DESIGN RESEARCH - BOTH THEORY AND STRATEGY

by

Arthur W. Westerberg

DRC-06-24-81

September 1981

DESIGN RESEARCH — BOTH THEORY AND STRATEGY

by

Arthur W. Westerberg
Carnegie-Mellon University

Presentation prepared for ChE Division Award Lecture sponsored by 3M, American Society for Engineering Education, Univ. So. Calif., Los Angeles, June 1981.

Abstract

Design research is often identified as the developing of theory to support design calculations. This presentation, an opportunity to be philosophical, will emphasize another important aspect to design, the strategy. Five guiding principles are stated: 1) Evolve from simple to complex, 2) use a depth first approach, 3) develop and use approximate criteria either as targets or heuristics to screen among alternatives, 4) use "top down[11] design alternatively with "bottom up[19] design, and 5) all things being equal, be optimistic. While some of these guidelines may be obvious, they are frequently unheeded and seldom taught.

Diverse examples will illustrate design strategies consistent with these guidelines. The examples will show how to use the guidelines to design or evaluate computer aids and even to "prove" the value of the guidelines themselves.

Introduction

This presentation is an opportunity to be philosophical about design, an opportunity not to be missed* The ideas to be given here are my version of ideas generated both in my own work and during several lively discussions with James Douglas (University of Massachusetts) and Bodo Linnhoff (Imperial Chemical Industries).

Design research is often narrowly viewed to be research to develop theory supporting computational methods useful for performing design calculations. The methods might be new convergence techniques, better stiff ODE integration methods, new optimization algorithms particularly well suited for systems of interconnected units, and so forth. Even the relatively new area of process synthesis is frequently viewed as solvable by using similar ideas, but perhaps using techniques which allow for a number of the variables to take on only discrete values.

The significant questions relating to synthesis were aptly stated by Simon (1969) and are further amplified in Motard and Westerberg (1978). They are 1) how does one represent the alternative configurations permitted when developing a design, 2) how does one establish a value for each alternative so as to identify which are the better ones, and 3) how does one search among the enormous number of alternatives one is certain to create.

The guidelines we wish to state here speak principally to the third issue and partly to the second. We hope to show that powerful guidelines do exist which can be used to solve most open ended design problems directly or which can be used to design and evaluate aids and strategies which will be useful for solving such problems.

We conjecture that these guidelines can be taught; we (I. Grossmann and the author) attempt to do just that in our undergraduate and graduate design classes* We hope also to convince the reader that this aspect of design research is a valid contribution but one frequently avoided or understated when presenting new results.

The Guidelines

We offer the following five guidelines to use when solving design problems.

1) Evolve from simple to complex
2) Use a depth-first approach
3) Develop approximate criteria either as targets or heuristics for screening among alternatives
4) Use "top down" design techniques alternatively with "bottom up" ones.
5) All things being equal, make optimistic assumptions.

We shall now explain each of these ideas in more detail and then, for the rest of the paper, examine their application to several examples. If the guidelines are true, then one should be able to use them to design a means to demonstrate their own validity; i.e., the ideas should be recursive.

Evolve from Simple to Complex

All earlier calculations for a design should be done using simple calculations even if one knows them to be quantitatively incorrect. The earlier calculations are for learning about the design qualitatively. Many of the major decisions can be made obvious by use of approximate calculations only. Hardly anyone experienced in design violates this guideline for long in practice, but when they do, failure to complete the needed calculations frequently results.

An obvious example is to prepare an outline to a research paper before writing it.

Use a Depth-First Approach

This guideline suggests one should go directly for a first feasible solution to the problem at hand, based on a sequence of best local decisions. One should avoid the tendency to backtrack at any point prior to finding an initial complete solution to the problem. (Outline the whole report.)

The reasoning is as follows. The initial design is an enormously effective learning device; it gives the designer his first glimpse as to the steps which are easy and to those which are the important difficulties to be encountered in the problem, with perhaps some difficulties being insurmountable. In this latter case, the design can be abandoned with minimal work expended.

"Depth first" is a term used to search a tree of decisions. It is a search strategy in opposition to "breadth first" searching. Breadth first allows backtracking prior to completing the first design if earlier decisions no longer appear to be likely winners.

To repeat this guideline — generally avoid backtracking. Go as quickly as possible to the first potential solution.

These first two quidelines permeate the recent publication by Douglas as well as the lecture notes for our own undergraduate design course.

Develop Approximate Criteria

One reason the design question is difficult to deal with is that design is caught in a dilemma. The final criteria used to assess the value of a design (if the criteria can be stated) cannot be evaluated without having in hand a completed design. Thus one must make initial decisions which one can only hope will result in solutions that are a good compromise with respect to the final criteria. To carry out the initial design, alternative approximate criteria must of necessity be used. Often these are in the form of heuristics. At other times they can be locally realizable targets.

A significant research contribution can be the discovery of effective approximate criteria, as we shall see has occurred in the synthesis of heat exchanger networks. The targets themselves may be considered the initial simple calculations needed for the earlier design stages. Linnhoff, in his research publications, is a vociferous advocate of target setting.

Use Top Down/Bottom Up Design Alternatively

Top down and bottom up design are forms used to describe how to design computer programs. The former, top down, refers to starting at the highest level with the overall goal of the design. This goal is then partitioned into subgoals which, if solved, will accomplish the higher goal. These subgoals are then each treated as the top level goals to be further partitioned, etc., until lowest level subgoals are discovered which can be implemented without further partitioning.

Bottom up design is to design first the lowest level building blocks which one assumes will be necessary to accomplish the design. In computer programming, writing a linear equation solving subroutine first would be part of a bottom up strategy for designing a nonlinear equation solving package, where one assumes such a subroutine will be needed.

What is being advocated here is to use the two strategies alternatively. The top down strategy should be used to scope out the alternatives in terms of high level tasks needed to solve the design. Once set, then bottom up design should be used to locate bottom level subtasks which will preclude a solution. Thus they will, for minimal effort, rule out an alternative suggested by top down design. To solve a bottom level subtask requires guessing the environment for the bottom level subtask.

