EROS - A PROGRAM FOR QUICK EVALUATION OF ENERGY RECOVERY SYSTEMS

by

Jigar V. Shah & Arthur W. Westerberg

DRC-06-2-79

January 1979

* Department of Chemical Engineering
University of Florida
Gainesville, FL 32611

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

Sponsered in part by NSF Grant No. ENG-76-80149.

Abstract

The program EROS (Energy Recovery Optimization System) is a flowsheeting package for evaluating and optimizing the performance of simple networks of heat exchangers. As a flowsheeting system it represents a prototype for what a really convenient system might be. Using EROS one can set up an arbitrary structure of heat-exchangers, stream splitters and mixers. Stream flow rates and entry and exit temperatures may be specified, free, or bounded above and/or below. Phase changes may be allowed to occur. Requiring no more user input (such as initial guesses), the program gathers together the modeling equations and appropriate inequality constraints. It then develops solution procedures repeatedly in the course of optimizing, initially to locate a feasible point and in the final stages to take advantage of tight inequality constraints to reduce the degrees of freedom.

Scope

Most of the existing flowsheeting packages are based on Sequential or Simultaneous Modular Approaches. In these approaches each unit is modeled by writing a computer subroutine which converts the input stream and equipment parameter values into output stream values. Systems based on Sequential or Simultaneous Modular Approaches are relatively easy to build, but the penalties paid are the lack of flexibility in the definition of the problem and the requirement for well-defined user specifications.

Equation solving approaches, as followed here and in Leigh, Jackson and Sargent (1974), Hutchison and Shewchuk (1974) and Kubicek, Hlawacek and Prochaska (1976) present an alternative way to treat flowsheets. The flowsheet is represented as a collection of non-linear equations which must be solved simultaneously. In following an Equation Solving Approach the user can specify many of the values for both unit inputs and outputs. The unit equipment parameters can then be calculated to give these desired transformations of inputs to outputs by the unit; in other words, the unit is designed to meet these requirements.

EROS is a prototype flowsheeting system capable of handling systems consisting of simple heat exchangers, mixers and splitters. It is an example of how one might approach the more general flowsheeting problems. EROS contains an optimization capability allowing a rather striking advantage for the user. Those variables for whose values the user has no preference are treated as degrees of freedom by the system. Their values are those selected automatically to minimize annualized cost for the flowsheet. EROS incorporates the general optimization strategy as outlined in Wfesterberg and deBrosse (1973) and demonstrates the applicability and effectiveness of their algorithm.

The two important problems in the design of energy recovery systems are to choose the configuration, and, given a configuration, to choose the design parameters and operating variables. A recent review on the effort directed to choosing a configuration can be found in Nishida, Liu and Lapidus (1977). In choosing a suitable configuration the general trend has been to evaluate networks using the heuristic of setting the minimum allowable approach temperature to 20°F, whereas the economics as stated often advocate a smaller value. In addition, when a stream is split, the need for finding the optimal value for the split fraction has been ignored. Grossmann and Sargent (1977) optimized several heat-exchanger networks and found considerable savings (sometimes as much as 25%).

The problem of optimizing a heat-exchanger network to obtain the most suitable values for the operating variables has been considered in the works of Westbrook (1961), Boas (1963), Fan and Wang (1964), Bragin (1966), and Avriel and Williams (1971). Typically each design problem is formulated and solved for as an optimization problem. Many investigators (Hwa (1965), Takamatsu, Hashimoto and Ohno (1970), Henley and Williams (1972), and Takamatsu et al. (1976)) have combined both the problems choosing a configuration as well as the operating variables, and formulated it as an optimization problem. All the methods mentioned for optimizing over the operating variables require the problem to be cast into a mathematical format. EROS precludes this need because of its capability as a flowsheeting package.

Again EROS is a prototype. Its flexibility is readily appreciated by the user if he has previous experience with other flowsheeting systems. Hopefully this flexibility, however implemented, will become a part of future flowsheeting systems.

