V_2 = velocity of slowest drops, cm./sec.

Greek Letters

α = ratio of core radius to column radius

σ^2 = variance of residence time distribution of drops in the whole column

$\sigma_p{}^2$ = variance of residence time distribution of drops in the core

$\sigma_{pt}{}^2$ = variance of residence time distribution of drops in the core related to the average residence time of drops in the whole cross section area

$\sigma_{bp}{}^2$ = variance of residence time distribution of by-passing drops

LITERATURE CITED

1. Angelo, J. B., and E. N. Lightfoot, *AIChE J.*, **14**, 531 (1968).
2. Greskovich, E. J., Ph.D. dissertation, Pennsylvania State Univ., University Park (1966).
3. Hazelbeck, D. E., and C. J. Geankoplis, *Ind. Eng. Chem. Fundamentals*, **2**, 310 (1963).
4. Hiby, J. W., *Proc. Symp. Inter. Fluids Particles*, London, 313 (June, 1962).
5. Kehat, Ephraim, and Ruth Letan, Progr. Rept., Israel National Council for Research and Development (Jan. 1967).
6. Kintner, R. C., "Advances in Chemical Engineering," Vol. 4, p. 52, Academic Press, New York (1963).
7. Letan, Ruth, D.Sc. dissertaton, Technion, Israel Inst. Tech., Haifa (1966).
8. ——— and Ephraim Kehat, *AIChE J.*, **13**, 443 (1967).
9. *Ibid.*, **14**, 398 (1968).
10. Levenspiel, O., "Chemical Reaction Engineering," John Wiley, New York (1962).
11. Liles, A. W., and C. J. Geankoplis, *AIChE J.*, **6**, 591 (1960).
12. Miller, S. F., and C. J. King, *ibid.*, **12**, 767 (1966).
13. Mixon, F. O., Whitaker, D. R., and J. C. Orcutt, *ibid.*, **13**, 21 (1967).
14. Vermeulen, T., Moon, J. S., Hennico, A., and T. Miyauchi, *Chem. Eng. Progr.*, **62**, (No. 9), 95 (1966).

Manuscript received August 18, 1967; revision received December 14, 1967; paper accepted January 10, 1968.

The Synthesis of System Designs

II. Heuristic Structuring

A. H. MASSO and D. F. RUDD

The University of Wisconsin, Madison, Wisconsin

The use of heuristic structuring strategies in the development of computer programs for the synthesis of process designs is examined. By the employment of selection weights which are adjusted as experience is gained from past successes and failures, the computer is able to learn the sequence of structuring decisions which leads toward the optimal process design. The computer can develop competence in the synthesis of systems in a limited area of technology.

A basic problem in engineering is that of synthesizing from available components a system which best performs a processing task. The trial and error synthesis of all possible system structures may be ruled out as a plausible design method; for, in all but the simplest of design problems, the number of ways in which the system might be structured is overwhelmingly large. In a previous report (*1*) the principle of problem decomposition was examined and shown to hold promise as a practical means of system synthesis. In this report, we examine a different approach and relate experiences in the development of a program which enables the digital computer to gain competence in the synthesis of heat exchanger networks. Although this work is still in a primitive stage, it holds promise of increasing the usefulness of the comupter in process design.

A dilemma is encountered in the synthesis component by component of a performing process. One cannot know how the addition of a component to the partially synthesized system will affect the system performance because the complete system does not yet exist. Thus, there can be no strictly valid criterion for selecting one component over another, or one structural arrangement over another while in the midst of system synthesis. The lack of a valid criterion leads directly to the enormous combinatorial problem of synthesizing all possible structures. Selection rules which favor the use of a given piece of equipment in certain phases of system synthesis evolve from experience and are thought to be part of the empirical skill of successful process designers. These rules may be wrong on occasion and will lead to nonoptimal systems, but the experienced designer requires only that the rules lead to efficient designs frequently enough to warrant their use.

Overwhelming problems which defy analysis arise in areas other than engineering, such as in the playing of chess or the proving of theorems in mathematics. The mental processes of the experienced chess player or the experienced mathematician seem to involve the use of rules of thumb that circumvent problems in analysis which are beyond detailed solution. These rules are called *heuristics*, and are described as methods of problem solving which are useful empirically but are unproved, or incapable of being proved.

This is a report on experiences in the development of heuristic structuring programs which enable the computer to learn to synthesize system structures. The general procedure is as follows. A basis for incorporating particular processing steps in the system structure is obtained from various design criteria with which the computer is supplied. Synthesis of the required process, then, results from the selection of a sequence of such steps. Hence, the problem is that of enabling the computer to teach itself to select the optimal sequence of design decisions. This is done by assigning selection weights to the criteria, and adjusting these weights to assimilate the experience gained through previous synthesis trials. The design criteria plus the means of manipulating the selection weights constitute the heuristic structuring rules.

The concepts and nomenclature used here are consistent with part I of these studies (*1*). In the section that follows, the relationship between heuristic structuring and problem decomposition is given. Then, the heat exchanger network problem is presented, and its synthesis procedure is outlined in detail. With this background, the solutions of sample problems are presented with a review of the important results.

THE HEURISTIC STAGE-BY-STAGE STRUCTURING DECISION CONCEPT

The contention that synthesis may be gainfully approached by the development of techniques for progressively breaking large problems into smaller, more manageable, subproblems is reiterated in this report. Hence, a solution approach to the synthesis problem is thought to involve the making of series of decisions which will progressively clarify the underlying topology of the system sought. In any event, the end product of this decision process should be a structured system with a measure of its economic performance.

Searching for Subproblems

During solution, decisions may be allowed to result in (a) the generation of new problems that are smaller than the original problem but for whose solutions further reductions may be required, or (b) the isolation of segments of the original problem whose features allow their immediate solution with the technology that is available. The first case is precisely the approach by decomposition outlined in the first report by Rudd (*1*). The second case represents a decided conceptual simplification of the decomposition procedure as summarized below.