Be Optimistic

Douglas (1979) conjectures that 99% of all initial design concepts will prove to be technologically or economically unsatisfactory — i.e., they will fail as concepts. The correct mindset, and one a designer usually fails to have, is to try to prove concepts will not work.

When attempting to use bottom up design to rule out design concepts, one should use optimistic guesses as to the environment for the bottom level task. If the task cannot succeed when being optimistic, then the failure to do the task can be used to rule out the top down concept requiring it. If one uses conservative guesses, and the bottom level task proves difficult, it may be because of the use of an overly conservative set of guesses as to the task environment, and thus one would be unable to use its behavior to rule out the concept.

A corollary to the above guidelines is that one should use the information learned from the original solution to move to subsequent improved solutions, using in one form or another a learning or evolutionary approach.

A second corollary to the above guidelines is that computer aided process design programs which do not cater to them will be significantly less useful than those which do.

The design problem is one of searching an enormous space of alternatives to select the correct building blocks and their interconnection, as well as also searching the space of continuous variables to establish the levels at which to operate any given structure. The guidelines are consistent with the following specific search strategy.

1) Select a limited technology within which to solve the problem.

2) Using heuristics sketch a good initial solution from within the allowed technology.

3) Examine this solution and develop alternative solutions by revising within the allowed technology or within a modified allowed set of technology, where the initial solution suggests the allowed set modifications. Iterate from Step 2 until a "best" solution is found.

4) Repeat Steps 2 and 3 using more complete models.

The guidelines also support the following specific strategy.

1) Select a limited technology within which to solve the problem.

2) Within this technology set up a superstructure within which is embedded all the alternatives of interest. Use heuristics to eliminate obviously useless portions of the superstructure as it is being developed.

3) Use algorithmic methods to discover the best substructure from among the alternatives embedded in the superstructure.

4) Examine the solution and develop modifications to the allowed technology within which to search.

5) Return to Step 3 until no improvements are possible.

6) Iterate Steps 2 to 5 with more complete models.

(Steps 3 and 4 can be very mathematical, giving rise to the development and use of sophisticated theorems, and thus perhaps satisfying many persons that quality abounds in the results.)

The advantage to this last approach is that parallel decisions are made in Step 3 so in a sense an optimal solution is found, but it is found by looking among a rather small set of alternatives. Fallible heuristics are used only to make the more riskfree problem reductions.

The sequential aspects to the approach are to learn which technological alternatives ought to be in the superstructure and to solve initially using simple models to get closer to the final solution before starting to do complex calculations.

Examples

We shall now describe four example "design" problems to illustrate the effectiveness of the guidelines.

An Entire Chemical Process

The first example is to scope out a process to hydrolyze ethylene (EL) to ethyl alcohol (EA) via the reaction

70 atm

560K

$$CH_3CH_2OH$$

EL W EA

The available ethylene feed contains one mole percent methane (M) and three mole percent propylene (PL). Propylene also hydrolyzes to iso-propyl alcohol (IPA) but to a lesser extent at the given reactor conditions. Croton aldehyde (CA), a C_4 aldehyde, forms as a trace by-product. Diethyl ether (DEE) forms in equilibrium with water and ethyl alcohol:

ICELjC&^OEL «* CHJCHJOCHJCB^ + HjO .

EA DEE W

Conversion of the ethylene is from 5 to *7Z,* with water in significant excess in the reactor feed.

Skipping lightly over many details, we start our design by scoping **out the** process using a top down view, getting at least the three structures illustrated in Figure 1.

Remembering that the strategy being advocated suggests striking out for a completed design without backtracking, we must select one of these sketches (or a variant); we use a bottom up design technique to rule out alternatives* We look for reasons a concept will likely fail and do a quick bottom level calculation to validate our conjecture, guessing the most optimistic environment we can for that calculation.

The first two variants in Figure 1 look as if they might fail because of the extremely low temperatures which may be required if we were to use distillation to effect the initial separation step. We need only a Mollier diagram for ethylene to see that at $P < P_c = 50.7$ atm, the highest temperature possible at the top of an ethylene/propylene column is 0°C« Refrigeration would be required, and, as an approximate criterion, we rule out using refrigeration if possible. The third option, if volatilities are examined, could be implemented to remove the methane and propylene by recycling them back with the ethylene to the reactor. Since methane is an inert here, it would build up and could be removed by bleeding it. The propylene will both convert to iso-propyl alcohol and be lost in part in the bleed. Finally comparing boiling points for water, iso-propyl alcohol, and the azeotope of water and ethyl alcohol suggests this separation is possible. All other separations look rather straightforward. We adopt option 3.

An automatic synthesis program for developing total flowsheets should be able to come quickly to this same result. If not, it must be working too hard. Remember this flowsheet is not purported to be the best