Conclusions and Significance

The optimization strategy chosen for EROS proves to be efficient. It is the authors' belief that the number of steps required for convergence is significantly lowered by rederiving a solution procedure every time a constraint violation occurs, and by the use of 'restriction' (Geoffrion (1970)). A very useful feature in EROS is its ability to find a feasible starting point, if none such is provided by the user. The solution yielded by EROS can account for portions of the network that already exist and for irregularities likely to occur in the process streams. Considering the nature of the problem treated, cost per a typical run of EROS seems small. The use of the program may also be extended in carrying out synthesis via structural parameters but with certain reservations (Shah and Westerberg (1976)). A global optimum is not always guaranteed on the application of EROS and a more complete discussion regarding global optimality is presented in Westerberg and Shah (1977).

Introduction

In order to evaluate and optimize heat-exchanger networks it is desirable to have a flowsheeting program which, on being given information about the configuration and stream properties, yields all the required information about the optimal network. Since the program will perform several different tasks it would be very attractive and in many instances necessary for it to possess the following features.

a) An ability to set up solution procedures. The program should gather together the appropriate equations which model the given network. It should also gather together all the relevant inequality constraints. It should be able to derive an appropriate solution procedure for these equations whereby it attempts to eliminate or reduce computational recycles by selecting which variables should be the decision variables and in which order it should use the equations to calculate the remaining variables.

b) An ability to obtain a feasible starting point. If computational recycles are involved in the calculation of the unknown variables of a system, locating a feasible starting point for optimization is not a simple task. In order to save users the time and trouble necessary to find a feasible starting point, the program should be capable of performing such a task on its own.

c) Efficient optimization routines. These would be required for selecting the optimal values for the decision variables.

However, the optimization of a system such as this one raises certain problems. The objective function is highly non-linear and multimodal. Also if phase changes are allowed, continuous derivatives cannot be obtained. These criteria force the use of a search algorithm such as the complex method. In having resigned to the use of the complex method for optimization, one must make all possible efforts to improve the efficiency of the approach for optimization. If, in the process of optimization, several of the inequality constraints are violated, one remedial action is the use of penalty functions, but this modification is inefficient and it increases the number of iterations required for convergence.

Hence, with regard to optimization, a few additional features would be deemed attractive.

d) The ability to rederive a solution procedure. When a constraint is violated, the program should be able to modify the equation set and rederive an efficient solution procedure so that the optimization may be continued with the best computational efficiency possible.

This strategy will lessen the number of iterations required for convergence as compared with the penalty function method. However, it is essential that the saving in computer time thus incurred compensates for the extra time required in rederiving a solution procedure.

e) The use of restriction as a solution strategy. In the presence of a large number of decision variables, some of them can be advantageously set to zero and optimization performed with only the remaining ones as search co-ordinates. Optimization is considered complete when the Kuhn-Tucker optimality conditions are satisfied with respect to the decision variables held at zero value. This strategy aids considerably in reducing the number of active search directions and thus the number of iterations during optimization.

EROS is the authors[1] attempt at developing an optimization program which incorporates all the features discussed above. Figure 1 illustrates the general structure of the EROS system.

The approach taken is to model each unit in a heat-exchanger network functionally by writing overall material and energy balances. The unit models themselves may be very complex internally, but the net effect of the flow sheet level modeling as used here is that each unit satisfies overall heat and material balances. In the current version of EROS only simple models are used. Using the functional equations, the modeling of such a system creates only a few equations per unit, and a solution procedure, or the order in which these equations are to be solved, is found automatically along with the degrees of freedom (or decision variables) to be chosen. The solution procedure sought is one that will eliminate, if possible, computational recycles at this level of modeling.

The above approach is useful because the equations are solved repeatedly as an inner loop to an optimization program. As illustrated in Figure 1, the optimizer directs all the activity. Its primary function is to adjust the decision variables to improve the objective function *J. For this system fi is the annual!zed cost of the equipment plus the annual cost of buying the utulities needed such as steam for the purpose of heating.

To evaluate ^, the optimizer supplies the block labeled "Solve Model Equations[11] with the values it wishes to try for the decision variables u. The remaining problem variables x(u) are then obtained by solving the model equations using the solution procedure which has been automatically generated for them. With u and x(u) values available, constraint violations are checked, and if some are violated, they are identified to the optimizer. Assuming none are violated, the units in the system are sized and an annualized cost *i*> is evaluated. The optimizer notes this cost and changes the decision variable values, with the aim of reducing <J. This calculation sequence is repeated many times during a typical optimization. If constraints are violated, special action is taken, which for this system will result in a modified set of model equations and a need to rederive a solution

procedure for them in an attempt to remove unnecessary computation recycles (this approach is based on the optimization strategies in deBrosse and Westerberg (1973) and Westerberg and deBrosse (1973)). The modified complex optimization algorithm (Umeda and Ichikawa (1971)) is used in searching for the decision variable values, u. It is restarted from the best point found so far each time a new solution procedure is derived.