In synthesis by decomposition, it is assumed that the means are available for estimating and updating this estimate of the optimum objective function, $O^*(\mathbf{X})$, for the design task $\mathbf{X}$. The original synthesis problem defined by the task constraints $\mathbf{X}$ is decomposed into, say, two smaller subproblems I and II defined by the constraints $\mathbf{X}_I \cup \mathbf{T}$ and $\mathbf{X}_{II} \cup \mathbf{T}$ where, as before, $\mathbf{X} = \mathbf{X}_I \cup \mathbf{X}_{II}$, $\mathbf{X}_I \cap \mathbf{X}_{II} = 0$ and $\mathbf{T}$ is a set of artificially imposed tear constraints. Next, the following problem is solved

$$O^{(i)}(\mathbf{X}) = \underset{\mathbf{X}_I,\,\mathbf{X}_{II}}{\text{Opt}} \left\{ \underset{\mathbf{T}}{\text{Opt}} \left(O^{(i)}(\mathbf{X}_I \cup \mathbf{T}) + O^{(i)}(\mathbf{X}_{II} \cup \mathbf{T}) \right) \right\} \tag{1}$$

where $O^{(i)}(\mathbf{X})$ is the estimate of the optimum objective function for the current level of decomposition after i trials. Whether further decomposition is required depends on the solubility of subproblems $\mathbf{X}_I \cup \mathbf{T}$ and $\mathbf{X}_{II} \cup \mathbf{T}$. Now, should the system have been completely synthesized $O^{(i)}(\mathbf{X})$ may still only be an estimate since the numerical values of all the tear variables may not be optimal, or the subproblems selected may not have been those conducive to the generation of the optimal arrangement. At the end of the ith trial, then, the available system performs according to an exact function $O_{\text{act}}(\mathbf{X})$ which by the additional operation

$$O^*_{\text{act}}(\mathbf{X}) = \underset{\mathbf{T}}{\text{Opt}} \left\{ \sum_{I} E(\mathbf{X}_j) \right\} \tag{2}$$

may indicate how to improve this system's performance by further adjustment of the tear variables. Finally, it is suggested that should $O^{(i)}(X)$ not compare favorably with $O_{\text{act}}(\mathbf{X})$ and $O^*_{\text{act}}(\mathbf{X})$ at all levels of decomposition, further iteration is called for.

Suppose, now, that the convention is made that decomposition of the original problem (or any unsolved part of it) produces subproblems at least one of which is immediately soluble with the available knowhow. Symbolically, some subtask $\mathbf{X}_j$ is sought which satisfies the relation $(\mathbf{X}_j \cup \mathbf{T}_j) \subset \mathbf{R}$, where $\mathbf{R}$ represents the region of existing technology such that there is available an exact economic evaluation $E(\mathbf{X}_j)$. Structuring now proceeds according to

$$O^*(\mathbf{X}) = \underset{(\mathbf{X}_j \cup \mathbf{T}_j) \subset \mathbf{R}}{\text{Opt}} \left\{ \underset{\mathbf{T}_j}{\text{Opt}} \; \{ O^*((\mathbf{X}_j \cup \mathbf{T}_j) \subset \mathbf{R}) + O^*(\overline{\mathbf{X}}_j \cup \mathbf{T}_j) \} \right\} \tag{3}$$

where $\overline{\mathbf{X}_j}$ is the complement of $\mathbf{X}_j$, that is, $\mathbf{X} = \mathbf{X}_j \cup \overline{\mathbf{X}_j}$, $\mathbf{X}_j \cap \overline{\mathbf{X}_j} = 0$. The implications of such a modified procedure make decomposition more tractable. Clearly, the requirement that subtask $\mathbf{X}_j \cup \mathbf{T}_j$ be immediately soluble aids in the selection of the tear variables $\mathbf{T}_j$. Furthermore, the conceptual basis of decomposition now states that optimal structuring proceeds according to an optimal sequence of soluble subtask selection decisions. However, the exterior search for the optimal subtask selection sequence requires further comment. Prior procedures have called for an estimation of the optimal objective function $O^*(\mathbf{X})$. The availability of such an estimate is mandatory because it is to serve both as a basis for decision making, and as a means of storing experience developed in the solution of a particular problem. Since it is not yet understood how to go about constructing the estimating function, and since there is no updating formalism available that will in some way direct the search to convergence, an alternative basis must be employed.

Heuristic Search Direction

Rather than estimating the consequence of a structuring decision as would be done by operation with the functions $O^{(i)}(\mathbf{X})$, it is proposed that structuring decisions be made on some external basis, and that exact measures of their economic consequence be obtained by virtue of the modified decomposition scheme outlined above. Since at every decision stage a soluble subproblem is to be isolated and solved, the idea is now to provide the means for tabulating, and selecting from the candidate subproblems. This is where the heuristic structuring strategies come into use.

Logical structuring rules that have proved or are thought to be useful design criteria in some processing area are applied. In the synthesis of heat exchange networks, for example, these rules would indicate what heat exchange, heating, or cooling operation to do next. Each of the rules available would be weighted, and the set of rules would be sampled at every level of decomposition to specify a single basis on which the next soluble subproblem is isolated. As a particular sequence of structuring decisions is found to lead to an arrangement which cannot be further improved upon, the selection weights reflect this by favoring those rules which lead to such an arrangement. Likewise, rules leading to unfavorable decisions are gradually suppressed by their acquisition of low weights. The gathering of experience embodied in improved estimates of the optimum objective function is thus reflected in the distortion of the selection weights. Furthermore, every design policy defined by a sequence of structuring decisions yields an exact objective $O^{(i)*}_{\text{act}}(\mathbf{X})$ after every trial. Eventual convergence of $O^{(i)*}_{\text{act}}(\mathbf{X})$ to the optimum

$O^*(X)$ now depends on the propriety and effectiveness of the selection rules in use.

