NETWORK FLOW MODELS FOR HEAT EXCHANGER NETWORK SYNTHESIS: PART 2 - FINDING MINIMUM MATCH SOLUTIONS

by

Jaime Cerda and Arthur W. Westerberg

DRC-06-26-81

March 1981

NETWORK FLOW MODELS FOR HEAT EXCHANGER NETWORK SYNTHESIS:
PART 2 — FINDING MINIMUM MATCH SOLUTIONS

by

Jaime Cerda*

and

Arthur W. Westerberg

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

March 1, 1981

This work was funded by NSF Grant No. CPE-780189

* Current Address: INTEC
Guernes 3450
3000 Santa Fe
Argentina

ABSTRACT

Using the fact that the lowest cost heat exchanger network nearly always possesses the highest possible degree of energy recovery and comprises the least number of positive stream/stream matches, a new mathematical formulation for the network design synthesis problem is proposed. The new representation accounts for the problem thermodynamic constraints, and its solution space only includes networks with maximum energy recovery. Neither cyclic nor split networks are ignored as long as they are thermodynamically feasible.

The new model admits an efficient solution procedure and each of its solutions represents at least a low cost network design. A near-optimum solution of the model is found by solving a much simpler mathematical relaxation of it. Frequently, it already comprises a minimum number of positive matches. If not so, a modified version of the Stepping Stone Algorithm commonly used to solve linear transportation problems permits one to derive an optimal solution.

Another simple mathematical relaxation of the network synthesis problem formulation is proposed to help the searching for all the other feasible networks comprising a minimum number of matches. A very" well defined and efficient search technique is presented and applied to two example problems.

INTRODUCTION

A significant industrial design problem has become the synthesis of energy recovery networks aimed at reducing the overall thermal energy consumption in a processing plant. Such networks are commonly used to recycle thermal energy within a process, preventing its wasteful loss with effluent materials. This synthesis problem was formalized by Masso and Rudd (1969), and its goal is the development of a systematic procedure capable of discovering the heat exchanger network which reaches process specifications at minimum cost.

Cyclic networks, that is networks in which two streams are matched against each other more than once, are not excluded by the problem statement. Neither are networks that contain *paiaAAeA ^pAitting* of one or more of the streams. Very often, cyclic and/or split networks represent the only options to reach maximum energy recovery. For instance, when the heat capacity times the flow rate of a cold (or hot) stream is excessively large as compared to those of the hot (or cold) streams it is usually impossible to get maximum energy recovery in an unsplit acyclic network. In such cases parallel stream splitting in the network becomes absolutely necessary (Ponton and Donaldson, 1974). Unsplit acyclic networks are those which include neither multiple matches between the same pair of streams nor parallel stream splitting.

As pointed out by Motard and Westerberg (1978), one of the crucial obstacles to overcome in Process Synthesis is the *modeUAing* of the problem. Before searching for the optimal solution, we should have at hand a representation of the problem *tick enough,* to embed all feasible alternatives but still *^impAe. enough* to admit a solution technique capable of quickly discovering the better networks.

There were so far three attempts to develop an *ancUyticaU* representation of the network synthesis problem (Kesler and Parker, 1971; Kobayashi, Umeda and Ichikawa, 1971; Cena, Mustacchi and Natali, 1977). In their review, Nishida et al. (1981) called them simultaneous match decision algorithms. Those three attempts describe in different ways the network synthesis problem as an assignment problem, a very well known mathematical model in Operations Research for which quite efficient solution algorithms are already available.

Kobayashi et al. (1971) formulated the network synthesis problem as an assignment problem where each cold stream can be matched at most with *orvly one.* hot or utility stream and conversely each hot stream can be matched at most with a single cold or utility stream. A major model limitation was the fact that the problem thermodynamic constraints were not analytically considered. Cena et al. (1977) avoided such a difficulty by using an approach similar to that of Kesler and Parker (1969). Their basic idea consisted of partitioning each process stream into a number of equivalent pseudostreams, all of them having the same elementary heat duty equal to some small value Q. The assignment model was then applied to the new set of pseudostreams.

A new analytical representation of the heat exchanger network synthesis problem is introduced in this paper. Such a representation is not restricted to special cases but it can be used for any network synthesis problem. It embeds acyclic networks as well as cyclic networks or those involving stream splitting. Its computational solution procedure permits one to find a maximum energy recovery network which is not only thermodynamically feasible, but it also contains the lowest possible number of active stream/stream matches. If such an optimal solution of the proposed model stands for an unsplit acyclic heat exchanger network, drawing the network structure is straightforward. If not, additional steps must be taken before drawing the structure of a split and/or cyclic network. They are described in part 3 of this work.

Feasible Networks with Maximum Energy Recovery

Once the utility usage (UU) problem or problem PI (Cerda et al., 1981) has been solved, the minimum heating and cooling requirements are known. At this point, a slight modification of the set of constraints of the UU problem is proposed which consists of cutting the amount of thermal energy flow available at (or demanded by) the auxiliary heating (or cooling) source to precisely its minimum value. At the same time, obviously, matches between those auxiliary sources become forbidden. In this way, a new get of constraints is derived which defines the set of all feasible networks with the highest possible degree of energy recovery, (S_M). The costs of the networks belonging to (S_M) differ only because of their distinct investment costs.

Setting Performance Targets

If one really wants to find the minimum cost solution, the real cost of a heat exchanger network should in principle be used as the objective function of the problem. However, such an objective function makes the mathematical description of the set of constraints much more difficult because the process stream temperatures at intermediate points in the network will arise as additional variables. Furthermore, the set of constraints will no longer be linear.

In order to get rid of such a complexity, we are going to set two *performance targets* usually reached by the economic optimum heat exchanger network (Linnhoff et al., 1980). They can lead to a more convenient objective function for the problem. Because of the variables it includes, the new objective function could admit a much simpler mathematical model and, what is more important, is still minimized by the lowest cost solution in most cases.

From the examples found in the literature, one can make two important observations about the optimal and near-optimal designs. In each case, the optimal (or a near-optimal) design is one comprising a minimum collection of matches among hot and cold process streams, including utilities, where the maximum degree of energy recovery is achieved. These two features can be chosen as the performance targets to reach in the design of heat exchanger networks.

Both performance targets have solid justifications. Soaring energy costs have made the operating cost the dominant component of the network total cost. Therefore, an economic network design should always minimize the heating and cooling requirements (Hohmann, 1971; Rathore and Powers, 1975; Nishida, Liu and Lapidus, 1977; Grossmann and Sargent, 1978; Linnhoff and Flower, 1978a; Flower and Linnhoff, 1980). On the other hand, the capital outlay cost is controlled by the number of heat transfer units in the network design (Hohmann, 1971; Nishida, Liu and Lapidus, 1977; Linnhoff and Flower, 1978b; Flower and Linnhoff, 1980).

Since each match between process streams is implemented by at least a single heat exchanger, *the minimum numbe* o{L matches 4oJL* up a Aowet bound on the numbea ofL heat exchangei* {Lot any {Lea^ibXe netwoik design.* Moreover, when each match is accomplished by a single unit, as happens in an acyclic network, there is no difference between them, and we can refer to either a match or its correspondent heat, exchanger. That situation arises in the optimal network designs of most of the test problems found in the literature like 4SP1, 5SP1, 6SP1, 7SP1, 7SP2 and 10SP1, because they are acyclic networks. However, such a feature also characterizes the optimal solution to the problem 4SP2 where parallel stream splitting is absolutely necessary to have maximum energy recovery (Linnhoff and Flower, 1978b).

The Mathematical Model

For the rest of this work we shall refer to a cold (or hot) pseudostream, provided by the partitioning procedure proposed by Linnhoff and Flower (1978a) and somewhat modified by Cerda et al. (1981), through a pair of subscripts. The first of them will indicate the primitive process stream from which it was generated, while the second stands for the level of the temperature interval to which it belongs. Thus, $q_{ik,j\ell}$ stands for the heat flow shipped from source $h^{j\ell}$ (the original hot stream j at temperature level ℓ) to destination c^{ik} (the primitive cold stream i at level k). We shall also use $b^{j\ell}$ to indicate the energy flow available at $h^{j\ell}$ and a^{ik} to represent the energy flow demand at c^{ik}.

The seven-stream test problem which will later serve to illustrate the performance of the new synthesis procedure can be used to get a better understanding of the nomenclature just described. Its set of relevant data is shown in Table 1. Table 2 gives the four intervals for partitioning the process streams, while the corresponding set of cold and hot pseudo-streams, their supply and target temperatures and their energy flow capacities ($b^{j\ell}$) or demands (a^{ik}) are listed in Table 3.

A network design reaching the two performance targets discussed in the last section is an optimal solution to the following problem P2.

P2: $$\text{Minimize} \sum_{i=1}^{C} \sum_{j=1}^{H} y_{ij}$$

subject to

$$\sum_{i=1}^{C} \sum_{k=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} = b_{j\ell} \quad , \quad \text{for} \quad j=1,2,\cdots,H \quad \ell=1,\cdots,L \tag{2.1}$$

$$\sum_{j=1}^{H} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} = a_{ik} \quad , \quad \text{for} \quad i=1,2,\cdots,C \quad k=1,\cdots,L \tag{2.2}$$

$$q_{ik,j\ell} \geq 0 \quad , \quad \text{for} \quad i=1,2,\cdots,C \quad j=1,2,\cdots,H \quad k=1,\cdots,L \quad \ell=1,\cdots,L \tag{2.3}$$

where

$$b_H = (b_H)_m \quad ; \quad a_C = (a_C)_m$$

and

$$\sum_{k=B_i}^{T_i} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} \leq U_{ij}\, y_{ij}, \quad \text{for} \quad i=1,2,\cdots,C \quad j=1,2,\cdots,H \tag{2.4}$$

$$y_{ij} = 0,1 \quad , \quad \text{for} \quad i=1,2,\cdots,C \quad j=1,2,\cdots,H \tag{2.5}$$

where y_{ij} is an integer variable that can only take binary values (1,0), depending on whether or not the match (c_i,h_j) is accomplished in the network. The number of cold process streams is C-l and the number of hot process streams is H-l. A single auxiliary heating source H and a single auxiliary cooling source C are considered in the problem formulation. $(a_C)_m$ and $(b_H)_m$ are the minimum values found when solving problem UU mentioned earlier. Frequently, either $(a_C)_m$ or $(b_H)_m$ is equal to zero and consequently its corresponding constraint equation can be deleted decreasing the number of rows (or columns) by one. $m_{ik,j\ell}$ is a binary coefficient which is equal one when the match $(c_{ik},h_{j\ell})$ is thermodynamically feasible or otherwise is equal zero. The, problem is partitioned into L temperature intervals - see part 1 (Cerda, et al., 1981). Finally, U_{ij} is the upper bound on the amount of heat to assign to the match (c_i,h_j).