Data Specification

Adequate data must be supplied to the computer to define the problem. A problem definition requires the following input.

1) the flowsheet structure
   a) the units and how they are interconnected by the streams

2) the unit data
   a) unit type (heat exchanger, splitter, or mixer)
   b) desired equipment parameter specifications, such as heat exchanger area
   c) cost data

3) the stream data
   a) desired specifications or bounds on overall stream flow and/or on temperatures of the stream entering or leaving the network
   b) physical property data (dew point, bubble point, heat capacity as a function of stream phase condition, heat of vaporization)
   c) film heat transfer coefficients as a function of stream phase condition
   d) cost per unit of flow (for utilities)

4) the segment data (Each stream is broken into segments as it passes from one unit to the next, see below. Each segment has its own temperature and flow rate.)
   a) associated stream identifier (i.e. what stream this segment is a part of)
   b) any specifications imposed on flow and temperature for the segment

5) general user specifications (see section on "Special Features" later for an example)

6) guessed set of inequality constraints to be held as equalities to aid the program in establishing a feasible starting point, if this input is desired.

Modeling Considerations

The modeling of a network as done in EROS will be illustrated using the example in Figure 2.

Hie network comprises a single hot process stream H_1 which is split and used to heat two cold process streams C_1 and C_2. It then merges to its exit conditions. Streams C_1 and C_2 are heated further by steam utility streams S_1 and S_2. The network has four heat exchanger units, 2, 3, 5 and 6, one stream splitter unit, 1, and one mixing unit, 4. These units are also referred to as nodes. Hie streams have been broken up into segmetns, of which there are 16 overall. For example, stream C_1 enters node 2 as segment 11. It exits and proceeds to unit 5 as segment 12, and finally leaves the system as segment 13. The naming scheme should now be evident. All the heat exchangers are assumed to be counter-current.

Hie 3 basic building block units used are the heat exchanger, stream splitter and stream mixer. The unit models are written functionally for the example flowsheet in Figure 2 by writing overall material and heat balances.

<u>Unit</u>

1	Material Balance (MB)	$F_2 = dF_1$	(1)
		$F_Q = (I - \hat{} F T$	(2)
	Heat Balance (HB)	$h_2 = \hat{}$	(3)
		$h_3 = h_x$	(4)
2	MB	$F_4 = F_2$	(5)
		$F_{12} = F_{11}$	(6)
	HB	$h_2F_2 + h_{11}F_{11} = h_4F_4 + h_{12}F_{12}$	(7)
3	MB	$F_5 - F_3$	(8)
		$F_{15} \quad F_{14}$	(y)
	HB	$h_3F_3 + h_{14}F_{14} = h_5F_5 + h_{15}F_{15}$	(10)
4	MB	$F_6 = F_4 + F_5$	(U)
	HB	$h_6F_6 = h_4F_4 + h_5F_5$	(12)
5	MB	$F_8 = F_?$	(13)
		$F_{13} = F_{12}$	(14)
	HB	$h_?F_7 + h_{12}F_{12}$ "Vs $+ h_{13}F_{13}$	(15)
6	MB	$F_{10} = F_9$	(16)
		$F_{16} = F_{15}$	(17)
	HB	$h_9F_9 + h_{15}F_{15} = h_{10}F_{10} + h_{16}F_{16}$	(18)

In addition to these 18 equations, the associated inequality constraints and the equipment sizing and costing relations can be written.

A basic inequality constraint is that at no point in any exchanger should the hot stream temperature equal or fall below that of the cold stream. Referring to the heat-exchanger in Figure 3a, this constraint is usually expressed as

$T_1 \wedge T_f + D$, where D is minimum allowable approach temperature

$T_2 \geq T_3 + D$

However the temperatures could cross over internally and the above constraints may not be adequate to detect it, particularly when a stream passes through a phase change. A check should therefore be made at several points along the exchanger to prevent a "crossover[11] of temperatures. We use very simple models in EROS at present so each stream is considered to have three constant "heat capacities,[11] one for liquid phase, one for vapor phase and a pseudo-heat capacity for phase transition. Pure components for example are modeled to have a phase transition over a very small but non-zero temperature interval. The pseudo-heat capacity is selected so the temperature interval times it gives the heat of vaporization. Because of this approach, several numerical problems are prevented and also crossover temperature constraints need only be checked at the exit and entrance and at the dew points and bubble points internally in an exchanger if they occur there.