Given the stage-by-stage procedure, the logic may be developed for synthesizing feasible systems. What needs further elucidation, however, is the mechanism by which the search will be directed toward the system with the most desirable performance. Whereas the details of this are relegated to a later section, the characteristics of the mechanism to be proposed are now summarized.

As stated earlier, the intent is to employ the computer to carry out the required synthesis. If the problem of arriving at an optimal structure were a purely mechanical task requiring no experience or decision making ability, it would be a relatively simple matter to program the required instructions. Likewise, if there were but one feasible structure, no search techniques would be needed; only the logic necessary to generate this structure need be programmed. The situation at hand requires the development of a true problem-solving routine having definite properties. These are: (a) a set of rules (heuristics) are to be used to generate feasible alternative systems; (b) sequential decisions are to isolate soluble subproblems until the original task constraints are met, and the system's performance becomes exactly measurable; (c) the set of feasible systems that could be generated is to be large, but not necessarily exhaustive; (d) the routine must be capable of enhancing its decision making ability by noting what rules result in improved solutions.

THE HEAT EXCHANGER NETWORK PROBLEM

Heat exchanger network design is undertaken to cope with situations where a variety of process streams are to undergo simultaneous state changes achievable by the addition and/or removal of heat. Economic advantages concurrent with the efficient use of the process streams, as heating or cooling media, serve as the incentive for carefully administering the specification of auxiliary equipment and facilities such as water coolers and steam or fired heaters. Under the usual circumstances, however, it is not clear beforehand how to go about matching heat demands and supplies in such a manner as to optimize the performance of the required system. This problem differs from that where the operation of an existing network is being optimized by proper selection of its internal temperatures, flow rates, heat exchanger areas, etc. For the situation at hand, there is no previous information on what the network looks like: there is no available structure. A simplified format for the heat exchanger network problem follows.

Problem Statement

Task Constraints. There is a total of s liquid process streams n of which are to be heated, while the remaining $m = s - n$ streams are to be cooled. Associated with the ith stream are its flow rate, w_i, input temperature, t_i^i, output temperature, t_i^o, and heat capacity, c_i, all in consistent units. The available auxiliary heat transfer media are saturated steam and cooling water. The steam is available at any flow rate at a pressure p_s, and is allowed to give up only its latent heat l_s. Cooling water is also available at any flow rate at a temperature t_w^i, and is allowed to undergo changes up to a maximum temperature t_w.

Unsynthesized System. The unsynthesized system has input information consisting of the stream descriptions given above, and additional data representing further constraints to be listed below.

Existing Technology. The equipment available includes heat exchangers of the shell-and-tube type operating as counter-current, single-pass units. For the fluids and conditions prevailing, average overall heat transfer coefficients U_{HE}, U_H, and U_C are achievable for heat exchange between any two process streams, steam heating, and water cooling, respectively. For heat exchange, heating, and cooling, the minimum allowable approach temperature differences are τ_{HE}, τ_H, and τ_C, respectively. The equipment undergoes maintenance checks and repairs resulting in α hours of downtime per year.

Economics. The economics of the system, or any part of it, are represented by yearly costs and are determined by using the information that follows. Heat exchanger cost as a function of its area is given by a correlation of the form $C = a\,A^b$ where a and b are constants. Cooling water costs C_w \$/lb. and steam costs C_s \$/lb. Operating and other costs are neglected. Total costs are computed on a yearly basis with fixed costs amortizing linearly over a period of M years.

Synthesis Objective. The objective is to structure a system capable of performing the prescribed tasks at minimum yearly costs.

THE SYNTHESIS OF HEAT EXCHANGE NETWORKS

Structuring under the modified procedures just proposed will proceed according to the objective of Equation (3). This is used with the understanding that convergence on $O^*(X)$ is achieved by searching the space of feasible design policies rather than the space of objective functions. Several specific points arise which are discussed within the context of the heat exchanger network problem. First of all, the means for the isolation of soluble subproblems are described. Next, the procedure for conducting internal optimization of the numerical values of the tear variables is given. With this background, the decision making process is sketched, and information transfer from trial to trial is explained. Finally, the working procedure for a single synthesis trial is summarized.

Subproblem Selection

Given a set of process streams whose temperatures are to be changed, the following convention is made: subproblem selection is to yield only heat exchange matches, and will exclude treatment by auxiliary heating and cooling as long as there are process streams in existence that can be feasibly matched in heat exchange. This implies that process stream transformations involving water cooling or steam heating will not arise from *bona fide* structuring decisions. When these do occur, it will be because there are no alternative means of satisfying the pertinent task constraints. In addition, this study is restricted to the manipulation of full process stream flows. That is, a process stream may not be divided into two or more other streams unless this has been done *a priori*, and included as a task constraint by fixing the division fractions. Every decision stage, then, leads to the selection of two streams that are going to be matched, and in the statement of need of a heat exchanger where the physical process is to occur. All of this indicates that the tear variables to be selected and adjusted are intermediate steam temperatures.

Tear Variable Adjustment

For the situation at hand, the internal problem of selecting numerical values for the tear variables could involve search over several variables at every decision stage. Unfortunately, this search is to be conducted in the absence of the objective function $O_{\text{act}}^{(i)*}(X)$ which, as will be recalled, is available only for the completely structured network. Rather than approaching this search problem iteratively, the tear variable is arbitrarily adjusted to guarantee an exchanger duty which is maximized subject to the pertinent task constraints, and the minimum allowable approach temperature difference. This is assumed to be a reasonable approximation. One source of justification lies in the favorable economics of maximized heat exchange as opposed to auxiliary heating and/or cooling.