The upper bound U_{ij} is the maximum amount of heat that can be transferred from hot stream h_j to cold stream c_i. It can be determined by applying the northwest corner rule to a standard transportation problem tableau that only includes the hot pseudostreams which come from h_j as heat sources and the cold pseduostreams that come from c_i as heat sinks. Heat sources and sinks should be ordered in decreasing temperature levels. In many cases, $U_{ij} = \min(a_i,b_j)$ where b_j is the total energy flow available at h_j and a_i is the total energy flow to be supplied to c_i. In general, $U_{ij} \leq \min(a_i,b_j)$. To see that U_{ij} can be less than $\min(a_i,b_j)$, think of the extreme case where c_i is everywhere hotter than h_j. For this case $U_{ij} = 0$ since h_j is too cold to supply heat to c_i. When a certain match is forbidden, its corresponding U_{ij} is reduced to zero.

In order to understand constraints (2.4) and (2*5), one can make the **following** observations. If for a given feasible solution $y_{ij} = 0$ the match (c_i, h_j) does not take part in the solution. In that case, the linear constraint (2.4) for the match (c_i, h_j) reduces to

$$\sum_{k=1}^{L} \sum_{l=1}^{L} m_{ik,jl} \, q_{ik,jl} \leq 0$$

Since all $q_{ik,jl}$ are non-negative, the above expression becomes a strict equality which is satisfied when all $q_{ik,jl}$ for k=1,...,L and l=1,...,L are equal zero.

When $y_{ij} = 1$ the match (c_i, h_j) is active in the heat exchanger network under consideration. The inequality (2.4) for that match becomes

$$\sum_{k=1}^{L} \sum_{l=1}^{L} m_{ik,jl} \, q_{ik,jl} \leq U_{ij}$$

which is not significant because such a restriction is already accounted for through constraints (2.1) and (2.2).

As proposed here, the heat exchanger network synthesis problem is a special type of mixed integer linear programming problem (M1LP) whose general mathematical formulation is

(MILP) **Minimize** $(c_1 q + c_2 y)$

subject to

$$A_1 q + A_2 y = b$$

$$q \geq 0$$

$$y \geq 0 \;, \quad \text{integer}$$

In P2, the vector c_1 **and the matrix** A_2 **are both null.**

A Relaxation of the Network Synthesis Problem; The Linear Transportation Problem P3

Let's ignore the integer constraints on the variables y_{ij}. In doing so, we generate a new problem $\overline{P2}$ which is a relaxation of the network synthesis problem P2 in the sense that every feasible solution of P2 is also feasible for $\overline{P2}$ but the reverse may not be true. Then $f^*(\overline{P2}) \le f^*(P2)$, where the asterisk indicates the problem optimal value. In other words, the relaxation of P2, i.e. $\overline{P2}$, provides a lower bound for the value for P2.

As Balinski (1961) proved, $(q_{ik,j\ell}, \bar{y}_{ij})$ is an optimal solution to $\overline{P2}$ only if 2.4 is a strict equality, i.e.

$$\sum_{k=1}^{L} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} = U_{ij}\, \bar{y}_{ij} \quad , \quad \text{for } i=1,2,\cdots,C \quad j=1,2,\cdots,H$$

The proof is relatively simple. If $y_{ij} = 0$, we have already seen that all $q_{ik,j\ell}$ for $k=1,\ldots,L$ and $\ell=1,\ldots,L$ are equal zero and the above expression holds. On the other hand, if $y_{ij} > 0$ and

$$\sum_{k=1}^{L} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} < U_{ij}\, y_{ij}$$

then, as the integrality constraints on y_{ij} have been dropped, $\bar{y}_{ij}$ can be decreased without violating constraints (2.4). In this way, the value of the objective function of $\overline{P2}$ is also decreased. The optimal value is achieved when (2.4) hold as strict equalities. Balinski (1961) proposed this theorem for the widely known fixed-cost transportation problem of which $\overline{P2}$ is a particular case. Then

$$\bar{y}_{ij} = \frac{1}{U_{ij}} \sum_{k=1}^{L} \sum_{\ell=1}^{L} \qquad i=1,2,\cdots,C \quad j=1,2,\cdots,H$$

This expression for $\bar{y}_{ij}$ assures that its value will be equal zero when the match (c_i, h_j) is not implemented by the network. Replacing $\bar{y}_{ij}$ for this expression in the objective function, the mathematical program $\overline{P2}$ is transformed into a new one:

$$\text{Minimize} \quad \sum_{i=1}^{C} \sum_{j=1}^{H} \bar{y}_{ij} \sum_{k=1}^{L} \sum_{l=1}^{L} m_{ik,jl}\, c_{ik,jl}$$

subject to the set of constraints (2.1), (2.2) and (2.3) (Balinski, 1961).

By assigning very high unit costs M to infeasible and forbidden routes ($m_{ik,jl} = 0$) to prevent them from being in the optimal solution, the coefficients $m_{ik,jl}$ can be removed from the problem representation. Now $\overline{P2}$ can be written as the linear transportation problem P3.

$$\text{P3:} \qquad \text{Minimize} \quad \sum_{i=1}^{C} \sum_{j=1}^{H} \sum_{k=1}^{L} \sum_{l=1}^{L} c_{ik,jl}\, q_{ik,jl}$$

subject to

$$\sum_{i=1}^{C} \sum_{k=1}^{L} q_{ik,jl} \le x_{jl} \qquad j=1,\ldots,H; \quad l=1,\ldots,L$$

$$\sum_{j=1}^{H} \sum_{l=1}^{L} q_{ik,jl} \le v_{ik} \qquad \text{for } i=1,2,\ldots,C; \quad k=1,\ldots,L \qquad (3.2)$$

$$q_{ik,jl} \ge 0, \qquad \text{for } i=1,2,\ldots,C; \quad j=1,2,\ldots,H; \quad k=1,\ldots,L; \quad l=1,\ldots,L \qquad (3.3)$$

where

$$c_{ik,j\ell} = \begin{cases} M, & \text{if } i = C \text{ and } j = H \text{ or } k > \ell \\ 1/U_{ij}, & \text{otherwise} \end{cases}$$

In the formulation of P3 the variables $\bar{y}_{ij}$ no longer appear. Both problems, the network synthesis problem P2 and the linear transportation problem P3, have the same set of constraints on the variables $q_{ik,j\ell}$. Therefore, *aveau {LzaAible, Aoluition to 93 i* cU^o {Lo.aAibl& fLoi 92.* A vertex of the common convex constraint set on $q_{ik,j\ell}$ will be the optimal solution of P2. The same can be said of P3. However, the same vertex does not always simultaneously minimize both problems.

It should be emphasized that no feasible heat exchanger network has been excluded from the solution space defined by the set of constraints (3.1), (3.2) and (3.3). It does not matter whether the network is acyclic, cyclic or includes stream splitting. However, one can only guarantee that the optimal solution to P2 will comprise the minimum number of units if it stands for an acyclic network.

A Very Good Solution to the Network Synthesis Problem

Like the utility usage problem P3 is a transportation problem in linear programming that can also be solved using Dantzig's algorithm (Dantzig, 1963). Both problems PI and P3 share the same set of constraints on the variables $q_{ik,j\ell}$ (the only changes are the values of a_n and $b_{..}$, On and the infeasibility of the match (C,H)). Therefore, after finding the minimum utility usage, to solve P3 only requires changing the cost coefficients for the permissible routes (i_k, j_ℓ) and applying Dantzig's algorithm again. We should use $(1/U_{ij})$ as the new cost coefficients.

Tiplitz (1973) suggested that a very good solution to the network synthesis problem P2 can be derived from the optimal solution to P3 by taking

$$y_{ij} = 1 \quad \text{if} \quad \sum_{k=1}^{L} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} > 0$$

or

$$y_{ij} = 0 \quad , \quad \text{if} \quad \sum_{k=1}^{L} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} = 0$$

The reason is rather simple. As the cost coefficients in P3 are the reciprocals of the upper bounds U_{ij}, it follows that the optimal solution to P3 will try to preferentially allocate the amount of heat to exchange among process streams to those matches with higher upper bounds. At the optimal solution to P3 the average value of the upper bound for the set of positive or active matches ($y_{ij} > 0$) is a maximum. On the other hand, the total amount of heat to exchange among process streams including utilities is a fixed quantity if the highest degree of energy recovery is to be achieved. Then, a minimum collection of active matches means that the average amount of heat flow to exchange in each match has been maximized. Frequently, *the average amount o{L heat q_{ij} exchanged in the active matches i* a maximum when the avziage vaXue o£ the uppet bound U_{ij} {JOI the <*et o{L active matches i* aJL^o maximum.*

Synthesizing a Good Heat Exchanger Network

Based on the linear transportation problem P3 a computational procedure is proposed to synthesize a good heat exchanger network. It is essentially the solution method of a standard transportation problem which is applied to get a very good solution to the network synthesis problem P2. The algorithm consists of four steps whose goals will be clearly stated while solving the seven-stream test problem shown in Table 1.

Step 1: Find the minimum heating and cooling utility requirements to meet the specifications of the process.

Cerda et al. (1981) have described a method to carry out Step 1 even if some of the feasible matches would have been forbidden. It consists in solving the utility usage problem PI, which is a linear transportation problem, using Dantzig's algorithm. As stated by the authors, an appropriate ordering of rows and columns greatly increases the algorithm efficiency. For the seven-stream test problem, the optimal tableau is displayed in Table 4 where hot pseudostreams (except for the auxiliary heating source) have been arranged in increasing number of infeasible and forbidden matches while cold pseudostreams were ordered in decreasing number of such matches. The optimum was reached by simply applying the northwest corner rule (Dantzig, 1963).

Step 2: Compute the upper bounds U_{ij} for each match (c_i, h_j).

For the seven-stream test problem, the tableaux to determine U_{ij} for all pairs (c_i, h_j) have been grouped in Table 5. Note that (c_1, h_7), (c_4, h_5) and (c_4, h_7) matches have upper bounds less than $\min(a_i, b_j)$ for the match. $U_{1H} = 236$ for steam and any cold stream c_1 is limited to the minimum amount found bv solving problem PI. All the values U_{ij} and the corresponding cost coefficients c_{ij} are listed in Table 6.

Step 3: Invent a good initial feasible network design.

First, the auxiliary heating and cooling source capacities are reduced to the limiting values required by a maximum energy recovery network. If it remains in the tableau, the match (C,H) between the utilities is forbidden.