The final set of constraints indicate that a positive heat transfer must occur; that is, the hot stream must be cooled and the cold stream heated. These are

$T_2 * T_1$ and $T_3 * T_4$

All the constraints associated with a typical exchanger such as the one shown in Figure 3 are listed in Table 1 with an appropriate code so that they can be precisely identified by numbers. An example is the interior constraint being checked at the hot stream dew point temperature, assuming this dew point temperature occurs within the exchanger. For this constraint the system creates the identifier (NODE number times 1000) plus 4. For node number 3 then the constraint identifier created is (3*1000) + 4 = 3004. No further constraints are needed for the splitter and mixer units.

The sizing calculation for an exchanger is to evaluate its area. This calculation can be very involved, but for preliminary design purposes may in fact be simplified by using log mean temperature driving forces and by assuming film coefficients based on the fluid, whether it is heating or cooling and whether it is boiling or condensing (see Perry et al. (1973)). The exchanger may again, for simplified design purposes, be considered to operate in zones as indicated in Figure 3. Each zone is then sized using the appropriate film coefficients and log mean temperature driving forces. Each zone is then assumed to be a separate heat exchanger which conforms to observed industrial practice. Other costing strategies are of course possible and EROS could readily be modified to accommodate them.

To place a crude cost on the exchanger we use an equation of the form (see Guthrie (1969))

$$\text{cost} = f_M f_P (aA^m)$$

where f_M is a materials factor, f_P a pressure factor, and A the area of a zone within an exchanger. The terms a and m are constants, with m being about 0.6 to 0.8. Constant costs are assumed to occur for the splitter and the mixer units and thus no cost is evaluated for them.

The last source of equations is the evaluation of physical propertieis. The system must be able to convert from stream temperature (and vapor fraction for a pure component in the two phase region) to enthalpy and vice versa. A cooling curve could in principle be provided for the stream if the stream is assumed to be at a constant pressure. Figure 4 illustrates a cooling curve, where T_D and T_B are the dew and bubble point temperatures, respectively. As stated earlier, we assume the cooling curve to comprise three straight line segments, one for each phase condition. The user must provide the dew and bubble points, and again as stated earlier, even for pure components the dew point must be greater than the bubble point even if only by a fraction of a degree.

Properties such as thermal conductivities and densities should also be provided if the film coefficients are to be determined from correlations. For design purposes, we require the user to provide typical values for film coefficients, thus these other fluid properties will not be needed here.

Deriving the Solution Procedure and Solving Model Equations

Consideration will now be given to developing a solution procedure and then solving the example problem. First the system must gather together the necessary equations, or at least establish their structure, so that a solution procedure may be prepared. The desired solution procedure should eliminate all recycle loops in the computations if possible or attempt to minimize their number.

The initial solution procedure will ignore all but the inequality constraints suggested by the user to be tight (i.e. held as equality constraints) at the solution. Thus for our example problem EROS sets up automatically the 18 heat and material relations shown in the last section. Assuming that the user has requested that constraints 55 and 33 be included, the additional relations

$$F_7 = F_{7LB} + \sigma_{55} \tag{19}$$

$$T_3 = T_5 + \sigma_{33} \tag{20}$$

are also set up where F_{7LB} is the lower bound for flowrate given by the user for F_7.

The inequality constraints have been converted to equality by the slack variables σ_{55} and σ_{33} which are then required by EROS to be non-negative. When the solution procedure is derived σ_{55} and σ_{33} will be required to be decision variables and will be given an initial guess of zero. In this way F_7 and T_3 will be forced to be equal to F_{7LB} and T_5, respectively.