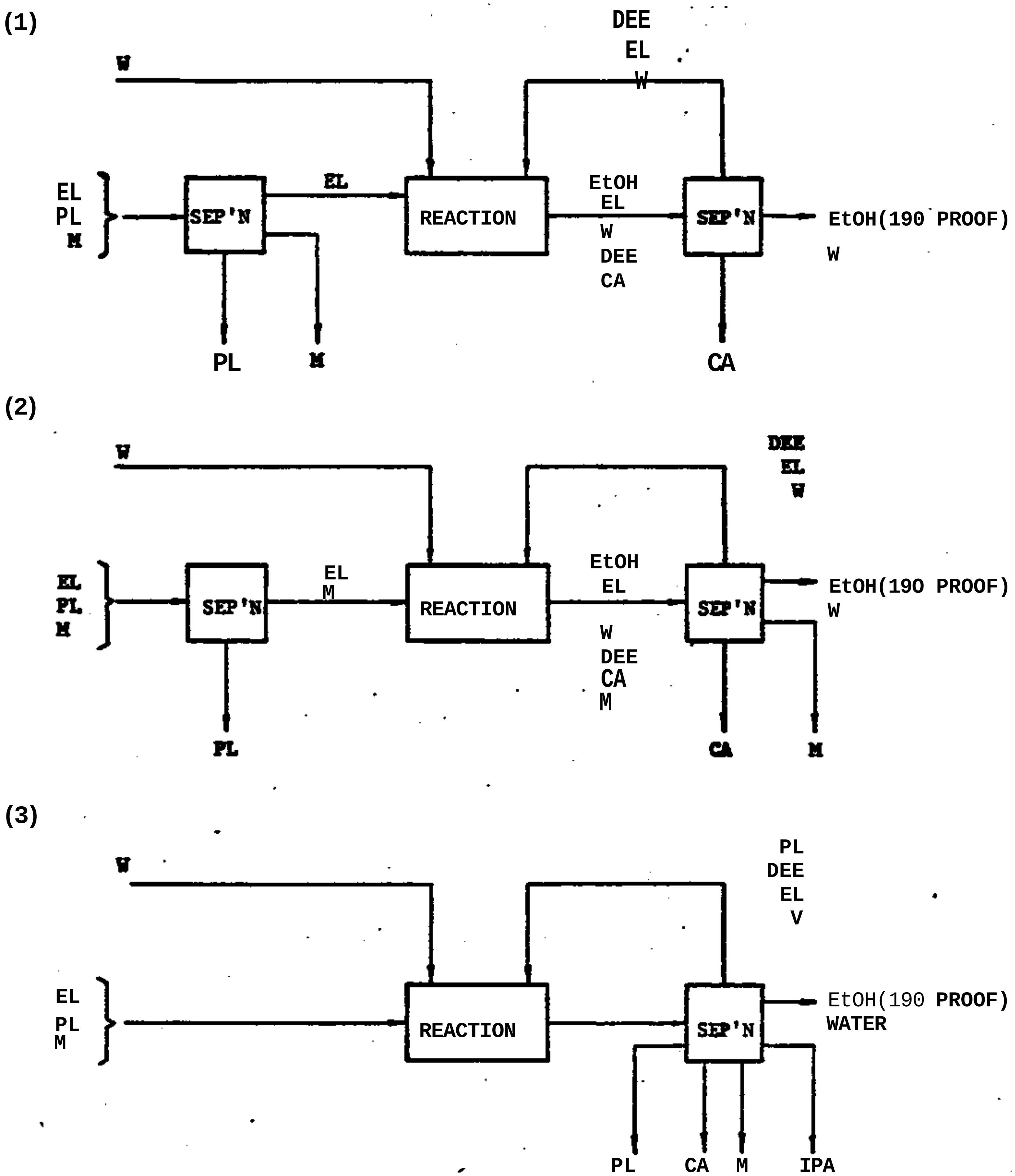

Figure 1. Top Down Sketches for Ethylene to Ethyl Alcohol Process

one, only a good first one from which we intend to learn about the process so our second guess as to the solution is done with much improved insight.

Separation System Synthesis

The second process example we shall look at is separation system synthesis. We have an obvious candidate in our previous example, the separation of methane, propylene, ethylene, diethyl ether, ethyl alcohol, water, iso-propyl alcohol and croton aldehyde into the product ethyl alcohol, a recycle of ethylene, diethyl ether and water, and the by-products of methane, propylene, iso-propyl alcohol and croton aldehyde. The separation step of the third option in Figure 1 illustrates the problem. Note the feed to that step is vapor at high pressure and the recycle is also a vapor which needs to be returned at high pressure.

The strategy we now look at will be the first one stated earlier, one we claim is consistent with the guidelines given:

1) Select a technology within which to solve the problem.
2) Using heuristics, sketch a good candidate solution. Evaluate it.
3) Examine the solution and develop alternate solutions by revising within the allowed technology or by adding new technology.

If we were trying to develop our earlier flowsheet fully, we would likely skip Step 3 above because it represents backtracking. If, on the other hand, the separation problem is our entire design problem, Step 3 is a refinement step, one that follows our having a first complete solution.

Figure 2 sketches a possible solution to the above separation problem using distillation technology. The heuristics used are ranked in order of importance and are a paraphrase and subset of those in Seader and Westerberg (1977). For the next separation