In order to improve the quality of the initial feasible network design, the ordering of the columns in the tableau will be somewhat modified. At this step, the auxiliary heat source column should be placed first in the tableau. When multiple heating sources are used, all the columns including the utility columns are to be arranged in increasing number of infeasible and forbidden matches. Similarly to PI, an initial feasible solution to P3 is obtained through the northwest corner rule (see Table 7). By merging the matches between the same pair (c_i, h_j) at different temperature levels one obtains the simpler tableau also shown in Table 7 from which an initial *tplit* network design can be derived (see Part 3).

Step 4: Find the optimal solution to P3 by using the solution method of a linear transportation problem.

In a linear transportation problem, there are usually at each iteration several variables $q_{ik,j\ell}$ whose introduction in the current solution will decrease the value of the objective function. All of them can be detected from applying the transportation problem algorithm. As is usually recommended, one should choose the one that produces the biggest improvement to speed up convergence to the optimal solution. However, if there are ties the following rules are given to break them; they should be used in the order they are listed:

(1) Place the variable $q_{ik,j\ell}$ into the solution that has the smallest

(2) Choose the variable $q_{ik,j\ell}$ located at the uppermost row.

(3) Pick up the variable $q_{ik,j\ell}$ at the rightmost column.

For the test problem, the optimal solution to P3 is shown in Table 8. The merging procedure just mentioned in Step 3 permits one to draw the unsplit acyclic network design depicted in Figure 1. As we shall discover, Table 8 is not an optimal solution to P2. We now present how to discover this fact and modify the solution to get an optimal solution to P2.

A New Relaxation of the Network Synthesis Problem

The "merged" table in the lower left corner of Figure 1 suggests a convenient relaxation of the network synthesis problem which provides a precise *lower bound* on the number of active matches for a given network design problem. Despite the fact we cannot guarantee that the optimum for this relaxed problem is a feasible solution to P2, it sometimes happens. In those cases, as its size will be much smaller than that of P2 the new relaxation implies a fast way to find a minimum match solution with maximum energy recovery by hand calculations, even for large problems.

When a subset of the linear constraints (2.1) or (2.2) is substituted for a single constraint that is the sum of them, a new mathematical problem is generated which is a relaxation of the original one. Such a single constraint, called a surrogate constraint, is weaker than the set of constraints it replaces. Therefore, each feasible solution to the original constraint set also satisfies the surrogate constraint but the reverse may not be true. By defining q_{ij} as the amount of heat exchanged in the match (c_i, h_j),

$$q_{ij} = \sum_{k=1}^{L} \sum_{\ell=1}^{L} m_{ik,j\ell}\, q_{ik,j\ell} \quad , \quad \text{for } \begin{array}{l} i=1,2,\cdots,C \\ j=1,2,\cdots,H \end{array}$$

a surrogate constraint can be generated from adding all the constraint equations (2.1) or (2.2) which stand for pseudostreams that come exclusively from the same h_j (or c_i) at different temperature intervals I (or k). In this way, a new relaxation of P2 formed by "merging" of constraints is obtained that is given by:

P2': Minimize $$\sum_{i=1}^{C} \sum_{j=1}^{H} y_{ij}$$

subject to

$$\sum_{i=1}^{C} q_{ij} = K_j \quad , \quad \text{for} \quad j=1,2,\cdots,H \qquad (2'.1)$$

$$\sum_{j=1}^{H} q_{ij} = a_i \quad , \quad \text{for} \quad i=1,2,\cdots,C \qquad (2'.2)$$

$$q_{ij} \geq 0 \quad , \quad \text{for} \quad i=1,2,\cdots,C; \; j=1,2,\cdots,H \qquad (2'.3)$$

$$q_{ij} \leq U_{ij}\, y_{ij} \quad , \quad \text{for} \quad i=1,2,\cdots,C; \; j=1,2,\cdots,H \qquad (2'.4)$$

$$y_{ij} = 0,1 \quad , \quad \text{for} \quad i=1,2,\cdots,C; \; j=1,2,\cdots,H \qquad (2'.5)$$

b_j is the amount of heat flow to remove from the hot stream h_j, while a_i is the quantity of heat flow to add to the cold stream c_i. It may happen that $U_{ij} < \min(a_i,b_j)$ for certain matches (c_i,h_j). We saw this for matches $(c_1,h_?)$, (c_4,h_5) and $(c_4,h_?)$ for our example 7 stream problem (see Table 6). Therefore, if they are active ($y_{ij}= 1$), the corresponding constraints (2'.4) are not already taken into account by (2'.1) and (2'.2) as for P2. Constraints (2'.1) and (2'.2) only bound a match to be less than $\min(a_i,b_j)$.

By ignoring the integer constraints on y_{ij}, we have a new "merged" problem $\overline{P2'}$ which reaches its minimum when constraints (2'.4) are strict equalities. That follows from Balinski's theorem because $\overline{P2'}$ is a particular type of fixed-charge transportation problem (Balinski, 1961). Therefore, its value can be found by solving,

$$\text{P4:} \qquad \text{Minimize} \sum_{i=1}^{C} \sum_{j=1}^{H} (1/U_{ij})\, q_{ij}$$

subject to the constraints (2'.1), (2'.2) and (2'.3) plus

$$q_{ij} \leq U_{ij} \quad , \quad \text{for all } (i,j) \in \Pi$$

where

$$\Pi = \{(i,j) \mid U_{ij} < \min(a_i,b_j)\}$$

If we want both problems $\overline{P2'}$ and P4 to have the same set of constraints on q_{ij}, the upper-bound type constraints for those matches (c_i,h_j) with $U_{ij} < \min(a_i,b_j)$ should be explicitly included in the formulation of P4.

For our seven stream problem the set II equals {(1,7), (5,4), (4,7)} implying the constraints

$$q_{17} \pm 940 < \min(946,1198)$$

$$q_{54} \wedge 707 < \min(1100,1256)$$

$$q_{47} \wedge SSO \wedge \min(1100,1198)$$

are needed in addition to the usual transportation problem constraints (2'.1), (2'.2) and (2'.3).

We remind the readers here of the problems defined so far to help reduce confusion in what follows•

Partitioned problems

P1: Minimum utility problem, with streams partitioned.

P2:' Minimum match/minimum utility problem, with streams partitioned.

$\overline{P2}$: P2 but with integer constraints ignored on y_{ij}.

P3: ¥2 with variables y_{ij} substituted out (alia Balinski).

Merged problems

$P2^1$: P2 but with merged surrogate constraints — i.e. streams not partitioned. Leads to much reduced problem size.

$\overline{P2^1}$: $P2^1$ but with integer constraints ignored on y_{ij}.

P4: $\overline{P2^1}$ with variables y_{ij} substituted out.

A Lower Bound on the Number of Positive Matches

P4 is a special type of transportation problem with upper-bound secondary constraints. Being a linear programming problem, every vertex of the convex constraint set of P4 is a basic solution with $((C-1)+(H-1)+N_U-1-d+s)$ positive q_{ij}, where s is the number of upper-bound type constraints that hold at the vertex, d is the number of basic variables q_{ij} which are equal zero and N_U is the number of utilities (steam and/or cooling water) required by a maximum energy recovery network (Llewellyn, 1964). For non-degenerate basic solutions, d=0. In other words, $((C-1)+(H-1)+N_U-1-d+s)$ is the number of linearly independent equality constraints at each vertex. For this kind of transportation problem, it need not be true that the basic cells at any basic solution form a tree in the standard transportation problem tableau. It is quite possible that a loop appears in the optimal solution with one of its basic cells or matches holding at its upper bound. The tableau in the lower left corner of Figure 1 contains a loop: (c_2,h_5), (c_2,h_7), (c_1,h_7), (c_1,h_5) and back to (c_2,h_5).

As the merged problem P2' is a particular version of the fixed-cost transportation problem, it immediately follows from using Balinski's theorem that an optimal solution to P2' is always an extreme point of the constraint set of P4. Moreover, it is a particular extreme point where the **number** of secondary upper-bound constraints that holds is as small as **possible.** In the limiting case, none of such upper-bound constraints would **hold** as equalities, and the optimal solution to P2' would only have $((C-1)+(H-1)+N_U-1-d)$ positive matches q_{ij}.

Let us now consider the fully partitioned network synthesis problem P2 again. Since P2' is a relaxation of P2, the lower bound on the objective function of P2', which is equal to $((C-1)+(H-1)+N_U-1-d+s)$, also is a lower bound for P2. Therefore, in the absence of degeneracy, the lower bound on the number of matches coincides with the lower bound on the number of heat transfer units proposed by Hohmann (1971), only if s=0.

Linnhoff and Flower (1978,a,b) reported two instances where even stream splitting solutions had to consist of a number of units larger than Hohmann's lower bound by one. One can interpret this to mean that d=0 and s=l at the optimal vertices of both problems. In their paper, Linnhoff and Flower (1978b) also remarked that "it is not impossible that heat loads of a hot and a cold stream are equal to each other (or to the total load on coolers or heaters), or residuals turn out to be equal to each other or to original loads. In any one of these situations, the minimum number of units is less than that suggested by Hohmann's rule.[11] In other words, the optimal solution can be a degenerate solution to P4, and that may happen if a partial sum of a_i is equal to a partial sum of b_j. It should be pointed out that the degree of degeneracy of the optimal solution can be greater than one. Grossmann and Sargent (1978) reported an optimal solution for Problem 20SP1 which comprises one unit less than Hohmann's lower bound.

Tree-Type Solutions to the Network Synthesis Problem P2

When the value of the network synthesis problem P2 is equal to $((C-1)+(H-1)+N_U-1-d)$, the corresponding feasible solution of P4 is a basic solution or extreme point with s=0. In other words, it forms a tree and no loops can be observed in the q_{ij}-tableau. If $d > 0$, one or more basic cells in the tree are equal to zero. Such solutions will be called tree-type solutions to the network synthesis problem P2. If $s > 0$, the basic solution to the merged problem P4 derived from an optimal soluti:" » to the fully partitioned problem P2 will contain a loop where one of the oasic cells holds at its upper bound $U^*_{ij} < \min(a_i, b_j)$. Therefore:

a) If the optimal solution to the fully partitioned, problem P3 has at most $((C-1)+(H-1)+N_U-1)$ positive q_{ij} then an optimal solution to the network synthesis problem P2 may be found. Unfortunately, if the optimal solution to P2 is a degenerate solution to P4 the only way to guarantee the optimality of the solution is by finding and comparing all of the tree-type solutions to P2.

b). If the optimal solution to P3 includes a loop when it is displayed on the q_{ij}-tableau (as happened here - see Table 8 and Figure 1), either of these Lhree cases occurs:

1. No optimal solution to P2 is an optimal solution to P3. In other words, none of them minimizes the objective function of P3. From that, it follows that a tree-type solution to the network synthesis problem P2 can be obtained by breaking the loop(s) in such a feasible way that the objective function of P3 *incaea^e^*. For any feasible solution to P3, the objective functions of both linear transportation problem P3 and the merged problem P4 take on the same values. This means that the loop(s) should be broken in such a feasible manner that the value of the P4-objective function becomes larger. Ordinarily, at most one or two loops are to be broken.