Relation (20) is in terms of temperatures rather than enthalpies. Hence the following relationships must also be added

$$T_3 = f(h_3) \tag{21}$$

$$T_5 = f(h_5) \tag{22}$$

The system can implicitly account for equations (20), (21) and (22) by the single expression

$$h_3 = f(h_5, \sigma_{33}) \quad . \tag{23}$$

An incidence matrix which shows which variables occur in which equations can now be created. It will be used by EROS to derive the solution procedure for the equations. However its size can be significantly reduced. Note that a large number of equations simply equate one variable to another. Equations (3), (4), (5), (6), (8), (9), (13), (14), (16) and (17) are precisely of this form. These equations will automatically be satisfied if the variables so equated are assigned the same storage location. Using this approach these equations are deleted by EROS.

Many of the variables in the incidence matrix are in fact specified by the user and are thus' fixed in value for the problem. Let the following be specified in data input for the example in Figure 2.

Flows	$^{F}1>^{F}11>^{F}14'^{F}7LB$
Enthalpies	h^ ,h^ ,h^ , h_g ,h^ f h^ , h ^ 9 h^ skj^ f^g

These specified variables along with the slack variables cr^c and <j«~ (which EROS requires to be decision variables) are eliminated from the incidence matrix. The resulting and much reduced matrix is illustrated in Table 2. Only variables which EROS permits to be dependent variables (i.e. calculated in terms of the independent or decision variables) remain.

A modification involving both a simplification and an extension of the Christensen and Rudd (1969) algorithm is applied to determine the solution procedure. The algorithm is given in detail in Shah (1978). The solution procedure that results on the application of this algorithm to the incidence matrix of Table 2 is shown in Table 3. We show here only the results which for this problem are easy to understand. The variables listed are calculated from the corresponding equations in the order indicated.

Note that EROS has derived a solution procedure in which there is an iteration loop involving the single 'tear' variable F_2 (from steps 4 to 9). F_9 (= F_4) appears in equation (12) and the iterations between steps 4 and 9 are continued until the value of F_2 guessed at step 4 is essentially the same as the value of F_2 calculated from equation (12) in step 9. This iteration loop cannot be eliminated — except perhaps by algebraic manipulation which EROS cannot do.

The execution of the solution procedure, that is, calculation of the variables from the equations assigned to them, is termed "Solve Model Equations" in Figure 1. Corresponding to every unit, a subroutine is required to calculate any variable involved in the heat and mass balances of the particular unit. These subroutines may be supplied by the user for more sophisticated models.

Starting the Problem: Finding a Feasible Point

If the user has not provided any information to aid in obtaining a feasible starting point, a modified version of an algorithm by deBrosse and Westerberg (1973) is used. As mentioned earlier, a significant effort would be required on the part of a user to provide a feasible starting point if computational loops are involved in solving model equations. Computational loops are almost inevitable in complex networks. However, the user does have the option of providing a feasible starting point.

The deBrosse and Westerberg (1973) algorithm uses an indirect approach. It hypothesizes that a subset of constraints has no feasible region and then attempts to verify this conjecture. If successful, the subset is identified as infeasible and obviously no feasible point exists.

If unsuccessful, either a new hypothesis can be generated or the algorithm has indirectly found a feasible point. A modified version of this algorithm with an application to a heat-exchanger network is presented in Shah (1978).

The Optimization Strategy

The optimization strategy is modeled after the algorithm presented in Westerberg and deBrosse (1973). The algorithm is invoked once a feasible point is available.

The sets of inequality constraints are divided into three sets.

V_T •* The set of constraints being held as equality constraints. That is, their slack variables are held at zero for the next optimization step.

V_R = The set of constraints present in the equation set as equality constraints with the difference that their slack variables (bounded below by zero) are used as search co-ordinates.

V_ — The set of all remaining constraints.

V,, s {v_,V_}, the set of all inequality constraints participating in the current optimization step.

Solution procedures are modified as inequality constraints are moved from one set to another. Adding constraints to the set being held, V_T tends to aid the optimization process by reducing the dimension of search space for what is usually a marginal or no added burden in solving an enlarged set of equations.