Rules for Soluble Subproblem Selection

At any point in the solution, the set of streams can be easily examined to determine what pairs may be feasibly matched in heat exchange. Of course, the list of available streams will indicate what task constraints remain to be met, and the computer is faced with the problem of determining what soluble subproblem it should attack next. The rules employed are:
From the current set of feasible matches

1. select the next match at random, or
2. select the match whose feasible heat exchange segment is performed at maximum cost, or
3. select the match which involves streams whose total processing cost by independent heating and cooling is maximum, or
4. select the match which results in unfeasible remainders* whose processing cost by independent heating and cooling is minimum, or
5. select the match which results in a maximum recovery of the available heat to be rejected by the component stream undergoing the exothermic state change.

Inclusion of the random rule requires comment. By the use of a particular set of nonrandom rules, two types of error may be made. On the one hand, the rules may result in the systematic omission of one or more matches. That is, even though these matches are detected as members of the feasible candidate set, the decision maker will not use any of them because none of the rules will lead to their selection. On the other hand, some chance occurrence may lead to the conclusion that a particular selection is optimal when indeed optimality is insensitive to the outcome of decision at that stage. By using the random rule clearly counteracts the error of omission in the former case, while it may point to the insensitivity in the latter. In passing, it might be added that such an artifact is used merely to help relieve the consequences of using impertinent selection rules or inefficient learning schemes.

Convergence Criteria

A synthesis trial yields a completely structured system with its economic performance which gauges the desirability of a sequence of stream matches. Each match in the sequence results from decisions according to some rule. Likewise, the selection of a rule is based on the distribution of weights over these rules. Information flow from trial to trial, therefore, consists of the best current match sequence, its cost, the rule or rules† which lead to selection of the match at each stage. The number of decision stages, or decomposition levels is also carried forth as an implicit part of the information transferred.

Search over the space of design policies is convergent when, for the set of rules in use, a sequence of structuring decisions (matches) yields a cost that cannot be improved upon after some reasonable number of trials. Rather than using this convergence criterion to mechanize the decision of when enough experience has been compiled in the solution of a given problem, the number of trials is arbitrarily preset, and externally altered as results become available.

Summary of Synthesizer Working Procedure

The following steps summarize the procedure followed by the synthesizer:

1. Process streams from the current set are segregated according to whether they are to be heated or cooled.
2. Each stream from one group is matched with each of the streams in the other group to construct a table of candidate feasible matches.
3. A rule (heuristic) is selected according to the weight distribution pertinent to the current stage (level of decomposition).
4. A matched pair of streams is selected according to the rule from step 3 and allocated to a heat exchanger.
5. If the original constraints on the matched pair of streams are not met, only the segments of the transformations which are feasible by heat exchange are carried out. The completed portions of the transformations are deleted from the groupings of streams.
6. Steps 2 through 5 are repeated until no further matches remain.
7. Residual tasks are relegated to auxiliary heating and cooling and the total system cost is computed.
8. The total cost of the current solution is compared to the previous minimum cost observed.‡
9. The weights of the rules used at every stage are decreased if the current solution is more costly than the previous best, otherwise they are increased.§
10. Items for information flow from trial to trial are compiled, and the next trial is begun by restarting at step 1 if the prescribed number of trials has not been completed.

SAMPLE PROBLEMS

The problems studied carried data which included the process stream flow rate in lb./hr., input and output temperatures in °F., and the heat capacity in B.t.u./(lb.)(°F.). Other physical and economic data defined earlier are as follows:

Steam pressure	p_s	450 lb./sq.in.abs. saturated
Cooling water temperature	t_w^i	100°F.
Maximum water output temperature	t_w	180°F.
Minimum allowable approach ΔT's		
heat exchanger	τ_{HE}	20°F.
steam heater	τ_H	25°F.
water cooler	τ_C	20°F.
Overall heat transfer coefficients		
heat exchanger	u_{HE}	150 B.t.u./(hr.)(sq.ft.)(°F.)
steam heater	u_H	200 B.t.u./(hr.)(sq.ft.)(°F.)
water cooler	u_C	150 B.t.u./(hr.)(sq.ft.)(°F.)
Equipment down time	α	260 hr./yr.
Heat exchanger cost parameters	a, b	350, 0.6
Cooling water cost	C_w	5×10^{-5} \$/lb.
Steam cost	C_s	1×10^{-3} \$/lb.

* An unfeasible remainder results from a match where the task constraints on one or both of the streams involved are not met.

† More than one rule may lead to the same decision outcome at some stage or level of decomposition.

‡ An initial estimate of the total cost is obtained by processing all streams by steam heating and/or water cooling.

§ The weights are initially set at 50 (a rule is suppressed by setting its initial weight at 0). Adjustment is made by adding or subtracting 10 from the current weight as required. If a rule has not been suppressed *a priori*, its minimum allowable weight is 5.

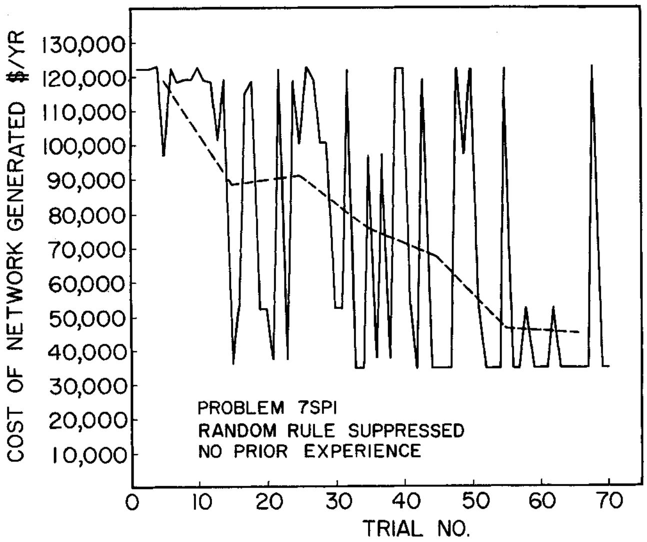

Fig. 1. Convergence on the minimum cost solution.