2. An optimal solution to P2 is an alternate optimal solution to P3. In this case, the loop is to be broken in a feasible way but keeping the value of the objective function of P3 or P4 unchanged.

3. An optimal solution to P2 is the optimal solution to P3 that has been found. It contains a loop when displayed on the merged q_{ij}-tableau, but there is no feasible way to break it. One of the positive matches on the loop holds at its upper bound U_{ij} $< \min(a_i, b_j)$.

Searching for an Optimal Solution to the Network Synthesis Problem: The Reverse Stepping Stone Method

The Stepping Stone Algorithm (see part 1 of this series of papers, Cerda et al., 1981) is a method to solve a standard transportation problem like P3. It involves the forming and breaking of loops in the solution tableau. The approach to finding an optimal solution is first to establish an initial feasible solution to the problem where the active (basis) cells form no loops. The Northwest algorithm will find such a solution.

Any inactive (nonbasis) cell which can complete a loop in the tableau for the current solution is a candidate cell to bring into the solution. To make this cell active requires increasing its value for $q_{IK,JK}$. To maintain the total heat delivered to the row (which is fixed to be a_{IK}) the active cell which is in the row and also in the loop with this previously inactive cell, must have its value for q reduced by the same amount. Similarly the q for the active cell in the same column must be reduced by this amount to maintain the fixed column total required by the problem.

Continuing around the loop the cells must be alternatively increased ("getter" cells) or decreased ("giver" cells) to maintain all row and column totals. A loop contains an even number of cells so getter and giver cells are the same regardless of the direction one moves around the loop. The "q" for the previously inactive cell can grow until "q" for one of the giver cells is reduced to zero. The newly formed zero valued giver cell breaks the loop leaving an altered feasible solution which again contains no loops.

Fortunately an easy calculation exists to assess whether introducing an inactive cell will increase, decrease or leave unchanged the objective function for the problem.

Given a feasible solution containing no loops, one finds an inactive cell whose introduction will reduce the objective function. A loop of active cells is then found which passes through this cell. The cell is grown until a "giver" cell in the loop becomes zero. This new tableau becomes the current feasible solution. The process is repeated until no cell can be found which reduces the objective function. The process terminates at the optimal solution (Rothenberg, 1979).

We are now going to define the Reverse Stepping Stone Method (RSSM) to turn solutions to P4 which contain loops into tree-type solutions, if possible. To break a loop one of its cells (i,j) must be deleted. Such a cell (i,j) must be a giver cell. We define the "value" of a loop associated with a cell as the sum of costs of the getter cells minus the sum of the costs of the giver cells. Then if the value of the loop is zero for a certain cell (i,j), it is also equal zero for the rest of the cells in the loop. If that happens, we are facing Case (b.2) where an optimal solution to P2 is an alternate optimal solution to P3. On the other hand, if the value of the loop is positive for a certain cell, then its exclusion from the solution will lead to an increase in the value of P3-objective function.

Given an optimal solution to P3, which contains a loop in P4, the Reverse Stepping Stone Method proceeds as follows:

1« Determine whether or not a positive q_{ij} in a loop should stay in the solution. In other words, we should verify if the removal of q_{ij} is a feasible modification. Such a test is done on the small-size relaxation problem P4 where each cell corresponds to a match between process streams or utilities. One knows that every q_{ij}-solution which is an infeasible solution for problem P4 is also infeasible for the network synthesis problem P2. Therefore, if a positive q_{ij} on a loop cannot be blanked without violating the non-negativity or upper-bound constraints of P4 for the other matches in the loop, such a cell must stay in the solution. When all of the matches in a loop are to stay, one of its positive matches q_{ij} holds at its upper bound $U_{ij} < \min(a_i, b_j)$ and it is very likely that any optimal solution to P2 will include a loop. Otherwise, we should go on to the next step.

2. The value of the loop is computed for an eligible positive match in the loop whose exclusion still leads to a feasible solution to P4. If it is equal zero, it is also zero for the other eligible matches in the loop. If it is different from zero, it will also take on the same value for the other cells in the loop which act as giver cells. For the getter cells, the value of the loop is also the same but with opposite sign.

   If the P3-optimal solution only contains a single loop whose value is different from zero, a tree-type solution to P2 could be obtained by removing an eligible match for which the value of the loop is *positive*. Usually, there is *only one* eligible match in the loop for which the value of the loop is positive. The same conclusion is still valid if the values of the other loops are equal zero.

   If the P3-optimal solution includes loops with zero values, it is recommended to remove from each loop the lowest cost eligible match. Usually, there is only one for each loop. This rule comes from the fact that the optimal solution to P3 is obtained by choosing always the lowest cost cell if two or more cells not in the basis would produce the same improvement in the objective function by introducing any of them into the solution.

If the P3-optimal solution contains more than a single non-zero loop, to get a tree-type solution to the network synthesis problem P2, an eligible match is to be removed from each non-zero loop in such a feasible way that the sum of their loop values is greater than zero. Non-negativity and upper-bound constraints for P4 serve to identify infeasible moves easily. Although it is not a frequent case, the number of different ways to break the loops is still very low.

3. After enumerating the distinct ways the optimal solution to P3 can be transformed into a tree-type solution to the network synthesis problem, the solution of a modified problem P3 will indicate whether or not it takes the current solution away from the feasible region of P2. In order to avoid new matches which are not part of the above set coming into the solution, a very high cost coefficient is assigned to each of them in the new P3. Furthermore, the match whose exclusion is attempted is also priced very high, although less than the previous ones. Now, the transportation problem algorithm is applied to the new P3 with the current optimal solution acting as the initial solution. If the new optimum comprises the same set of positive matches, the procedure has failed to produce a tree-type solution to P2 and another alternative should be tried. It is clear that we can sometimes derive from the optimal solution to P3 more than a single tree-type solution to the network synthesis problem.

Synthesizing an Optimal Solution to the Network Synthesis Problem P2

The implementation of the Reverse Stepping.Stone Method introduces a new step in the synthesis algorithm already proposed:

Step 5: Verify whether the optimal solution to P3 includes cycles when represented in the q..-tableau. If so, apply the Reverse Stepping Stone Method (RSSM) Vo get a tree-type solution to the network design synthesis problem P2.

Figure 1 shows us that the optimal solution to P3 for our seven stream example problem contains a loop involving streams c_2, c<* h_5 and h_7. The P3 and P4 problem costs for these active cells are found in Table 6: (c_2,h^) 7.96(xlO~^{4}), (c_2,h_7> 8.35, (c^hj) 10.64 and (c^*^) 10.57. Alternate cells around the loop are "givers" and "getters"; we are looking for ones which the givers will *i/icyie.a^e.* the P3 objective. If (c_2,h_5) is a "getter" the loop value is 7.96 - 8.35 + 10.64 - 10.57 = -0.32 and if (c_9,h_5) is a "giver" the value is +0.32. Thus we need only consider the latter where cells (c_2,h_5) and (c_1,h_7) are givers, with (c_1,h_7) the one which will go to zero first. Thus (c_1,h_7) is our only candidate.

We price all inactive cells in problem P3 very high, we price (c_1,11-) not so high but high and resolve P3. The prices should preclude any inactive cell from entering and should force (c_1,h_7) out if feasible but leave it in if not.

Solving we find (c_1,h_7) is successfully removed, generating the minimum match solution illustrated in Figure 2.

Note we have improved the objective function for P2 but had to increase the objective function for P3 to do it. We used the reduced problem P4 to screen out the alternatives we needed to consider.

Solving Related problems

—Frequently, control and safety constraints or other reasons rule out certain matches between process streams. They should be kept out of the network design by assigning to them very high costs. If additional constraints are added to the statement of the seven-stream problem which indicate that the matches (c_2,h_3) and (c_6,h_7) are forbidden, i.e. $U(c_2,h_3) = U(c_6,h_7) = 0$, the optimal solution to the new problem P3 stands for the tree-type network design depicted in Figure 3. In this case, no loops exist and thus the use of RSSM is not required. The formulation of problem P2 can easily handle this kind of constrained network design problem.

The feasible region of problem P2 can be enlarged to include network designs whose utility consumption is slightly higher than $(a_C)_m$ or $(b_H)_m$. One should change the constraint (2.2) and (2.3) for the utilities in the following way:

$$\sum_{i=1}^{C} \sum_{k=1}^{L} m_{ik,H}\, q_{ik,H} \le (b_H)_m + \Delta b_H$$

and

$$\sum_{j=1}^{H} \sum_{\ell=1}^{L} m_{C,j\ell}\, q_{C,j\ell} \le (a_C)_m + \Delta a_C$$

whose Δb_H and Δa_C are arbitrarily selected small values, Searching for All Tree-Type Solutions to the Network Synthesis Problem P2

By solving P3 and subsequently applying the Reverse Stepping Stone Method, if necessary, one can usually find a single tree-type solution to the network synthesis problem P2. However, we are interested in *aXX* of the solutions to P2 that comprise at most the same number of active matches as. the current best solution; i.e. all basic solutions to P4 which are also feasible for P2 or P3. Thus, one not only synthesizes all low-cost heat exchanger networks for a given problem but can find the minimum cost network satisfying our assumptions as well.

In Figure 4, tableau 1 portrays the tree-type solution already found for the seven-stream test problem through P3 and RSSM. We can generate a new basic solution to P4 by bringing a non-basic match into the solution. There are several alternatives: $(c_4,h_5>$, $(c_4,h_7>$, (c_2,S) and so on. It may happen, however, that the new basis is infeasible even for P4; i.e. it violates its secondary upper bound constraints. If feasible for P4, the new basis could require an additional amount of utilities.

We consider introducing an inactive cell. First we check problem P4. The cell is grown in P4 until a giver cell is brought to zero in the loop formed by introducing the inactive cell. If this change cannot be made **without** violating one of the secondary upper bound constraints on P4, we **reject** introducing this cell.

Next we check to see if introducing the new cell and deleting the corresponding giver cell leads to an increase in the utilities required. We do this step by solving problem PI after assigning very high costs to (i) **the** active match to drive it from the solution, (ii) all the inactive matches (c_i, h_j) except the one we are trying to introduce. That is, if match (c_i,h_j) in P4 is inactive, then all matches (c_{ik},h_{jt}) in PI are given high costs. To get a much better initial solution through the Northwest corner rule, rows and columns in the Pl-tableau are to be ordered as recommended for constrained heat recovery problems (Cerda et al., 1981). Except for those involving utilities, highly priced matches are ignored during the implementation of such a rule. Solving PI provides a new tree-type solution for P2 and its minimum utility requirements in a single step. Further studies on that solution are unnecessary if an increase in the utility consumption is observed. Otherwise, the algorithm to be proposed in Part 3 is applied to derive the network design structure.