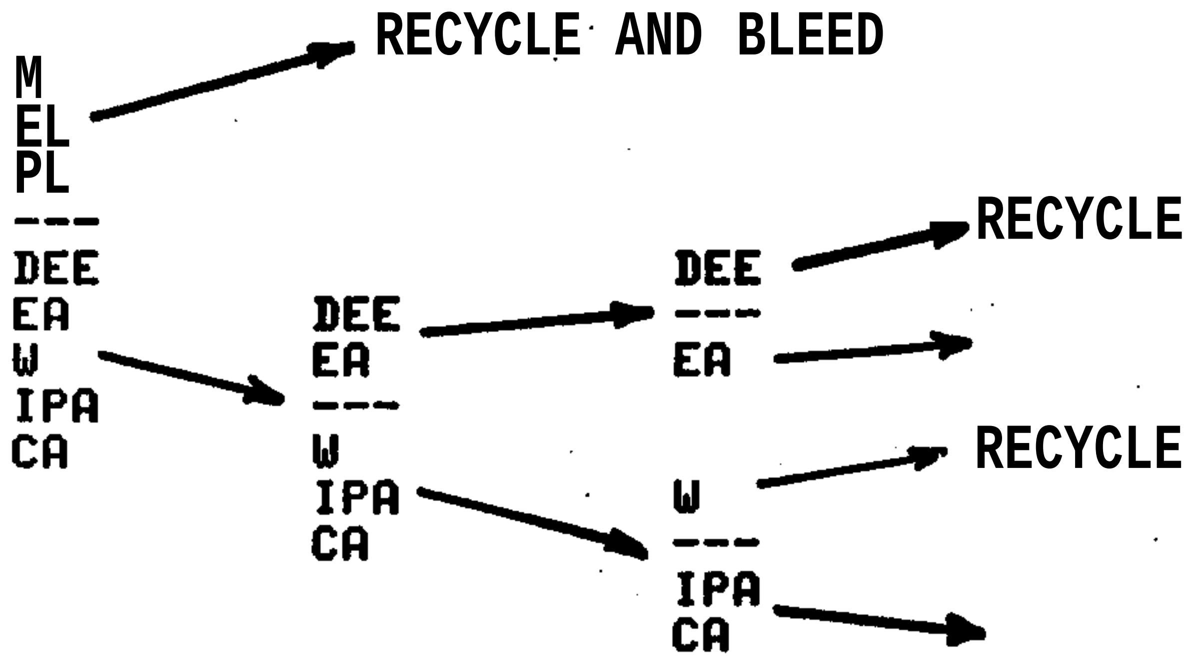

Figure 2. First Sketch for Separation System

1) do the easy split or
2) remove the most bountiful component or
3) remove the most volatile component.

The split between diethyl ether and ethyl alcohol can be done easily; do it first. The recycle can tolerate methane and propylene so let them recycle, but then remove methane using a bleed stream. Go after the water which is plentiful next but, using heuristic 3 also, split above it to remove the ethyl alcohol. Finally split off the water from IPA and CA.

At this point let us consider the separation problem as the whole problem we are solving. For this problem Mark Andrecovich, a Ph.D. student of mine, is discovering that the second strategy stated earlier, where one creates a sequence of superstructures to be optimized, seems to be very effective. Figure 3 illustrates the solution found to a 3 component separation using this approach. It is 11% less expensive than all obvious competitors on an annualized cost basis which considers both investment and operating costs. Note the complexity of this structure. The research question is to establish a means to locate it quickly.

ASCEND-II: An Analysis Aid for Arbitarily Configured Processes

We shall move off on an entirely new tack at this point and describe briefly the ASCEND-II flowsheeting system (Locke, et al (1980)) that we are developing in my research group at Carnegie-Mellon University. The persons involved are Michael Locke (Locke (1981)), Selahattin Kuru (Kuru (1981)), Peter Clark (Clark (1980)), Dean Benjamin and Andrew Hrymak. The messages to be conveyed by this example are two: the breadth of research activities which support this project and a description of the use of this system to develop a working analysis model for a process in a manner which is consistent with the design strategy that has been the main theme of this paper.

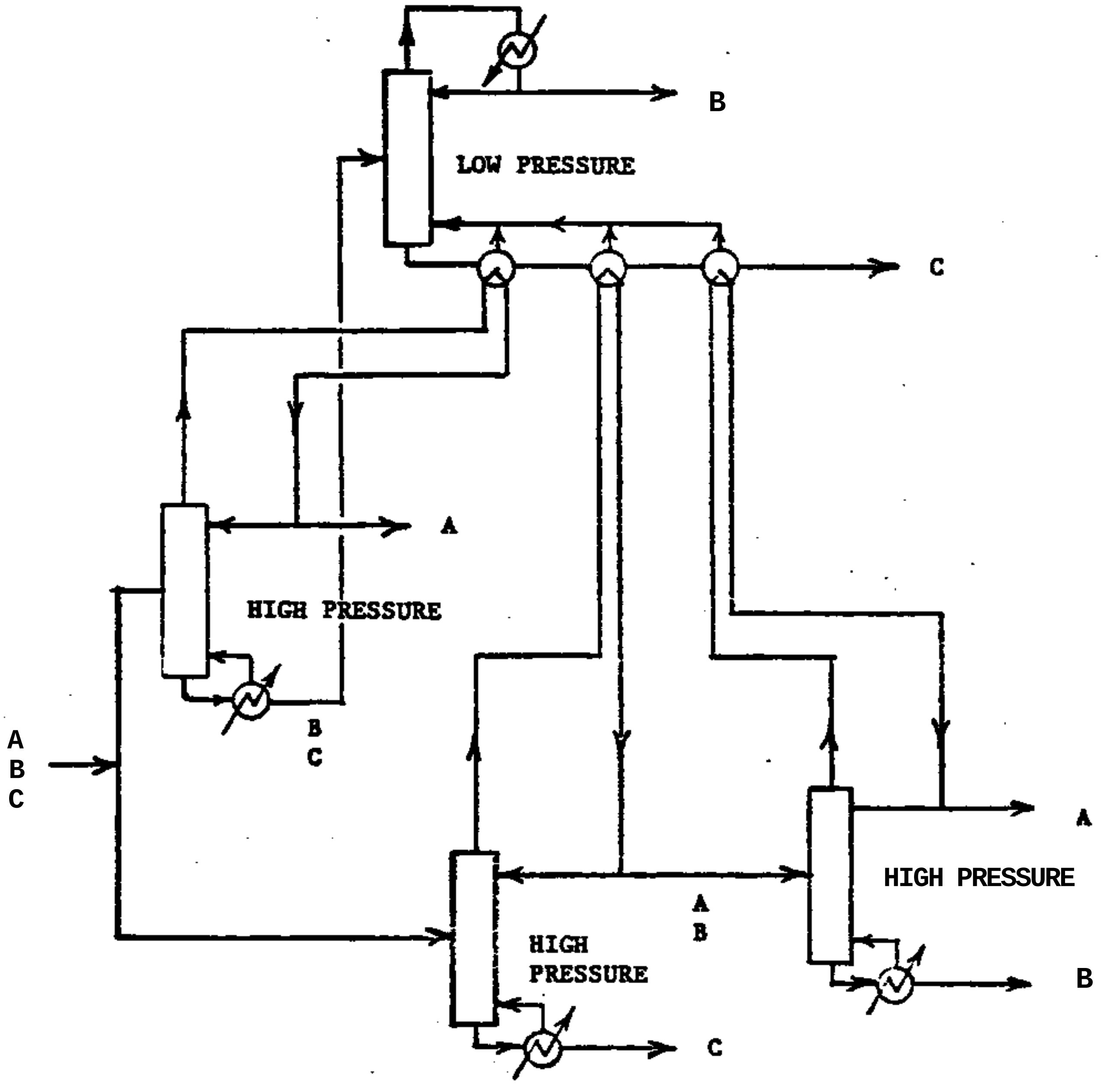

Figure 3. Highly Heat Integrated Distillation Scheme
Using Multiple Effect Columns

To examine ASCEND-II we need first to establish what we mean by analysis. We include the following types of analysis for a given but arbitrarily configured process flowsheet.