Problem 5SP1

Stream no.	Flow rate	Input temp.	Output temp.	Heat cap.
1	27,000	100	400	0.80
2	42,000	480	250	0.75
3	35,000	150	360	0.70
4	36,000	400	150	0.70
5	38,000	200	400	0.65

Problem 7SP1

Stream no.	Flow rate	Input temp.	Output temp.	Heat cap.
1	20,000	100	430	0.80
2	40,000	440	150	0.70
3	35,000	520	300	0.68
4	36,000	180	350	0.91
5	31,000	200	400	0.85
6	32,000	350	410	0.62
7	42,000	390	150	0.80

Problem 7SP2

Stream no.	Flow rate	Input temp.	Output temp.	Hea. cap.
1	20,000	200	400	0.80
2	20,000	100	430	0.80
3	27,000	590	400	0.88
4	48,000	300	400	0.86
5	19,000	471	200	0.83
6	41,000	150	280	0.64
7	22,000	533	150	0.60

SUMMARY OF RESULTS

Experience gained from the solution of the sample problems is summarized in the sections that follow. First of all, an attempt is made to further clarify the nature of the search spaces under consideration. Next, the problem of converging on the minimum cost solution is revisited before examining the computer's manifestations of learning. Finally, the evolution of the networks synthesized is traced after presenting the outcome of an experiment to determine the usefulness of learned synthesis policies.

Topology of the Search Spaces

The concept of stage-by-stage decision making can be exploited to provide a simple picture of the search problem being studied. Assuming that all the selection rules are active, decomposition at level l is tantamount to selection from among r_l candidate stream matches. Through the various levels, this number of candidates generally decreases, with the last level involving only the last two possible matches. For an unsolved problem, the required number of decomposition levels is unknown. Likewise, the effect of particular match selections on the required number of decompositions is not ascertainable beforehand. If L represents, on the average, the number of decomposition levels required, and r_l now represents the average number of candidate matches at level l, the number of distinct networks that may be generated is approximated by $\prod_{l=1}^{L} r_l$. For example, problem 7SP1 required an average of four decompositions, and it is estimated that no less than 50, while possibly as many as 90 distinct networks may be generated. It is understood that these distinct networks all satisfy the task constraints, and that under the procedures outlined earlier, structurally different arrangements are implied.

The above gives some idea regarding the enumerability of the elements in the space of synthesis policies. If these elements are now mapped in a space reflecting their economic performance, additional features are visible. The continuity of observable costs is of particular interest. First of all, there will be some voids naturally generated by the nonexistence of feasible networks at costs in some ranges. In addition, and as has already been mentioned, synthesis may fail to lead to feasible solutions within given cost ranges because of the selection rules in use. The points of real importance are that natural spatial irregularities complicate the development of efficient means of directing the search; and that improper rules may lead only to low cost but nonoptimal solutions.

Convergence on the Minimum Cost Solution

Quite simply, if the proposed procedure is to be considered successful, it must converge on a low cost solution after a reasonable number of trials. Furthermore, aside from inevitable erroneous excursions, it must lead to a repetitive generation of the best solution it can produce, until it is stopped. Indeed, such is the case illustrated in

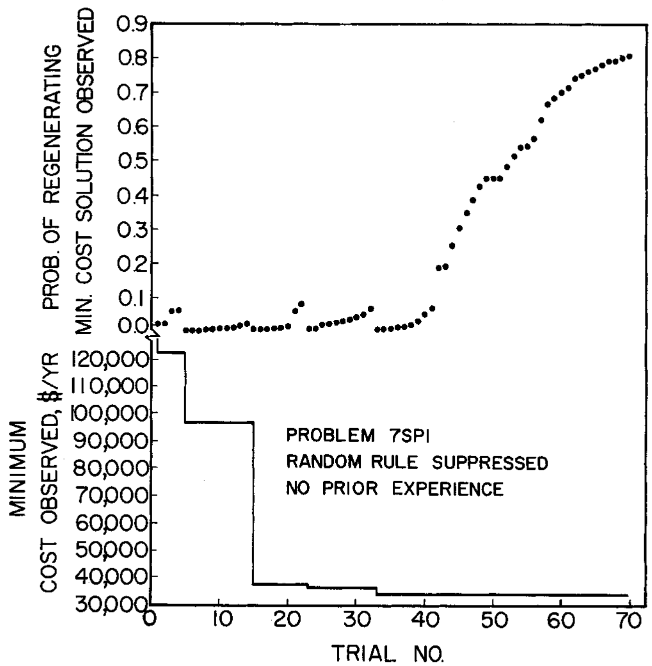

Fig. 2. The learning curve.