To illustrate more fully the ideas, we consider Problem 5SP1.

The relevant data (Lee et al., 1970), are shown in Table 9. A complete description of the searching procedure to find all of its tree-type solutions shown in Figure 5 is detailed in Tables 10 and 11. Aimed at saving computing time the procedure includes an initial test which verifies if any of the matches in the current basis (node 1) must stay in it to keep the utility requirements at its minimum level. As indicated in Table 10 putting (c_1,h_4) out of the starting basis makes the utility consumption higher.

At the starting basis, node 1, there are four non-basic cells which **are** candidates to enter the basis (see Table 11). The initial test (not **having** (c_1,h_4) must increase utilities) rejects one of them, i.e. alternative (A.4). Another candidate (c^,h^) is discarded because bringing that **cell** into the basis would violate one of P4-upper bound secondary constraints. When each remaining, candidate (c_3,H) or (c_1,H) enters the starting basis, a new one is generated by removing the smallest "giver[11] **cell** in the cycle, i.e. (c^,H). Both new bases represent maximum energy recovery network designs that are identified, as nodes 2 and 3 in the search tree depicted in Figure 5. After using tools given in Part 3 of this paper, we will discover that node 2 stands for a network design which requires the splitting of streams c_1 and h^ and will require six heat exchangers. We will also discover that, for therraodynamic reasons, it will be necessary to accomplish match (c_1, h_4) in two units.

As said before, putting the nonbasis cell (c_3, h_2) or (c^h_4) in the basis is not studied. Sometimes, however, a new maximum energy recovery basis is generated if both nonbasis cells enter the current basis in a sequence. Such a solution would have never been found by the usual procedure of entering a single nonbasis cell one at a time if by themselves each would be rejected. This possibility may occur when one of the candidates removes a basis match (c_i, h_j) whose upper bound U_{ij} is exceeded when one attempts to include the other candidate. We see this for alternatives (A.4) and (A.3) where either removes the match (c_1, h_4) or exceeds the upper bound constraint on that match. By doing (A.3) first, then (c_5, h_4) enters the basis by replacing a match other than (c_1, h_4), and the upper-bound secondary constraint for (c_1, h_4) ceases to be violated. In (A.5) the match (c_5, h_2) is excluded instead of (c_1, h_4) and $q(c_1, h_4)$ does **not** exceed lUc^n_4). See Table 11. The new maximum energy recovery basis is called 4 in Figure 5,

Grimes (1980) proves that all solutions having positive heat loads **for** each match are connected and may be discovered by the search procedure being proposed here. The matches may not, however, be thermodynamically feasible——which relates precisely to our discovery that both matches (c_3, h_2) and (c_5, h_4) get rejected but the two together do not. A <u>complete</u> **search** must not reject a match because its precursor was rejected in the search tree.

In table 11 several ways have been used to reduce the search for all maximum energy recovery bases. We just mentioned that an alternative can be ignored due to either of the following facts:

(i) Violation of any of the upper-bound secondary constraints for Problem P4.

(ii) Increase in the minimum utility consumption when a * basis match is not included in all starting bases. Such a verification has been called the Initial Test.

(iii) Generation of a non-maximum energy recovery basis.

Other procedures were also employed in Table 11:

(iv) Avoid considering alternatives that would again put in matches which were previously removed from the basis somewhere along the branch connecting the root node and the current node in the search tree. This avoids generating the same node more than once.

(v) Avoid considering alternatives that would remove matches which have been brought into the basis somewhere along the branch connecting the root node and the current node in the searching tree. This avoids moving backward along the branch and producing nodes which belong to other branch in the tree.

(vi) In spite of (iv) and (v) a node could be generated more than once and this node is then ignored.

The set of maximum energy recovery bases found by the search are **shown** in Figure 5. Although there are six, only five of them were reported **by Flower** and Linnhoff (1980).

Now, we apply the search technique to finish identifying all of maximum energy recovery bases for a constrained version of the seven-**stream** test problem. For control or safety constraints we assume that the **matches** (c^'*1?) and (c1>h7) must stay out of the network. Moreover, we **require** that (c^,h^) must be included in the heat exchanger network. These requirements significantly decrease the number of alternatives which result. Returning to Figure 4, we find the search tree for this *constrained* seven-stream problem. It includes 20 nodes. Using techniques from Part 3 we find that 17 stand for unsplit networks comprising a minimum number of units. Their total fixed costs are listed in Table 12. They were evaluated based upon the design data used by Masso and Rudd (1969). The lowest cost network, i.e. node 15, which is perhaps the best answer for the unconstrained seven-stream problem, is depicted in Figure 6. Clearly this is not the only useful information provided by Table 12. Three other network designs have fixed costs higher than the lowest one by at most 0.56%, an insignificantly small difference since costs are only estimated approximately. Node 17 represents an interesting design because it could use a cheaper auxiliary heating source. Its temperature level must be only 148^{o} or higher (see Figure 7).

CONCLUSIONS

1. A well-known mathematical model in Operations Research is proposed to describe the heat exchanger network synthesis problem. Each of its solutions represents at least a near-optimum network design* This goal is achieved by analytically forcing the problem feasible region to include only networks with maximum energy recovery and by selecting the number of active matches as the objective function.

2. A solution procedure is presented which provides an optimal solution to the mathematical model. Its implementation requires a relatively small storage size and execution time even for large problems.

3. If the lowest cost network is an acyclic structure then it is an optimal solution to the proposed model.

4. Since the problem solution space includes cyclic and/or split networks, this method will yield an answer even if such types of structures are needed to achieve maximum energy recovery.

5. If the network discovered is cyclic or includes stream splitting, then an additional procedure to be introduced in Part 3 should be applied to derive the network structure.

6. The new synthesis method has been applied to a seven-stream test problem subject to several additional constraints. It yielded a maximum energy recovery network whose total fixed cost is only 6.8% higher than the optimum.

7. A simple mathematical relaxation of the network synthesis problem formulation is defined to help in the searching for all the other optimal solutions to the synthesis model, even if they stand for cyclic and/or split networks.

8. A well-defined search technique is presented to discover all of the optimal solutions to the synthesis model and applied to problem 5SP1. Only 15 alternatives should be considered before finding the six optimal solutions. The method was also used to solve a constrained seven-stream test problem yielding 17 unsplit network designs which contain a minimum number of units.

REFERENCES

Balinski, M.L., "Fixed-Cost Transportation Problem," Naval Research Logistics Quarterly 8(1), 41 (1961).

Cena, V., C. Mustacchi and F. Natali, "Synthesis of Heat Exchanger Networks: A Non-Iterative Approach," Chem. Eng. Sci. 3£, 1227 (1977).

Cerda, J., A.W. Westerberg, D.R.M. Mason and B. Linnhoff, "Predicting Minimum Utility Requirements in Constrained Heat Recovery Problems," to be published.

Dantzig, G.B. "Linear Programming and Extensions," Princeton University Press (1963).

Flower, J.R. and B. Linnhoff, AIChE J. 26, 1 (1980).

Grimes, L.E., "The Synthesis and Evolution of Networks of Heat Exchange," M.S. Thesis, Carnegie-Mellon Univ., Pittsburgh, PA (1980).

Grossmann, I.E. and R.W.H. Sargent, "Optimum Design of Heat Exchanger Networks," Computer and Chemical Engineering 2, 1 (1978).

Kesler, M.G. and R.O. Parker, "Optimal Networks of Heat Exchanger," Chem. Eng. Progr. Sym. Series No. 92, 61^, 111 (1969).

Kobayashi, S., T. Umeda and A. Ichikawa, "Synthesis of Optimal Heat Exchange Systems——An Approach by the Optimal Assignment in Linear Programming," Chem. Eng. Sci. 26, 3167 (1971).

Hohmann, E.C., "Optimum Networks for Heat Exchange," Ph.D. Thesis, University of S. California (1971).

Linnhoff, B. and J.R. Flower, "Synthesis of Heat Exchanger Networks. Part I——Systematic Generation of Energy Optimal Networks," AIChE J. 24, 633 (1978a).

Linnhoff, B. and J.R. Flower, "Synthesis of Heat Exchanger Networks. Part II——Evolutionary Generation of Networks with Various Criteria of Optimality," AIChE J. 24, 642 (1978b).

Llewellyn, R.W., "Linear Programming," Holt, Rinehart and Winston Inc., (1964).

Masso, A.H. and D.F. Rudd, "The Synthesis of System Designs. II. Heuristic Structuring," AIChE J. 15, 10 (1969).

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

Nishida, N., S. Kobayashi and A. Ichikawa, "Optimal Synthesis of Heat Exchanger Systems——Necessary Conditions for Minimum Heat Transfer Area and Their Applications to System Synthesis," Chem. Eng. Sci. 27, 1408 (1971).

Nishida, N., Y.A. Liu and L. Lapidus, "Studies in Chemical Process Design and Synthesis — Part III. A Simple and Practical Approach to the Optimal Synthesis of Heat Exchanger Networks," AIChE J. 23, 77 (1977).

Nishida, N., G. Stephanopoulos and A.W. Westerberg, "A Review of Process Synthesis," submitted for publication (1981).

Ponton, J.M. and R.A.B. Donaldson, "A Fast Method for the Synthesis of Optimal Heat Exchanger Networks," Chem. Eng. Sci. 2£, 2375 (1974).

Rathore, R.N.S. and G.J. Powers, "A Forward Branching Scheme for the Synthesis of Energy Recovery Systems," Ind. Eng. Chem. Process Design Development 14, 175 (1975).

Rothenberg, R.I., "Linear Programming," Elsevier North Holland (1979).

Tiplitz, Ch.I., "Convergence of the Bounded Fixed Charge Programming Problem," Naval Research Logistics Quarterly 20(2), 367 (1973).

NOTATION

a_i thermal energy flow required by the cold stream i, kw.

a_{ik} thermal energy flow at temperature level k required by the cold stream i, kw.

b_j thermal energy flow to be removed from hot stream j, kw.

$b_{j\ell}$ thermal energy flow at temperature level ℓ to be removed from hot stream j, kw.

B^i bottom temperature level of the cold stream i, dimensionless.

B^j bottom temperature level of the hot stream j, dimensionless.

c^i primitive cold process stream i, dimensionless.

c^{ik} primitive cold process stream i at temperature level k, dimensionless.

$c_{ik,j\ell}$ - cost of shipping a single kw from heat source $h^{j\ell}$ to heat sink c^{ik}, ¢ kw^{-1}.