1) Simulation. The inputs to the process, the temperature and pressure levels at which to operate and the equipment sizes are fixed. The calculation is to discover how the equipment performs, a rating calculation.

2) Design. Some outputs from the process and some intermediate stream variable values may be specified in exchange for calculating an equal number of the inputs, levels of operation and/or equipment sizes.

3) Dynamics. The dynamic behavior of a process may be required.

4) Optimization. We may wish to optimize the process over the set of continuous variables that describe equipment sizes and process operating levels.

Figure 4 illustrates the breadth of questions which one can address in the area of design research. Many persons identify design research with only the two aspects highlighted with a dark line: 1) Identify Abstract Problem and 2) Develop Relevant Mathematical Theory. We have been arguing all along about the importance of developing a correct design strategy. Support techniques are often shrugged off as not fundamental enough, but, if not done correctly, the implementation of the theory will likely prove too complex to be practical. Finally one should not overlook the problem of placing sophisticated tools into the hands of unsophisticated users. There is research lurking there too.

The abstract problem for developing ASCEND-II is how to solve large sets of simultaneous nonlinear, sparse algebraic, ordinary and partial differential equations, perhaps subject to inequality constraints and perhaps containing discrete variables. There is certainly enough of a problem here to require considerable effort.

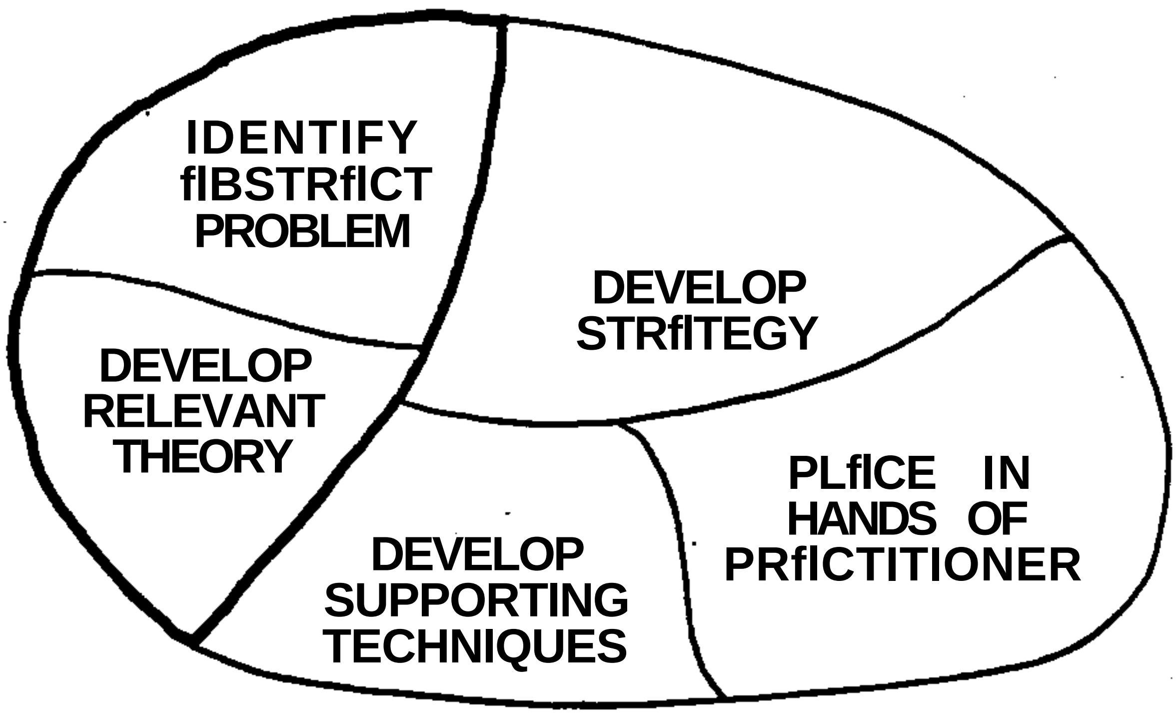

Figure 4. Aspects of Design Research

Relevant theory includes convergence proofs, analysis techniques to take advantage of structure and Lagrange theory. We have already discussed strategy ideas at length. The supporting techniques include consideration of data structures, problem decompositions (see Westerberg and Berna (1978), Berna, et al (1980), Clark (1980)), data bases and use of network computing. Finally the ideas involved in placing the tools into the hands of the practitioner include language design, level of interaction, online documentation system design and use of graphics. We are making considerable progress at dealing with the above ideas and others in the development of ASCEND-II.

figure 5 illustrates the underlying evolutionary aspect of ASCEND-II.

ASCEND-II is intended to help a process engineer "design[1]* a computer model for his process, using the available building blocks provided within tite program. "Design[11] here refers to finding and solving a model of the Heeded complexity to answer the questions being asked of the process, whete the engineer is learning both about the questions he should ask and about the model as he proceeds. We could broaden the meaning of design to that of designing the process for which the model is being developed, a task for which ASCEND-II is also well suited, but we want to limit ourselves here to the narrower model design problem.

the axes in Figure 5 are axes along which the model design can evolve. Model complexity can evolve from simple to complex, where simple models consist of only a few units and the use of the simplest of physical property models — e.g., a flash unit using constant relative volatilities.