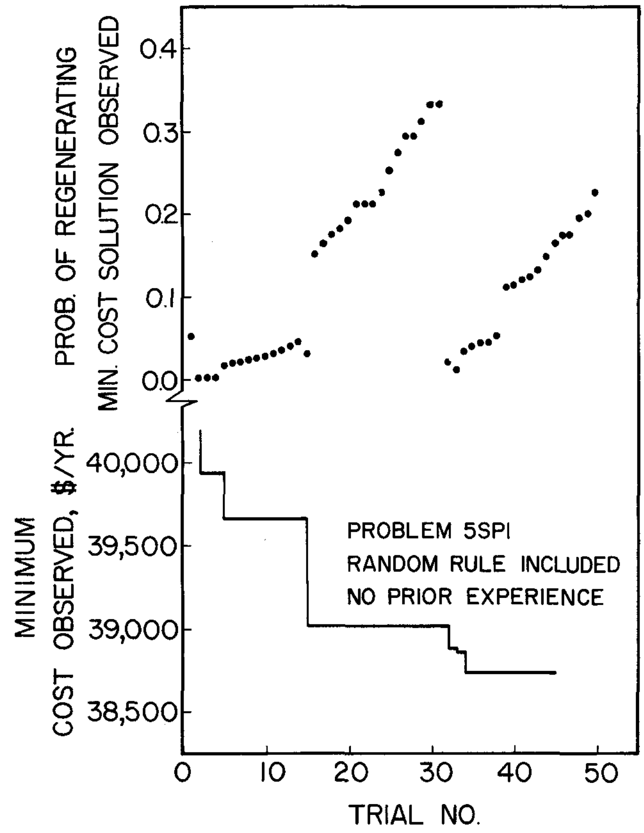

Fig. 3. The recovery power of the heuristic technique.

Figure 1 where the dotted line points to a steady decline in the average cost observed over successive blocks of trial solutions. In this light, a question of real importance pertains to the propriety of the random match selection rule. If the nonrandom rules proposed are not conducive to reliable convergence on the optimum network configuration, the random rule should, on occasion, prove to be useful. Briefly, inclusion of the random rule did not improve on the solutions otherwise obtained for problems 7SP1 and 7SP2. However, for problem 5SP1 inclusion of the random rule was essential for the generation of the minimum cost solution. This was unattainable under any other conditions because, at the second level of decomposition, the optimal match was erroneously omitted from consideration by the nonrandom selection rules.

Learning

A very important indication sought from the results is the achievement of learning. Basically, convergence and learning go hand in hand. Both are manifested in the eventual repetitive generation of a particular solution. That is, both are the result of selecting a synthesis policy as being better than the rest. Nevertheless, learning is treated separately because of interest in its characteristics as indicators of the path and rate of convergence. The information needed is obtainable from a learning curve to be described below.

Ideally, every trial should yield a reduction in the cost of satisfying the task constraints. Unfortunately, with no prior experience in the solution of a particular problem, the synthesizer will show some initial meandering. With the likelihood of generating undesirable solutions being high, as would be reflected by the relatively undistorted selection weights, considerable hunting prevails initially as shown by Figure 1. This continues even after the eventual minimum is observed since this solution will be in jeopardy until there is sufficient evidence in its favor. Practically speaking, then, a plot of the minimum observed cost versus trial number yields a series of descending steps. The width, or duration of each step indicates the difficulty with which a reduction in cost becomes viable. To better show how confidence in the current best solution might build, an additional characteristic is superimposed on this plot.

Since it is possible to record the selection rules employed in the definition of any synthesis policy, it is also possible to determine the probability with which a particular solution may be regenerated in the next trial. This probability is calculated in the obvious manner from the pertinent selection weights. A completed learning curve now includes the minimum cost observed, and the probability with which it will be regenerated in the next trial, plotted against the trial number. Figure 2 shows typical results obtained from the solution of problem 7SP1.

An important consideration rests on what is termed the recovery power of the heuristic technique. Results from the solution of problem 5SP1 which are illustrated in Figure 3 offer a case in point. For this problem, the solution first obtained at trial number 15 proved to be sufficiently persistent to accumulate considerable weight in its favor. Furthermore, after 15 additional trials, its regeneration probability had grown to over 0.34. This misplaced confidence on the part of the synthesizer was mainly due to erroneous decisions at the second level of decomposition. As has already been pointed out, a chance discovery of the optimal subproblem selection at the second level corrected the situation. This power of recovery is also evident in Figure 2, although in a less dramatic fashion.

Tables 1 and 2 are offered to illustrate the distortion of the selection weights after given numbers of trials, and to indicate the optimal rule sets found for each problem.

The Usefulness of Prior Optimal Synthesis Policies

An experiment of interest relates to the use of experience gained from the solution of one problem as an aid in the solution of a new problem. The weights obtained after 60 trial solutions of problem 7SP1 were ap-

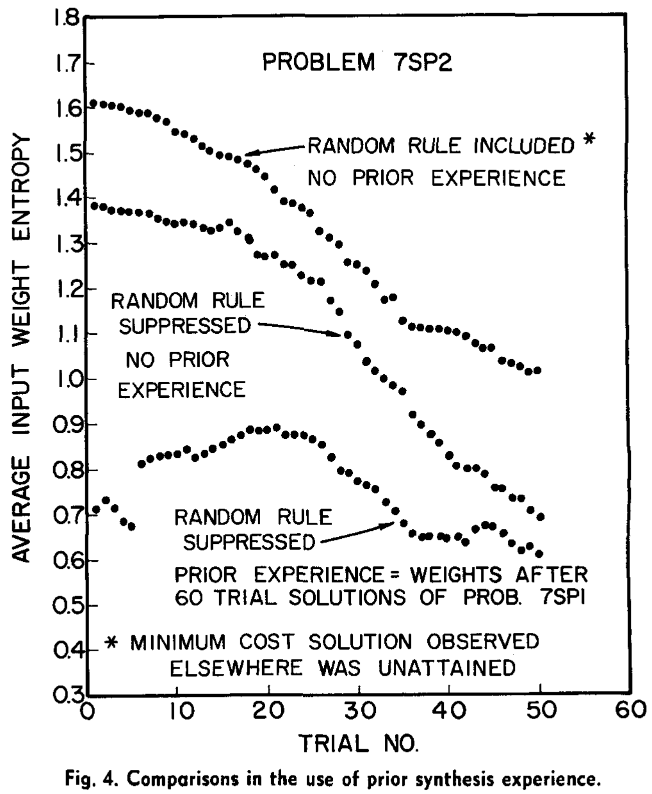

Fig. 4. Comparisons in the use of prior synthesis experience.