C cold utility stream index; also cold utility stream.

C-1 number of cold process streams, dimensionless.

F_i heat flow capacity of the process stream i, kw/°C.

h^j primitive hot process stream j, dimensionless.

$h^{j\ell}$ primitive hot process stream j at temperature level ℓ, dimensionless.

H hot utility stream index; also hot utility stream.

H-1 number of hot process streams, dimensionless.

$m_{ik,j\ell}$ binary coefficient which indicates whether the match $(h^{j\ell}, c^{ik})$ is thermodynamically feasible, dimensionless.

q^{ij} thermal energy flow exchanged in the match (c^i, h^j), kw.

$q^{ik,j\ell}$ thermal energy flow exchanged in the match $(c^{ik}, h^{j\ell})$, kw.

T^i top temperature level of the cold stream i, dimensionless.

T^j top temperature level of the hot stream j, dimensionless.

U^{ij} upper bound on the thermal energy flow exchanged in the match (c^i, h^j), kw.

y^{ij} binary variable which indicates whether the match (c^i, h^j) is accomplished in the heat exchanger network, dimensionless.

LIST OF TABLES

TABLE 4

		1892	707	550	125	466	390	83	1888	70	10000
		h_{34}	h_{54}	h_{74}	h_{33}	h_{53}	h_{73}	h_{52}	h_{72}	h_{71}	H
1100	c_{14}	1100			I	I	I	I	I	I	
608	c_{24}	608			I	I	I	I	I	I	
473	c_{14}	184	289		I	I	I	I	I	I	
473	c_{23}		418	55				I	I	I	
473	c_{13}			473				I	I	I	
697	c_{63}			22	125	466	84	I	I	I	
228	c_{22}						228			I	
419	c_{62}						78	83	188	I	70
236	c_{21}									70	166
9764	C										9764

236kw

Q_s = 10,000 - 9764 - 236kw

Q_w = 0kw

TABLE 5

		1892	125	707	466	83	550	390	188	70
		h_{34}	h_{33}	h_{54}	h_{53}	h_{52}	h_{74}	h_{73}	h_{72}	h_{71}
1100	c_{44}	1100	I	707	I	I	550	I	I	I
608	c_{24}	608	I	608	I	I	550	I	I	I
473	c_{23}	473		99	374	I		390	I	I
228	c_{22}	228			92	83			188	I
236	c_{21}	236								70
473	c_{14}	473	I	473	I	I	473	I	I	I
473	c_{13}	473		234	239	I	77	390	I	I
697	c_{63}	697		697		I	550	147	I	I
419	c_{62}	419		10	409			243	176	I

TABLE 6

Match	U_{ij} m	$c_{ij} \times 10^{+4}$
(c_2, h_3)	1545	6.47
(c_2, h_5)	1256	7.96
$(c_2 > h_?)$	1198	8.35
(c_6, h_3)	1116	8.96
(c_6, h_5)	1116	8.96
$(c_6, h_?)$	1116	8.96
(c_4, h_3)	1100	9.09
$(c_r h_3)$	946	10.57
(c_l, h_5)	946	10.57
$(c_r h_7)$	940*	10.64
(c_4, h_5)	707*	14.14
$(C_4$, ty	550*	18.18
[illegible]	236*	42.37

*$U_{ij} < \min(a_i, b_j)$

≠U_{ij} = ialnimum utility bound on steam

TABLE 7

		236	1892	707	550	125	466	390	83	188	70
		H	h_{34}	"54	h_{74}	h_{33}	h_{53}	h_{73}	h_{52}	h?2	h_{71}
1100	c_{44}	236	864			I	I	I	I	I	I
.608	c_{24}		608			I	I	I	I	I	I
473	c_{14}		420	53		I	I	I	I	I	I
473	c_{23}			473					I	I	I
473	c_{13}			181	292				I	I	I
697	c_{63}				258	125	314		I	I	I
228	c_{22}						152	76			I
419	c_{62}							314	"83	22	
236	c_{21}									166	70

		236	2017	1256	1198
		H	h_3	h_5	h_7
1100	c_4	236	864		
1545	c_2		608	625	312
946	c_1		420	234	292
1116	c_6		125	397	594

TABLE 8

		236	1892	707	550	125	466	390	83	188	70
		H	h_{34}	h_{54}	h_{74}	h_{33}	h_{53}	h_{73}	h_{52}	h_{72}	h_{71}
1100	c_{44}	236	864			I	I	I	I	I	I
608	c_{24}		608			I	I	I	I	I	I
473	c_{14}			473		I	I	I	I	I	I
473	c_{23}		420			53			I	I	I
473	c_{13}			234	12		227		I	I	I
697	c_{63}				538			159	I	I	I
228	c_{22}					72	156				I
419	c_{62}							231		188	I
236	c_{21}						83		83		70

TABLE 9

Streams	F_i(kw/°C)	T_{in}(°C)	T_{out}(°C)	Q_i(kw)
c_1	11.40	38	205	1904
h_2	16.62	249	121	-2127
c_3	12.92	65	182	1512
h_4	13.29	205	66	-1847
c_5	13.03	94	205	1446
				ΣQ_i = 888

TABLE 10

Match Out	Minimum Utility Usage (kw)
(c_1, h_4)	1148 (> 888)
(c_5, S)	888
(c_5, h_2)	888
$(c_r h_2)$	888
(c_3, h_4)	888

TABLE 11

At Node	Alternative No.	Match In	Match Out	Minimum Utility Usage (kw)	New Node	Reasons for Rejecting the Alternative
1	1	(c_3,B)	(c_5,H)	[illegible]	2	
	2	$(c_v H)$	(c_s,H)	[illegible]	3	
	3	(c_3,h_2)	(c_3,h_4)			Rejected: $q(c_r h_4)=1847>U(c.j,h_4)$
	4	(c_s,h_4)	$(c_r h_4)$			Rejected: due to initial test
1	5.	(c_3,h_2) (c_5,h_4)	(c_3,h_4)	.888	4	
2	6	(c_3,h_2)	$(C_3,V$			Rejected: $q(c_v h_4) > U(c,,h_4)$
	7	(c_5,h_4)	$(c_r h_4)$'	•		Rejected: due to initial test
2	8	(c_3,h_2)	(c_3,h_4)	•		Rejected: $q(c_\$,h_4). > U(c_\$,h_4)$

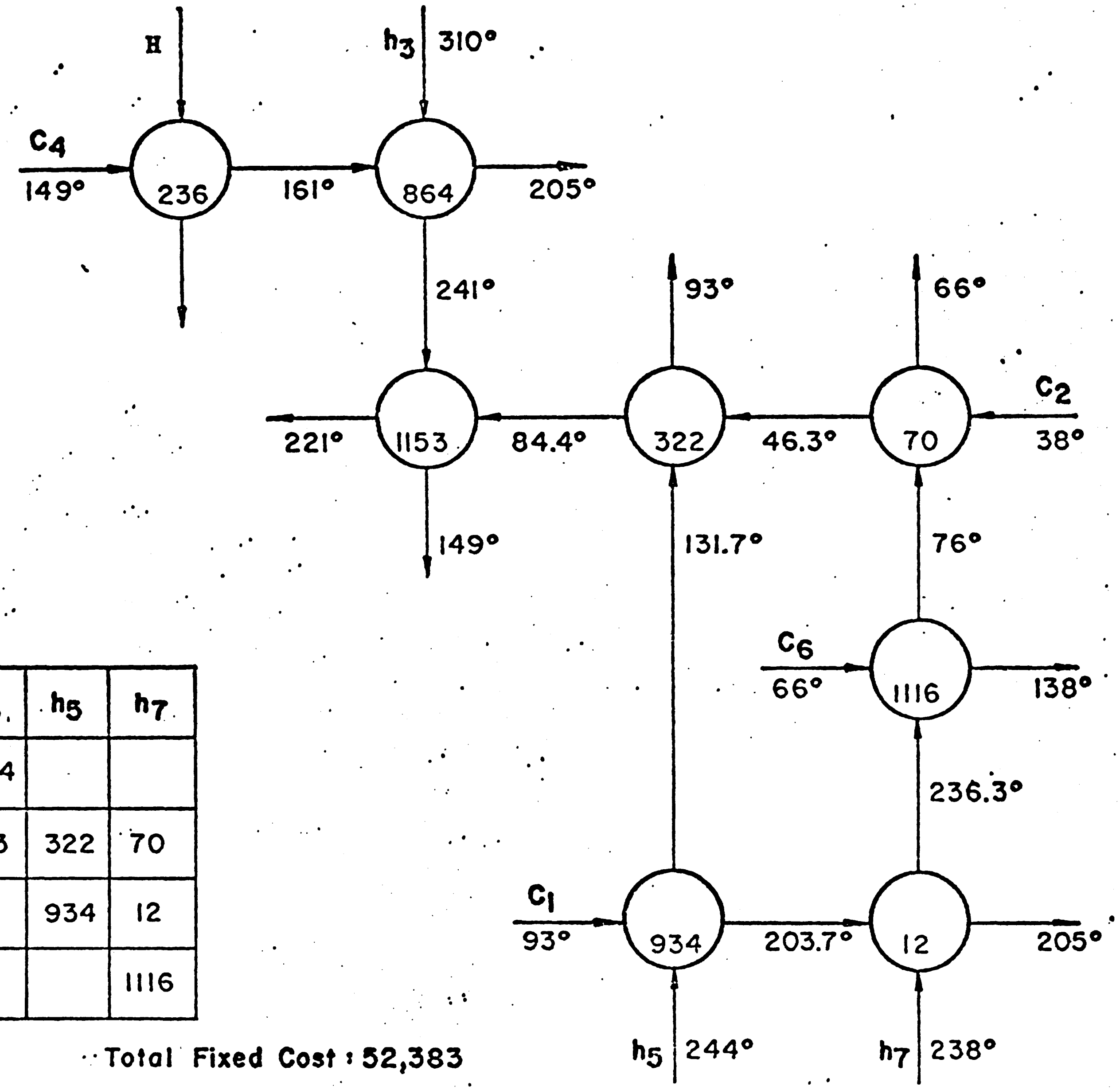
TABLE 11

At Node	Alternative No.	Match In	Match Out	Minimum Utility Usage (kw)	New Node	Reasons for Rejecting the Alternative
3	9		. CcrV	888	5	
	10	(c_s, h_4)	$Cc_r h_4)$			Rejected: due to initial test
4	11	$(c_r H)$	(c_5, H)			Rejected: $q(c_s, h_4) > U(c_\$, h_4)$
	12	(c_3, H)	(c_5, H)			Rejected!.$q(c_5, h_4) > U(c_r h_4)$
5	13	(c_s, h_4)	(c_3, h_4)	888	6	
	14	$(C_3, HJ$	(c_3, h_4)			Rejected: $q(c_r h_4) > U(c_r h_4)$
6	15	(c_3, H)	(c_5, h_4) •			Rejected: $q(c_r h_4) > U(c_r h_4)$