As optimization proceeds, the values of all variables (including slack variables) are stored for the point that yields the best value for the objective function. Hence, even when V_- is changed, optimization can be and is started at the best point discovered up to this moment. This modification makes a significant improvement to the Westerberg and deBrosse (1973) method. Figure 5 illustrates the typical dilemma faced by the Westerberg and deBrosse optimization algorithm when stepping from a current best point, point 'e', through one or more inequality constraints to point 'f'. At point 'f' the constraint g_1 is detected as being violated. The algorithm will respond by changing the solution procedure so that the slack variable a_1 for g_1, becomes a decision variable. The other decision variable will be either x_1 or x_2. There are several options now as to where the optimization may be started. The algorithm could hold x_1 or x_2 (whichever is selected as the decision variable) at its current value and find the point where a_1 is zero leading to point P_1 or P_2, respectively. Alternatively it could attempt to locate p_3 by searching along the direction leading from 'e' to 'f'. All of these options can, and often do, lead to a next point which has a higher, and thus worse value for the objective function. By saving all the variable values for the best point, the search can always start, even after developing a new solution procedure, from that point, that is from point 'e'. This change reduces cycling because a change in the solution procedure cannot lead to a point that is worse.

The actual search strategy used is the modified complex method (Umeda and Ichikawa (1971)). The complex method is considered suitable because gradients are not required. The treatment of phase changes creates discontinuities in first derivatives of functions. Details concerning the algorithm for optimization and its application to a simple example are presented in Shah (1978). The algorithm is quite complex as it handles automatically many degeneracies which can arise.

Special Features

a) Using existing heat-exchangers

The program may be used to analyze a network where some of the exchangers may already be available. The program assumes that these exchangers are available at no cost. In Figure 6, an exchanger with an area of A_1 is available. On analysis, however, it is discovered that an exchanger with an area A_2 is required at that particular site in the network. In the program, the costs assumed for different conditions of A_1 and A_2 are shown in Figure 6. The physical significance of 1) is that a by-pass (such as plugging some tubes) will be used within the specified exchanger. Note that the exchanger is free and still it is not economic or perhaps not feasible to use it entirely within the network for this case. 2) implies that an exchanger with area equal to $A_2 - A_1$ must be purchased in addition to the exchanger already available.

b) Reliability analysis

The program has been extended to permit its use in preliminary reliability studies. The reliability studies will be demonstrated with the help of an example. Figure 7a represents a simple network as it operates under normal conditions. Cold process streams C_1, C_2, and C_3 are heated to their final temperatures with the help of a hot process stream H_2 and a flue gas H_1. The flow rate and the outlet temperature of stream H_1 are undefined but are required to be within specified bounds. Now let us assume that two abnormal occurrences take place separately, for certain periods of the year, namely,

1) Stream H_2 is unavailable.

2) Heating of Stream C_2 is no longer required.

Case 2) could arise for example if C_2 is steam being raised and sent to another process. Occasionally this other process does not operate and cannot use the steam.

The aim now is to find an optimal network such as the one shown in Figure 7a, fully provided for to meet the contingencies with the aid of by-passes around exchangers and/or with the aid of auxiliary exchangers. For this example the designer permits a change in the flow of stream H_1 and its outlet temperature for the emergency situations, provided they stay within specified bounds. In the case of failure mode 2), a cooling utility stream is proposed to cool stream C_2 to 230° so that it may be recycled again to be heated to 250°, as it is thought this might aid because it will maintain a semblance to the normal operation, the flow rate of C_2 is allowed to vary between 0 and 70,000 in this instance.

Figure 7b represents the network when stream H_2 is unavailable, and Figure 7c when stream C_2 is not required to be heated. Additional user specifications to those shown in Figure 7 are presented in Table 4. In order to find the optimal operating system, networks in Figure 7a, 7b and 7c are optimized together. The objective function ϕ is given by

$$\phi = c_{H_1}F_{H_1} + c'_{H_1}F'_{H_1} + c''_{H_1}F''_{H_1} + c''_{UC}F''_{UC} +$$

$$\sum_{i=1}^{5} \text{(cost of exchanger area at the site i)}$$

where c_i and F_i represent the cost coefficient and the flow rates of stream i, respectively. The cost coefficient c_i should reflect the expected fraction of the year that the network is in the particular state being represented. For example, for the problem in Figure 7 it is assumed that the networks in Figure 7a, 7b and 7c are operational 777o, 11.57o, and 11.5% of the time in a year, respectively. Hence if c_{H1} is 0.1, then c'_{H1} and c''_{H1} are 0.015 each.