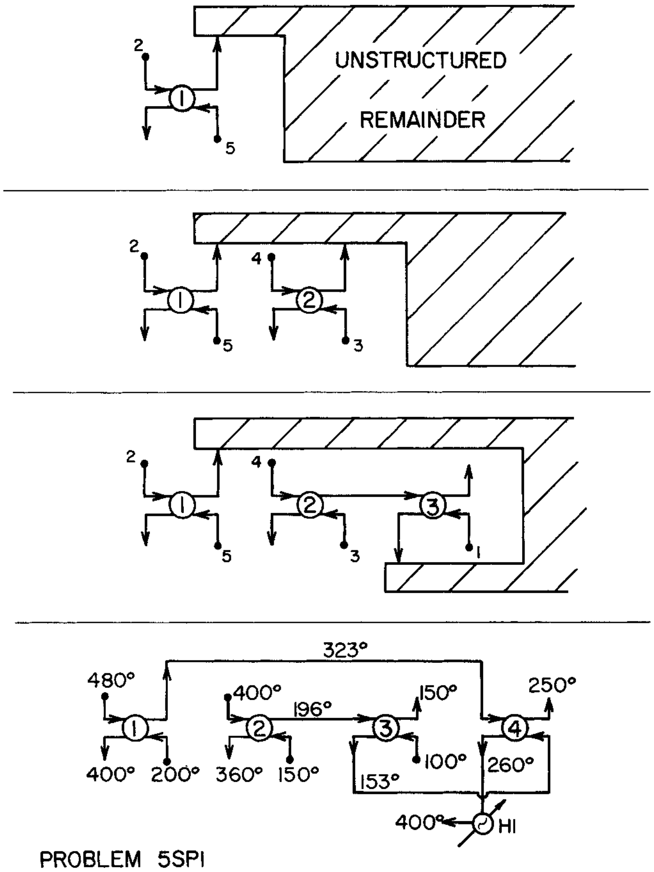

Fig. 5. Evolution of a network; problem 5SP1.

Table 1. Weight Distortions

Rule / Stage	1	2	3	4	5
	10	70	5	90	30
1	0	160	5	10	5
	0	60	110	5	5
	110	5	10	60	5
2	0	10	130	5	5
	0	130	30	5	5
	60	60	30	30	5
3	0	40	110	5	5
	0	150	5	10	40
	60	20	110	20	50
4	0	20	70	5	100
	0	170	20	5	10
	40	80	40	50	40
5	0	40	90	110	60
	0	5	5	140	5
	50	50	50	50	50
6	0	50	40	50	50
	0	70	20	30	130
	50	50	50	50	50
7	0	50	50	50	50
	0	40	40	50	50
	50	50	50	50	50
8	0	50	50	50	50
	0	50	50	50	50

KEY: Problem 5SP1 after 45 trials for problem 5SP1.
Problem 7SP1 after 60 trials for problem 7SP1.
Problem 7SP2 after 50 trials for problem 7SP2.

plied to problem 7SP2. Here, the use of prior experience proved to be helpful. To verify this, problem 7SP2 was rerun with uniform initial weights, and with both inclusion and suppression of the random rule. Neither of the latter two cases produced a better solution than that obtained before. Instead of noting all the details, Figure 4 is submitted to summarize the nature of potential benefits which accrue from the use of prior information. The average input weight entropy calculated over the active decision stages is plotted against the trial number for the three cases. For the first case, the initial rise in the average entropy is due to some readjustment of the weight distributions and the acquisition of new information on how to make an extra decision required for the solution of problem 7SP2.

Looking in retrospect, should the solution to problem 5SP1 have been attempted using prior experience from the solution of problem 7SP1 or 7SP2, or vice versa, an altogether different conclusion may have been drawn. A brief look at Tables 1 and 2 would indicate that the synthesizer would have had to discard considerable amounts of useless prior experience, and acquire new information to handle the disparities in the required levels of decomposition.

Table 2. Optimal Rule Sets

Rule / Stage	1	2	3	4	5
				5SP1	
1		7SP1			
		7SP2	7SP2		
	5SP1				
2			7SP1		
		7SP2	7SP2		
	5SP1	5SP1			
3		7SP1	7SP1		
		7SP2			7SP2
4		7SP1	7SP1		7SP1
		7SP2	7SP2		
5				7SP2	

The Evolution of Synthesized Networks

Figure 5 shows the generation of the minimum cost network observed for problem 5SP1. The outcomes of the three decision stages are presented indicating the streams that were matched. It is noted that at each stage, stream temperatures were completely determined. Synthesis was finalized upon specification of the steam heater for a stream remainder that could enter into no further matches. With the random rule suppressed, the best solution after 45 trials is that shown in Figure 6. Although the difference in costs is on the order of 0.5%, there is a decided advantage in the structural simplicity of the net-

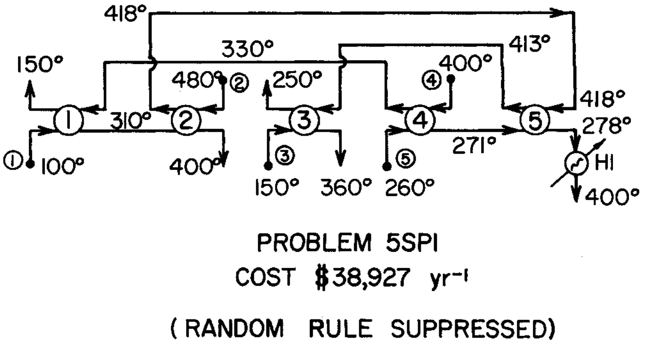

Fig. 6. A nonoptimal solution of problem 5SP1.

work in Figure 5.