TABLE 12

Network	Total Fixed Cost	Total Heat Exchange Area (sq.m.)
1	51,902	121.6
2	52,928	119.1
4	50,050	117.8
6	52,691	118.0
7	53,715	115.5
8	49,786	113.0
10	49,089	113.6
11	49,376	111.0
13	49,169	111.2
14	48,825	108.8
15*	48,592	107.6
16	49,110	113.4
17	49,994	110.3
18	48,867	113.9
19	48,824	108.9
20	50,223.	114.3

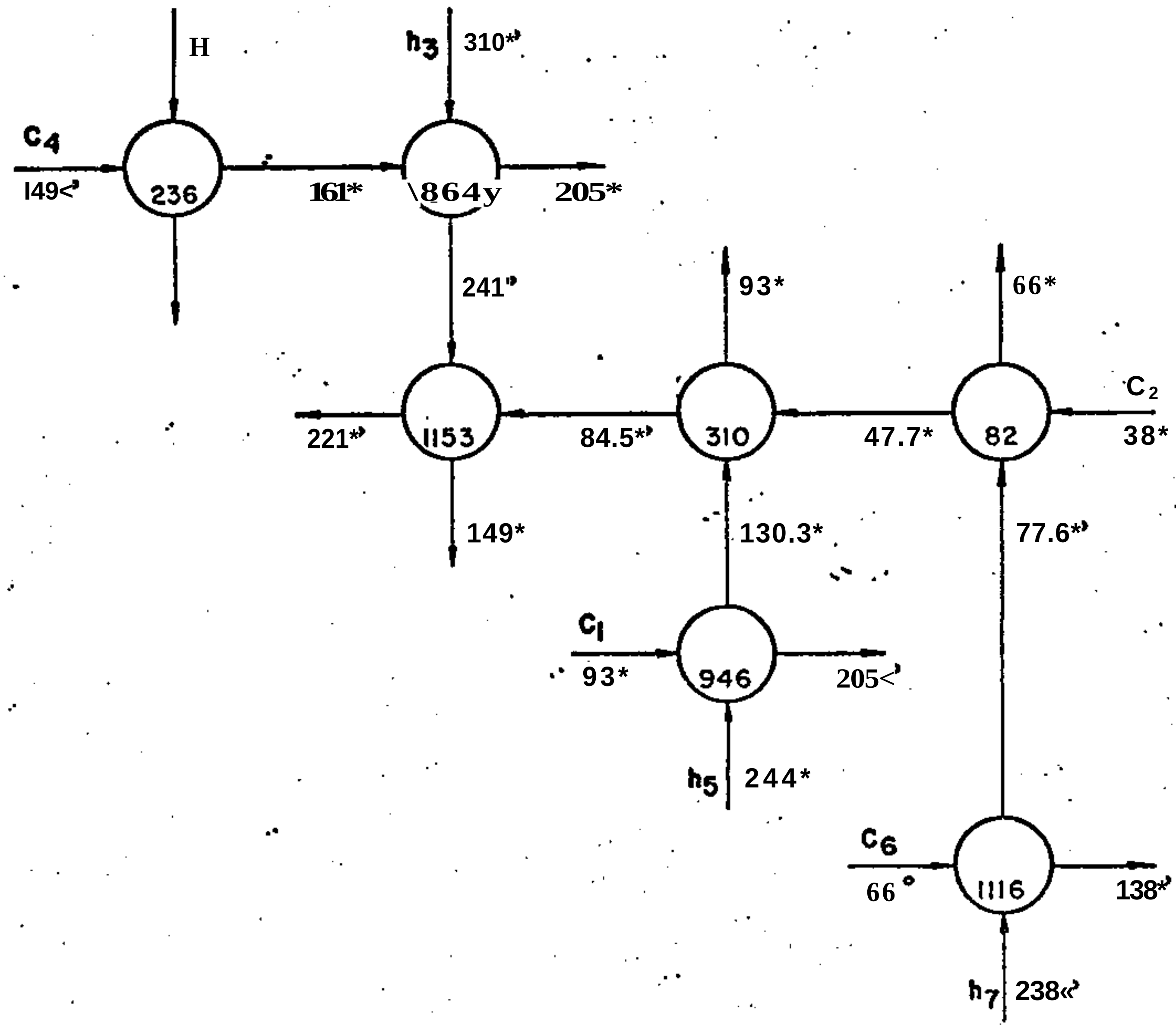
LIST OF FIGURES



	H	h3	h5	h7
C4	236	864		
C2		1153	322	70
C1			934	12
C6				1116

Total Fixed Cost : 52,383

Figure 1



H
h3
310*
c4
149
236
161*
864
205*
241
93*
66*
C2
221*
1153
84.5*
310
47.7*
82
38*
149*
130.3*
77.6*
c1
93*
946
205
h5
244*
c6
66
1116
138*
h7
238

Figure 2

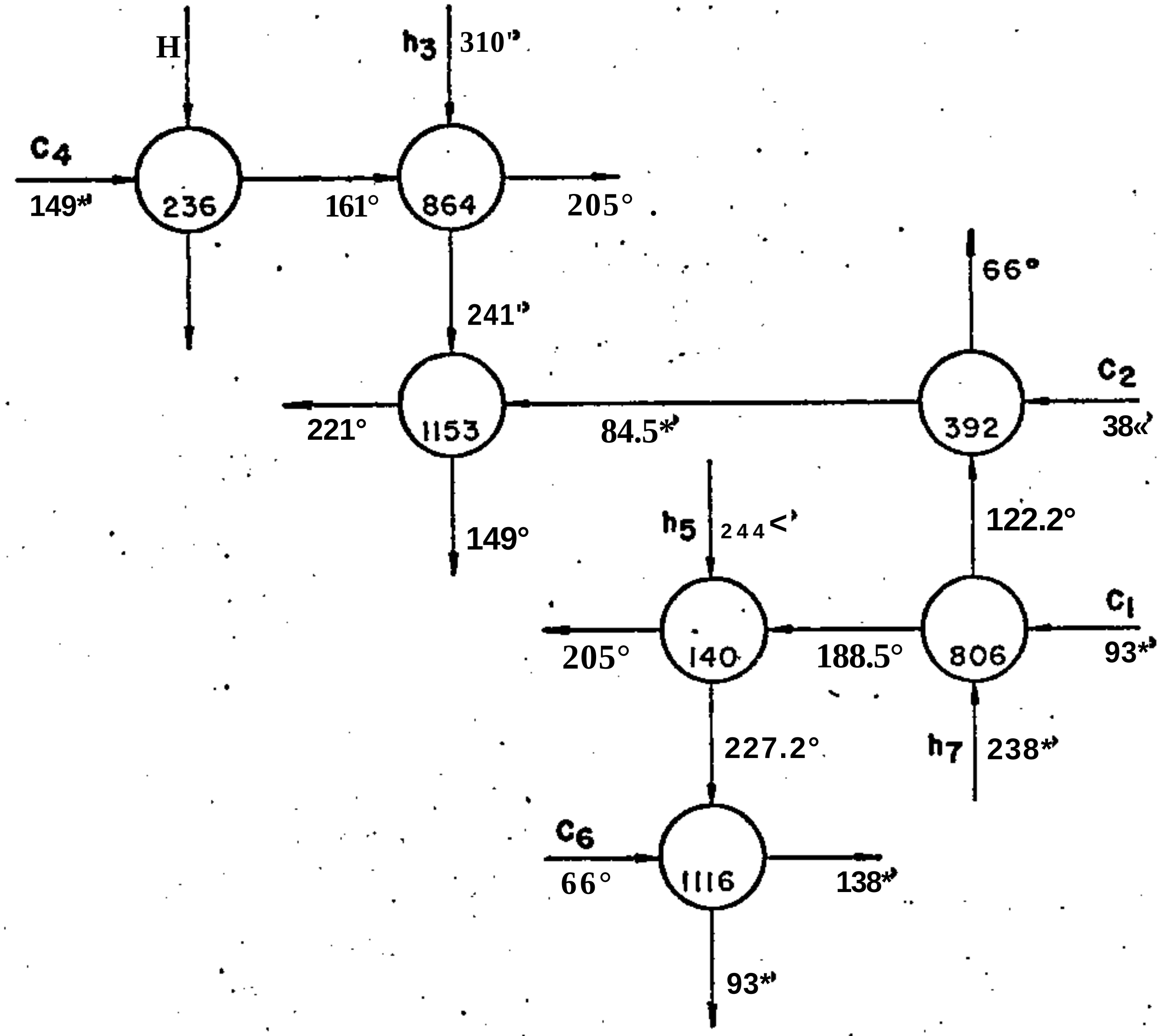
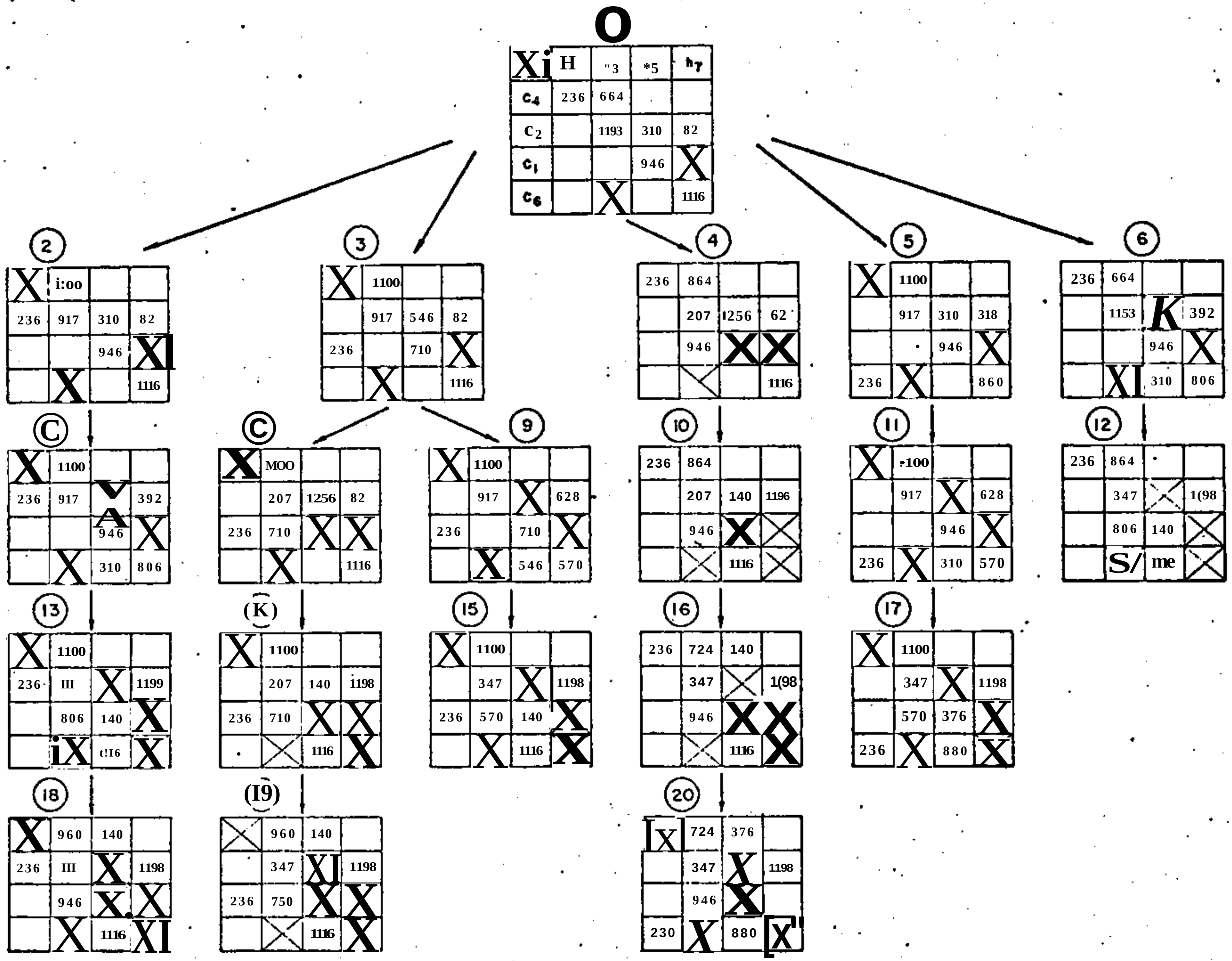