With each model a range of analysis type can be performed, starting with simulation, moving to design and finally (when ASCEND-II is further developed) to optimization. Simulation is intuitively the easiest mode to use for the engineer. In that mode he can usually establish a set of

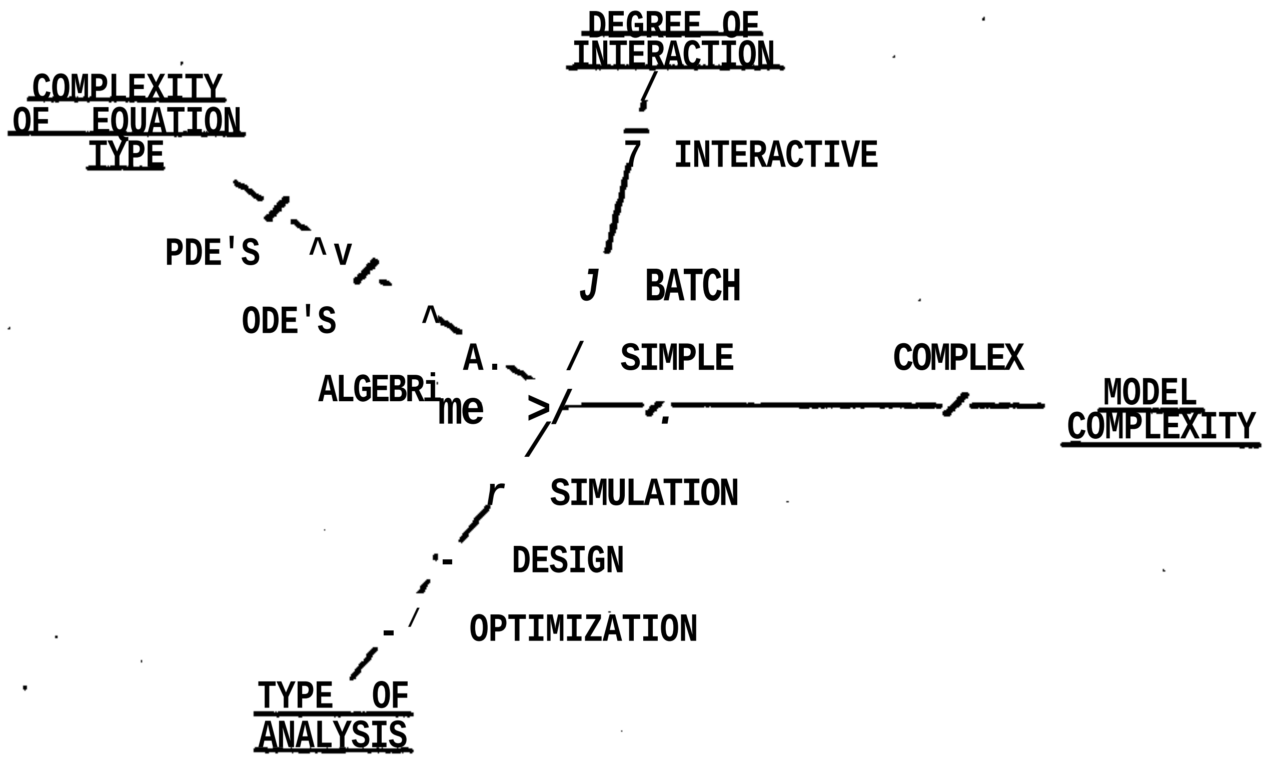

Figure 5. The ASCEND-II Floweheetlng System

specifications which will lead to a solution for the remaining variables. For example, one has some confidence that, if he fixes the feed stream to a flash unit, fixes the fraction of the feed which will vaporize and the flash pressure, then the flash unit will have to operate and so will the corresponding calculation. Why not allow the user to start them with this "comfortable" calculation? Once he can simulate the flash unit ASCEND-II allows him to alter the set of variables to be specified. For example, he could require that the recovery fraction of one of the components be specified and that the pressure be calculated. If the trade is illegal, he will be warned immediately.

Running through a few design calculations will acquaint him with the shape of the solution space, and when he gets near to a good solution, he can switch into doing an optimization calculation.

Once this sequence is solved using simple algebraic models, he can selectively add more complexity to the model by adding more units and/or more sophisticated physical property calculations and continue.

A type of complexity which can be added is to broaden the type of equations which are used to model portions or all of the process, i.e., by allowing models involving ODE's (Kuru (1981)) and PDE's to be introduced. With ODE's and PDE's one can consider doing dynamic studies.

The last axis is that reflecting the degree of interaction ASCEND-II will have with the user. In ASCEND-II a standard command file can be created which will attempt to solve any model once it is set up. Invoking this "standard" file is like running the problem in batch mode on a computer. At the other extreme, the commands can be executed interactively one at a time in a fairly arbitrary order. (The computer is a DEC-20 which provides a very friendly interactive environment.)

Examples of the types of commands available are 1) to input some more structure to the flowsheet, 2) to delete some of the existing structure, 3) to save and retrieve variable values, 4) to initialize variable values (selectively), 5) to change the set of variables whose values are to remain fixed, 6) to cause variables and equations to be rescaled to reflect current variable values, 7) to do one or more Newton-Raphson iterations, 8) to determine the constrained derivative of one variable with respect to another, 9) to display variables selectively, and 10) to display equation errors selectively.

With this structure for ASCEND-II, the user can "drive" his computation around computational obstacles much as he drives a car and can become very effective at getting solutions quickly, even for stubborn problems.

We have set the stage now to argue that ASCEND-II allows the model for a given process to be designed using our earlier guidelines. Clearly the first guideline is dealt with: evolving from simple to complex. The depth first strategy can be followed by developing first a simple model for the entire process.