Synthesis of the minimum cost network generated to meet the task constraints of problem 7SP1 is given in Figure 7. To illustrate the importance of arriving at the proper sequence of subproblem isolations, should the series of stream matches for this problem have been (7/1 2/6, 7/4, 3/5, 2/5), the cost of the network would be nearly three times the minimum. The consequence of committing streams one and six to undesirable early matches is clear.

Problem 7SP2 displayed the special feature that all the available process heat is recoverable, and that total recovery is achievable by many configurations. Indeed, the synthesizer examined no less than thirteen distinct networks which would recover all the available heat. The difference in cost between the highest and lowest costing networks resulting in total recovery was on the order of 2%. The minimum cost network shown is in Figure 8. To a limited extent, this should bolster confidence in the resolving power of the heuristic technique used, although in no way is the possibility precluded that other selection rules may perform as well, if not better.

CONCLUDING REMARKS

This study has delineated a new problem area replete with significant avenues of investigation. There are two items of outstanding importance. First, there is the need for a systematic compilation of the logic employed in particular design areas. Second, there is a pressing demand for the further development of learning and/or synthesis search techniques. The authors believe that once there is some light shed on these topics, there can be real advances in the computerized synthesis of system designs.

Even this primitive attempt carries difficulties that were somewhat sidestepped. For example, interior optimization of the tear variables was approximated by invoking an additional rule of thumb rather than an exact objective. Likewise, the conclusion of a series of synthesis trials was hardly the result of a mechanized decision based on stopping criteria. Perhaps efforts along these lines will be premature until synthesis procedures become more solidly based on proved formalisms. The ground has hardly been broken in the field of mechanized process synthesis.

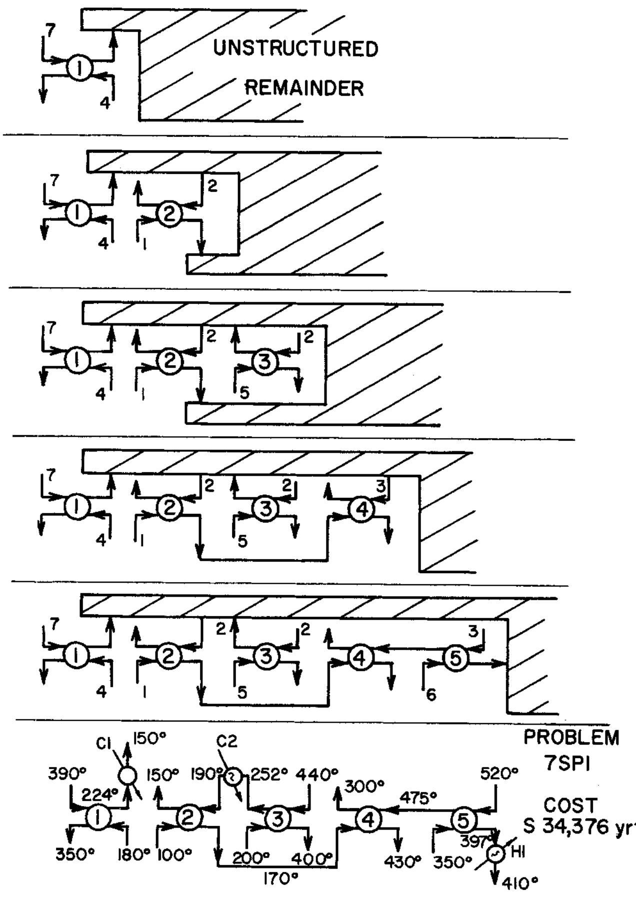

Fig. 7. Solution to problem 7SP1.

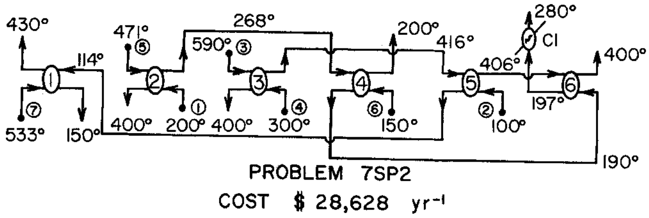

Fig. 8. Solution to problem 7SP2.

ACKNOWLEDGMENT

This research has been supported in part by a grant from the National Science Foundation.

NOTATION

A = heat exchanger area
$A \subset B$ = the set A is contained in the set B
$A \cup B$ = the union of sets A and B
$A \cap B$ = the intersection of sets A and B
a, b = heat exchanger cost parameters
α = plant yearly downtime
C = heat exchanger cost
c_i = heat capacity of stream i
C_w = cooling water cost
C_s = steam cost
$E(X_j)$ = the economics of existing technology for task X_j
l = decomposition level
l_s = steam latent heat
L = average number of decomposition levels for complete synthesis
M = equipment payout time
m = number of streams to be heated
n = number of streams to be cooled
$O^{(i)}(\mathbf{X})$ = an estimate of the optimal objective function achievable for task $\mathbf{X}$
$O^*(\mathbf{X})$ = the optimal objective function achievable for task $\mathbf{X}$
$O^{(i)}_{\text{act}}(\mathbf{X})$ = the actual objective function achieved at synthesis trial i
$O^{(i)*}_{\text{act}}(\mathbf{X})$ = the optimized actual objective function achieved at synthesis trial i
Opt $\{\cdot\}$ = optimize the function in the argument
p_s = steam pressure
$\mathbf{R}$ = the region of existing technology
r_l = the number of candidate matches at level l
s = total number of streams
$\mathbf{T}$ = the tear constraint set
t = stream temperature
τ = minimum allowable approach temperature difference
$\mathbf{X}$ = task constraint set
$\mathbf{X}_j$ = jth subset of the task constraints

LITERATURE CITED

1. Rudd, D. F., *AIChE J.*, 14, 343 (1968).

Manuscript received November 1, 1967; revision received December 26, 1967; paper accepted January 10, 1968.