H
h3
310°
C4
149°
236
161°
864
205°
241°
66°
C2
221°
1153
84.5°
392
38°
h5
244°
122.2°
C1
205°
140
188.5°
806
93°
227.2°
h7
238°
C6
66°
1116
138°
93°

Figure 3

Figure '4



1

	H	h_2	h_4
c_5	888	558	
c_1		1569	335
c_3			1512

(A.1) → 2

2

X	H	h_2	h_4
c_5	X	1446	
c_1		681	1223
C_3	888		624

(A.2) → 3

3

	H	h_2	h_4
c_5	X	1446	
c,	888	68!	335
c_3			1512

(A.5) → C

C

	H	h_2	h_4
c_5	888	X	558
c_1		615	1289
c_3		1512	X

(A.9) → 5

5

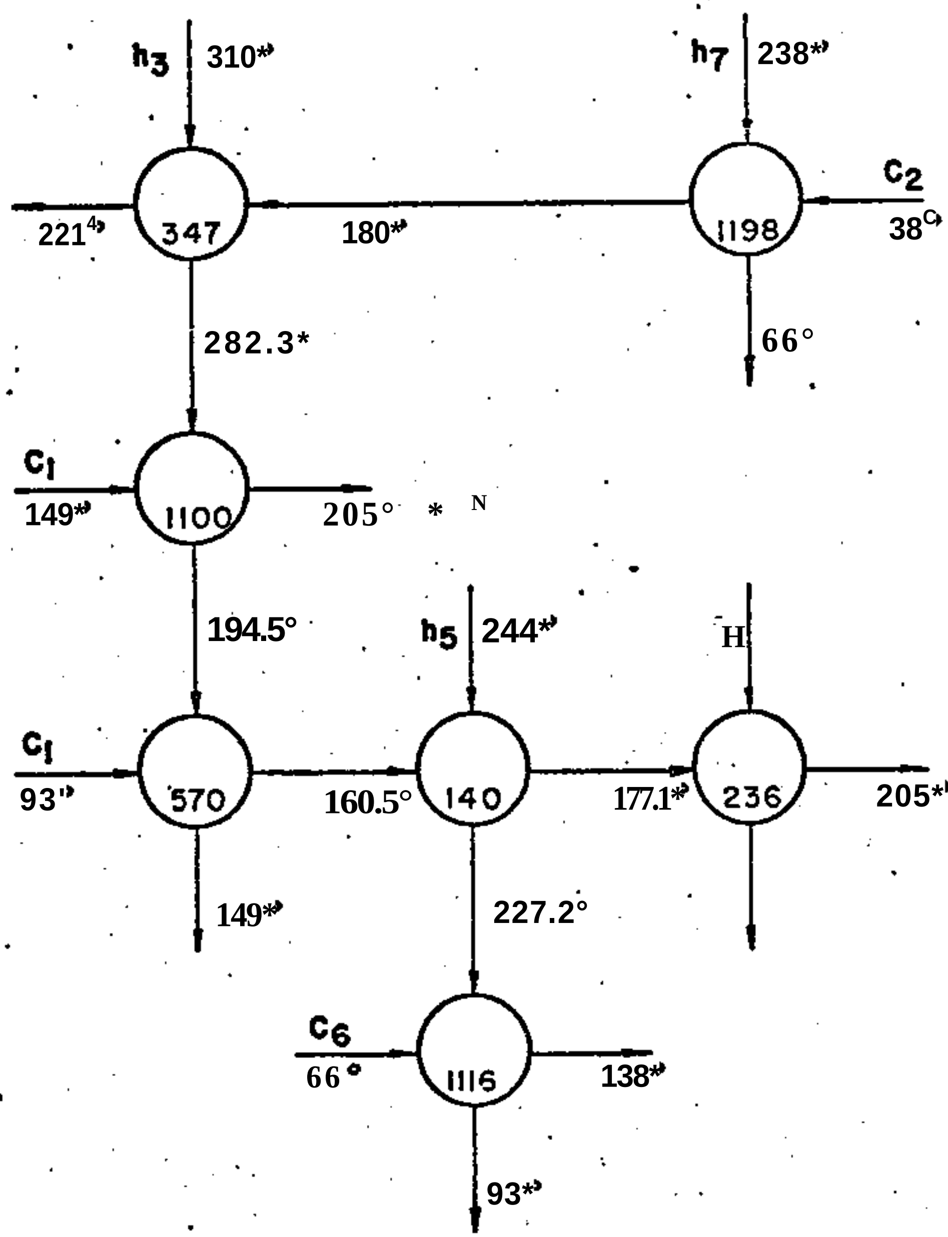
	H	h_2	h_4
c_5	X	1446	
c_1	888	X	1016
C_3		68)	831

(A.13) → A

(A)

	H	h_2	h_4
c_5	X	615	831
c_1	888	X	1.016
c_3		1512	X

Figure 5


h3
310°
h7
238°
C2
221°
347
180°
1198
38°
282.3°
66°
C1
149°
1100
205°
194.5°
h5
244°
H
C1
93°
570
160.5°
140
177.1°
236
205°
149°
227.2°
C6
66°
1116
138°
93°

Figurq 6

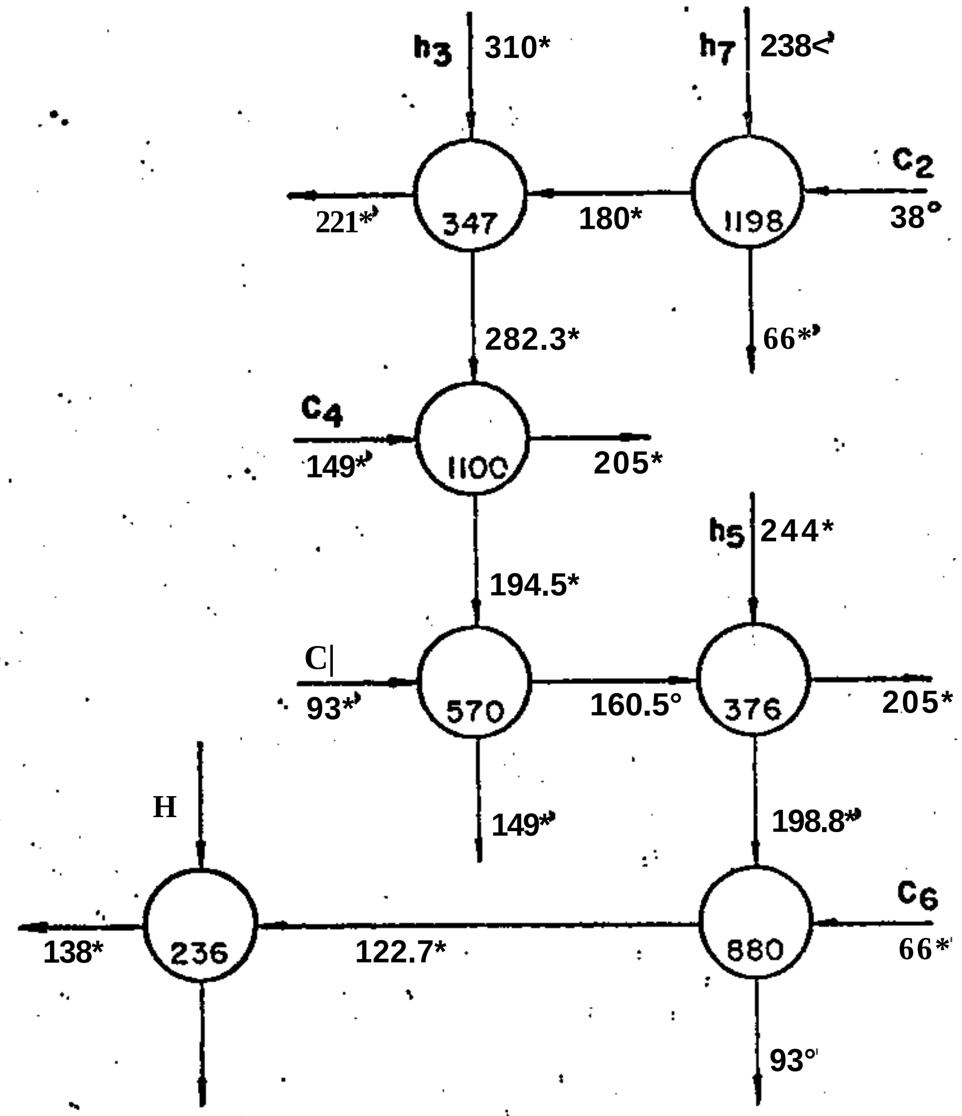

h3
310*
h7
238<
c2
38°
221*
347
180*
1198
282.3*
66*
c4
149*
1100
205*
h5
244*
194.5*
C|
93*
570
160.5°
376
205*
H
149*
198.8*
c6
138*
236
122.7*
880
66*
93°

Figure 7