Unlike conventional flowsheeting systems, each unit within a flowsheet can be tested by itself in ASCEND-II, permitting a bottom up solving of the units at any time. In this mode and using the simple model as the base case design, much testing can be done to see where to add complexity, and where perhaps to remove complexity. Answers can be obtained to a simpler version of the problem to use as a starting solution point for the more complex versions, a strategy often needed when solving highly nonlinear equations. The notion of developing and using approximate criteria is also possible. One usually gets a solution to the equations, perhaps far from the desired solution point. This solution may be from doing a simulation rather than the desired design calculation. Not unlike the idea behind a continuation method, the calculation can be converted to the desired design calculation in terms of which variables are specified. Then one can move to the solution point desired through a series of small steps, converging to the solution at each step.

While it is obvious that much of the power of a program like ASCEND-II comes from its being interactive, it is equally as obvious when using it that the ability to find a base case solution and then to move from that solution in almost any manner desired (top down/bottom up, simulation/design, etc.) is the heart of the rest of its power. It is the learning that can occur which helps to decide the nature of the next calculation, to see its impact and to alter one's path as a consequence, that makes ASCEND-II so useful. Traditional flowsheeting systems (and for that matter, traditional equation solving packages) do not offer the flexibility provided by ASCEND-II for this approach.

ASCEND-II has been designed under the assumption that calculations will often fail until one learns about the problem. Diagnostic tools are thus provided to allow the user a chance to detect where the failures are occurring. As mentioned only briefly before, these include interactive access at any time to <u>every</u> variable in the problem by a convenient name and similarly to every' <u>equation error</u>. This latter access allows one to note, for example, that the phase equilibrium equations on stage 3 of the diethyl ether column are not converging. The variables around that stage can then be examined to see if one is perhaps too large or worse yet, negative. Having located the problem, the user can then start to work on correcting it.

Interestingly, the current version of this system is a third generation version. We designed ASCEND-II following our guidelines by prototyping it twice, at each step improving the design based on the previous version* This was and remains a deliberate policy for creating ASCEND-II.

Heat and Power Integration of a Process

The last process problem to be considered is to integrate the heat and power requirements for a process for which one has just set temperature and pressure levels for each of the units and has solved the process heat and material balances (using ASCEND-II for example). Great progress has been made for solving this problem. The heat integration portion is usually called a heat exchange network synthesis problem. See Nishida et al (1981) for an extensive review of the heat exchanger network synthesis problem.

The heat exchanger network synthesis problem epitomizes the effective use of approximate criteria to locate excellent final network designs. Using thermodynamic arguments, one can predict a priori the least amount and kind of utilities needed to solve this problem. Also using graph theoretic ideas one can guess the fewest number of heat exchanger units likely to be needed. Experience has shown that the better designs meet these goals, or come very close to meeting them. Finally, effective design techniques exist to aid one to find such designs.

In preanalyzing the heat integration problem, one discovers for most problems a bottleneck will occur to further heat integration in the form of a temperature pinch. Figure 6 illustrates the way Hohmann (1971) located this pinch. He merged all hot streams into a single "super" hot stream and all cold into a single "super" cold stream. Placing them as illustrated on a temperature versus total enthalpy diagram reflects the

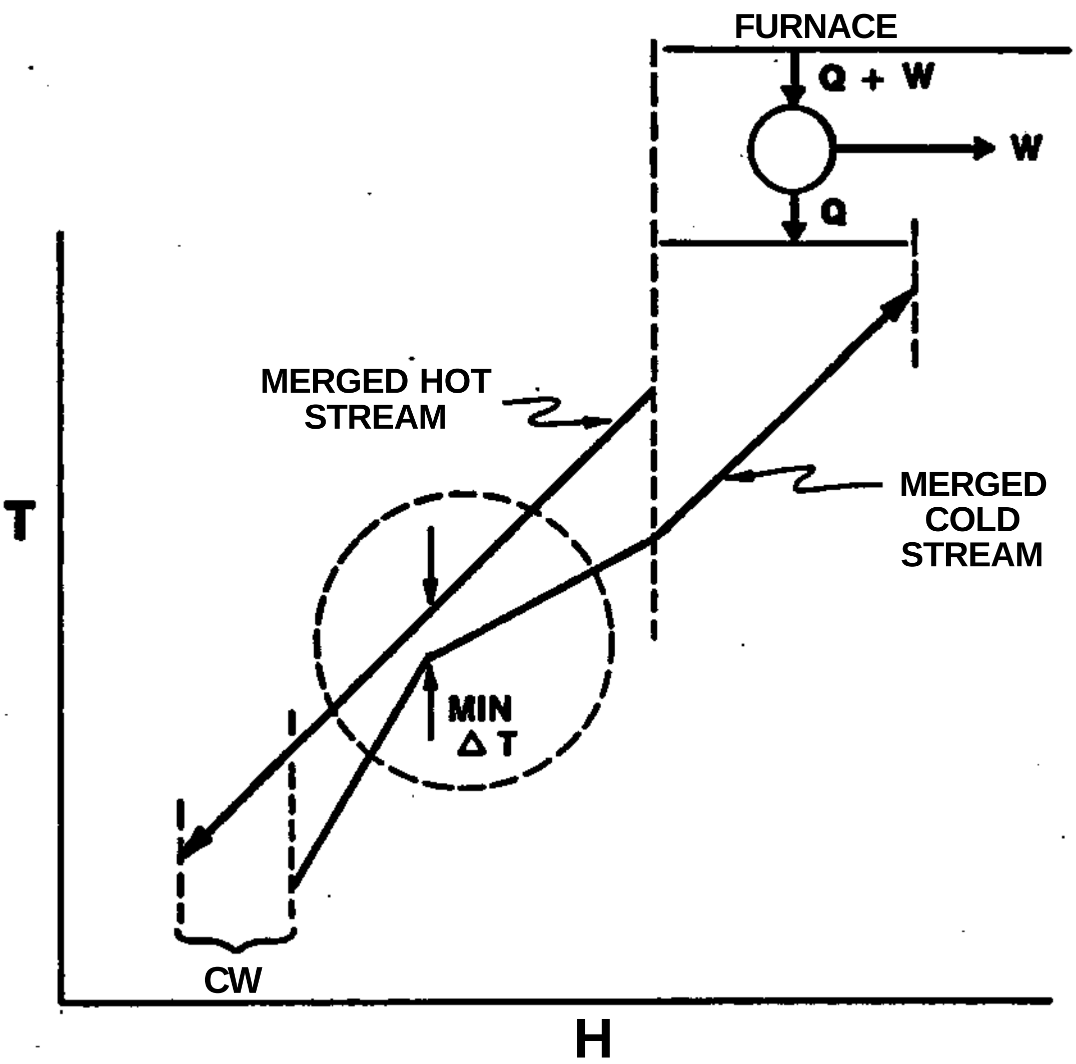

Figure 6. Minimum Utility Calculation. Upper right illustrates how to extract "work" efficiently from process.