The cost of exchanger area at a site will be illustrated with the help of an example. At node 2, exchangers of different areas are required in Figure 7 and are denoted as $A^2_{\sim}$, A^2_{-f} and $A^{\wedge}_n$, respectively. Assume that $A^{\wedge}$, is smallest and $A^{\wedge}_n$ the largest area.

The cost of exchanger area at site 2 is defined as

$$35\left\{(A_{2,})^{0.6} + (A_2 - A_{2,})^{0.6} + (A_{2lt} - A/)^{.6}\right\}$$

This manner of costing areas in doing reliability analysis appears to be a good formulation of the real system. If, because of some departure from the normal mode, more exchanger area is required at a particular site, then one must pay for the auxiliary exchanger. If the exchanger area required is more for the normal mode, then it is supplied by two exchangers in series which are normally operating with a by-pass around one for the emergency situation. The results obtained on optimization of the system in Figure 7 are shown in Table 5.

Discussion

A typical application of EROS to a heat-exchanger network is illustrated by the example in Figure 8, the stream specifications for which are shown in Table 6. Stream I is a flue gas and three streams of type IV and two of type VIII are steam and cooling water utility streams, respectively. The flow rates for these streams are not defined but the total flows are required to be within specified bounds for each type. Some of the streams in the network change phase and are characterized by a dew and a bubble point. Thus the network is analyzed to ensure that minimum approach temperature violation does not occur inside the exchanger with these streams owing to discontinuities at the dew and bubble points. In fact, at the optimal solution for the example in Figure 8 both streams III and VII change phase inside node 5, and the minimum approach temperature constraint at the bubble point of stream III is operative. The approach temperature constraint between streams V entering and II leaving in node 3 is also operative at the solution. Exchanger 3 was assumed available already with an area of 1500 units. The objective function Φ is defined as

$$\Phi = \sum_{\text{streams, I,IV,VIII}} \text{cost coefficient} * \text{flow rate}$$

$$+ 35 \sum_{i=1 (\neq 7,8)}^{13} (A_i + 10)^{0.6} - 10^{0.6}$$

where A_i is the area associated with exchanger i. In the calculation of cost associated with an exchanger, the relation

$$\text{cost} = 35\,\{(A + 10)^{0.6} - 10^{0.6}\} \qquad (1)$$

rather than

$$\text{cost} = 35\,A^{0.6} \qquad (2)$$

because whenever the area of an exchanger currently at zero value is increased, the cost for the exchanger as calculated from (2) increases abnormally compared to the change in cost for the rest of the system. The modification as shown in relation (1) dampens this ill-behaved effect.

There are 8 decision variables for this problem and the optimum results after 328 iterations from an infeasible starting point. The stopping criterion is a 1×10^{-5} difference between the worst and best objective function values in the current set of points retained by the complex algorithm. The value of *6* at the optimum is 152,109 \$/yr.

Results for 10 examples are shown in Table 6. In all the examples the feasible point results in very few iterations. It may be observed that the time required for the rewriting of solution procedures after finding a feasible point is relatively small as compared to the time taken for function evaluations during optimization. The maximum ratio of these two times occurs in example 6, but it is still less than 1/3. This observation indicates that the penalty paid for rewriting solution procedures whenever constraint violations occur is indeed very small.

The figures shown for time in Table 4 are those required on IBM 360/67. The cost per second of CPU time is about 1.4 cents and the longest run (example 10) cost \$8.56 for complete execution while example 1 cost \$0.40. The size limitations are 50 process streams, 25 nodes and 150 stream segments in the current version of the program. The program is fairly well tested and gave satisfactory results when used for sixteen different problems set up by students in a recent design course.

Acknowledgement:

This work was sponsored in part by NSF Grant No. ENG76-80149.

LITERATURE CITED

Avriel, M. and A.C. Williams, "An Extension of Geometric Programming with Applications in Engineering Optimization," J. of Engr. Math., 5, 187 (1971).

Boas, A.H., "Optimization via Linear and Dynamic Programming," Chem. Eng., 70, 85 (1963).

Bragin, M.S., "Optimization of Multistage Heat Exchanger Systems," Ph.D. dissertation, New York University (1966).

Christensen, J.H. and D.F. Rudd, "Structuring Design Computations," AIChE J., 15, p 1184 (1969).

deBrosse, C.J. and A.W. Westerberg, "A Feasible-Point Algorithm for Structured Design Systems in Chemical Engineering," AIChE J., 19, 251 (1973).

