Jiraponghan, S., J.F. Boston, H.I. Britt and L.B. Evans, "A Nonlinear Simultaneous Modular Algorithm for Process Flowsheet Optimization," 73rd Annual Meeting, AIChE, Chicago, IL, Nov. 16-20 (1980).

Kevorkian, A.K., "Decomposition Methodology for the Solution of Large Non-linear Problems," Foundations of Computer-Aided Chemical Process Design Conference, Engng. Foundation Conferences, Henniker, NH, July 6-11 (1980).

Kumamoto, H. and E.J. Henley, "Safety and Reliability Synthesis of Systems with Control Loops," AIChE J., £5, 108 (1979).

Kuru, S., "Dynamic Simulation with an Equation-based Flowsheeting System," Ph.D. Thesis, Carnegie-Mellon Univ., Pittsburgh, PA (1981).

Lapp, S.A. and G.J. Powers, "Computer-Aided Fault Tree Synthesis of Fault Trees," Chem. Engng. Prog., 72, 89 (1976).

Lapp, S.A. and G.J. Powers, "Computer-aided Synthesis of Fault Trees," IEEE Trans, on Reliability, R-26, 2, April (1977).

Lin, T.D. and R.S.H. Mah, "A Sparse Computation System for Process Design and Simulation: Parts I and II," AIChE J., 24, 830, 839 (1978).

Locke, M.H., S. Kuru, P.A. Clark and A.W. Westerberg, "ASCEND-II: An Advanced System for Chemical Engineering Design," 11th Annual Pittsburgh Conference on Modeling and Simulation, Univ. of Pittsburgh, May 1-2 (1980).

Mahalec, V. and R.L. Motard, "Procedures for the Initial Design of Chemical Process Systems," Computers and Chem. Engng. J., 1, 57 (1977a).

Mahalec, V. and R.L. Motard, "Evolutionary Search for an Optimal Limiting Process Flowsheet," Computers and Chem. Engng. J., 1, 149 (1977b).

Morari, M., Y. Arkun and G. Stephanopoulos, "Studies in the Synthesis of Control Structures for Chemical Processes. Part I: Formulation of the Problem. Process Decomposition and the Classification of the Control Tasks. Analysis of the Optimizing Control Structures," AIChE J., 26, 220 (1980a).

Morari, M., D.F. Marselle and D.F. Rudd, "Synthesis of Resilient Energy Management Systems," paper 3f, 73rd Annual Meeting, AIChE, Chicago, IL, Nov. 16-20 (1980).

Morari, M. and G. Stephanopoulos, "Studies in the Synthesis of Control Structures for Chemical Processes. Part II: Structural Aspects and the Synthesis of Alternate Feasible Control Schemes," AIChE J., 26, 232 (1980b).

Morari, M. and G. Stephanopoulos, "Studies in the Synthesis of Control Structures for Chemical Processes. Part III: Optimal Selection of Secondary Measurements within the Framework of State Estimation in the Presence of Persistent Unknown Disturbances," AIChE J., 26, 247 (1980c).

Motard, R.L., PRODYC - A Simulation Program for Process Dynamics and Control Users Guide, Univ. of Houston, TX (1970).

Nishida, N., G. Stephanopoulos and A.W. Westerberg, "A Review of Process Synthesis," to appear AIChE J. (1981).

Patterson, G.K. and R.B. Rozsa, "DYNSYL: A General-Purpose Dynamic Simulator for Chemical Processes," Comp. and Chem. Engr., Vol. 4, 1-20 (1980).

Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations," presented at 1977 Dundee Conference on Numerical Analysis (1977).

Powers, G.J. and F.C. Tompkins, "Fault Tree Synthesis for Chemical Processes," AIChE J., 20, 376 (1974).

Rivas, J.R., D.F. Rudd and L. Kelly, "Computer-Aided Safety Interlock Systems," AIChE J., 20, 311 (1974a).

Rivas, J.R., D.F. Rudd, "Synthesis of Failure Safe Operations," AIChE J., 20, 320 (1974b).

Rudd, D.F., "Modeling the Development of the Intermediate 'Chemicals Industry," Chem. Engng. J., 9, 1 (1975)..

Saito, H. and L.E. Scriven, "Study of Coating Flow by the Finite Element Method," 73rd Annual Meeting, AIChE, Chicago, IL, Nov. 16-20 (1980a).

Saito, H. and L.E. Scriven, "Autonomous Boundary Conditions at Free Surfaces," American Physical Sco. Fluid Dynamics Division Meeting, Cornell Univ., Ithaca, NY, Nov. 25 (1980b).

Siirola, J.J., G.J. Powers and D.F. Rudd, "Synthesis of System Designs. Ill: Toward a Process Concept Generator," AIChE J., 17, 677 (1971).

Sophos, A., E. Rotstein and G. Stephanopoulos, "Thermodynamic Bounds and the Selection of Technologies in the Petrochemical Industry," Chem. Engng. Sci., 35, 1049 (1980).

Stadherr, M., Ph.D. Thesis, Univ. of Wisconsin, Madison, WI (1976).

Stadherr, M., W.A. Gifford and L.E. Scriven, "Efficient Solution of Sparse Sets of Design Equations," Chem. Engng. Sci., 29, 1025 (1974).

Stadherr, M. and D.F. Rudd, "Systems Study of the Petrochemical Industry," Chem. Engng. Sci., !U, 1019 (1976).

Stadherr, M. and D.F. Rudd, "Resource Management in the Petrochemical Industry," Management Sci., 24, 740 (1978).

Teague, T.L., "A Study for the Design of Operating Procedures," Ph.D. Thesis, Carnegie-Mellon Univ., Pittsburgh, PA (1980).

Umeda, T. and A. Ichikawa, "Modified Complex Method for Optimization," I&EC Process Design and Develop., K), 229 (1971).

Westerberg, A.W., "Optimization in Computer-Aided Design," Engng. Foundation Conference on Computer-Aided Process Design, Henniker, NH, July 6-11 (1980).

Westerberg, A.W., H.P. Hutchison, R.L. Motard and P. Winter, Process Flowsheeting, Cambridge Univ. Press, Cambridge, England (1979).

Wipke, W.T., H. Braun, G. Smith, F. Choplin, and W. Sieber, "SECS," in Computer Assisted Organic Synthesis, Ed. W.T. Wipke and W.J. Howe, ACS Symp. Ser., 61, 97 (1977).