opposing hot and cold stream temperature profiles one would see if these super streams met in a single counter current heat exchanger. The pinch is the point which precludes further integration. Cerda (1980) from our group and, in parallel, Mason and Linnhoff at ICI recently developed an approach which generalizes this minimum utility calculation.

Umeda et al (1979) have exploited the pinch to aid in locating where in a process the reestablishing of temperature and pressure levels will permit more heat integration. Very recently Linnhoff and coworkers (Townsend and Linnhoff (1981), Dunford and Linnhoff (1981)) have shown how to exploit large temperature differences between these super streams which occur either entirely above or entirely below the pinch. They show how to convert heat entirely to mechanical work or obtain some "free" separation work within a process. For example, the upper right part of Figure 6 shows how one can place a turbine to get 100% of the thermal energy which must be added into the process converted to the desired mechanical work. The cost is the degrading of the thermal energy which enters and is later rejected by the turbine. If that energy can be degraded and still be rejected at a temperature where it is useful as heat input to the process <u>and</u> if that heat can be extracted and rejected entirely above or entirely below the pinch, then 100% of the extra energy added to drive the turbine is converted to work.

The design strategy is to establish first a process design not yet heat integrated. Then by examining the process, one finds the pinch temperature and predicts the minimum utility costs associated with the process. Next one can modify the process near the pinch if further heat integration is desired. Finally one can place some turbines if possible so they degrade thermal energy either entirely above or below the pinch. The design can then be reassessed and improved from this thermally integrated base case.

Proving the Strategy Itself

How would one "prove" that the design guidelines are basically sound? That problem is itself a design problem and should be (if we are correct) solved using a strategy consistent with the guidelines themselves. The concept should be recursive. In our case it is leading to the design and testing of the ASCEND-11 system. We are only at present proving we are right by demonstrating how rapidly one can put together a working computer model for a process using this system. In one example a model was constructed using a conventional flowsheeting system and the exercise took two full time days. Using ASCEND-II, it took two hours.

Since teaching these guidelines to our students in the undergraduate design course, we see a noticeable reduction in the time needed to get realistic designs.

In Conclusion

The guidelines suggested to aid one to do design more efficiently have been illustrated on several diverse problem types. Only qualitative "proof" exists as to their correctness. If correct a principal use can be to examine a proposed or existing design tool (or design effort) to see if it abides by them. Where it fails should suggest modifications to the tool which could significantly change its effectiveness. Designers have to make a conscious effort to stick to the guidelines as they do not always coincide with the most natural approach. They can be taught; we try to do so in the undergraduate design class.

Literature Cited

Berna, T.J., M.H. Locke and A.W. Westerberg, "A New Approach to Optimization of Chemical Processes," AIChE J., 2£(1), pp 37-43 **(1980)**.

Cerda, J., Transportation Models for the Optimal Synthesis of Heat Exchanger Networks, Ph.D. Dissertation, Carnegie-Mellon Univ., **Pittsburgh**, PA 15213 (1980).

Clark, P.A., An Implementation of a Decomposition Scheme Suitable for Process Design Calculations, M.S. Thesis, Carnegie-Mellon Univ., Pittsburgh, PA 15213 (1980).

Douglas, J.M., Manuscript for Process Design Text (1979).

Dunford, H.A. and B. Linnhoff, "Energy Savings by Appropriate Integration of Distillation Columns into Overall Processes," Paper No. 10, Cost Savings in Distillation Symposium, Leeds, July 9-10, 1981.

Hohmann, E.C. Optimum Networks for Heat Exchange, Ph.D. Dissertation, Univ. of So. Calif. (1971).

Kuru, S., Dynamic Simulation with an Equation Based Flowsheeting System, Ph.D. Dissertation, Carnegie-Mellon Univ., Pittsburgh, PA 15213 (1981).

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.

Locke, M.H., Computer-Aided Design Tools Which Accommodate an Evolutionary Strategy in Engineering Design, Ph.D. Dissertation, Carnegie-Mellon Univ., Pittsburgh, PA 15213 (1981).

Motard, R.L. and A.W. Westerberg, Process Synthesis, AIChE Advanced Seminar Lecture Notes, AIChE, New York, NY (1978).

Nishida, N.,* G. Stephanopoulos and A.W. Westerberg, "A Review of Process Synthesis,[11] AIChE J;, £7(3), pp 321-351 (1981).

Seader, J.D. and A.W. Westerberg, "A Combined Heuristic and Evolutionary Strategy for Synthesis of Simple Separation Sequences,[11] AIChE J., 2M6), pp 951-954 (1977).

Simon, H.A., The Sciences of the Artificial, MIT Press, Cambridge, MA (1969).

Townsend, D.W. and B. Linnhoff, "Heat and Power Networks in Process Design - Part 1, Criteria for Placement of Heat Engines and Heat Pumps in Process Networks," submitted for publication (1981).

Umeda, T., K. Niida and K. Shiroko, "A Thermodynamic Approach to Heat Integration in Distillation Systems," AIChE J., 2M3), pp 423-429 (1979).

Westerberg, A.W. and T.J. Berna, "Decomposition of Very Large-Scale Newton-Raphson Based Flowsheeting Problems," Computer and Chemical Engng. J., £, pp 61-63 (1978).