Fan, L.T. and C.S. Wang, "The Discrete Maximum Principle," Wiley, N.Y. (1964).

Geoffrion, A.M., "Elements of Large-Scale Mathematical Programming," the Rand Corporation, R-481-PR, Santa Monica, Calif., November (1969).

Grossman, I.E. and R.W.H. Sargent, a personal communication (1977).

Guthrie, K.M., "Capital Cost Estimating," Chem. Eng., pp 114-142, March 24 (1969).

Henley, E.J. and R.A. Williams, "Graph Theory in Modern Engineering," Academic Press, N.Y. (1973).

Hutchison, H.P. and C.F. Shewchuk, "Computational Method for Multiple Distillation Towers," Trans. Instu. Chem. Eng., 52 (4), p 325 (1974).

Hwa, C.S., "Mathematical Formulation and Optimization of Heat Exchanger Networks Using Separable Programming," Symp. Series No. 4, pp 101-106, AIChE-IChemE Joint Meeting, London (1965).

Kubicek, M., V. Hlawacek and F. Prochaska, "Global Modular Newton-Raphson Technique for Simulation of an Interconnected Plant Applied to Complex Rectification Columns," Chem. Engr. Sci., 31, pp 277-284 (1976).

Kuhn, H.W, and A.W. Tucker, "Non-Linear Programming," in Proc. 2d Berkeley Symp. on Math. Statistics Prob., 481, Univ. of Calif. Press, Berkeley (1951).

Leigh, M.J., G.D. Jackson and R.W.H. Sargent, "SPEED-UP — A Computer-Based System for the Design of Chemical Processes," paper presented at CAD-74, Imperial College, London, England, Sept. 24-27, 1974.

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

Perry, R.H. and C.H. Chilton, Chemical Engineering Handbook, 5th Edition, McGraw-Hill, New York, pp 10-39 to 10-42 (1973).

Shah, J.V., "The Optimization of Energy Recovery Systems," Ph.D. dissertation, University of Florida (1978).

Shah, J.V. and A.W. Westerberg, "Process Synthesis Using Structural Parameters: A Problem with Inequality Constraints," AIChE J., 23, 378 (1977).

Takamatsu, T., I. Hashimoto and H. Ohno, "Optimal Design of a Langle Complex System from the View Point of Sensitivity Analysis," Ind. Engng. Chem. Proc. Des. and Dev., 9, 368 (1970).

Takamatsu, T., I. Hashimoto, H. Nishitani and S. Tomita, "The Optimal Design of Large Complex Chemical Processes — Development of Max-Sensitive Method," Chem. Engng. Sci., 31, 705 (1976).

Westbrook, G.T., "Use This Method to Size Each Stage for Best Operation," Hydrocarbon Processing, 40, 201 (1961).

Westerberg, A.W. and C.J. deBrosse, "An Optimization Algorithm for Structured Design Systems," AIChE J., 19, 335 (1973).

Westerberg, A.W. and J.V. Shah, "Assuring a Global Optimum by the Use of an Upper Bound on the Lower (Dual) Bound," submitted for review, Comput. Chem. Engng. (1978).

Nomenclature

a	constant in heat exchanger equipment cost equation
A_i	area of heat exchanger i
AT	minimum approach temperature between steams which are indicated is active at optimal solution
$c_i^{\#}$	cost coefficient for stream i, $/unit flow
C_i	cold stream i (a cold stream is to be heated)
C_P	heat capacity
CPU	central processing unit for a computer
CW	cooling water
D	minimum allowable approach temperature within a heat exchanger
F_i	flow rate for stream segment i
f_M	materials factor for heat exchanger equipment cost equation
f_P	pressure factor for heat exchanger equipment cost equation
g_i	inequality constraint i
HB	heat balance equation(s)
h_i	enthalpy per unit of flow for stream segment i
H_i	hot stream i (a hot stream is to be cooled)
1AT	minimum approach temperature constraint inside heat exchanger is active at optimal solution
LB	lower bound (may be a subscript)
m	constant exponent in heat exchanger equipment cost equation. Typically around 0,6 to 0.8.
MB	material balance equation(s)
NODE	node number. Each piece of equipment is a node and is given a unique node number.
SEG	stream segment number
S_i	steam utility stream i
ST	steam

T_B	bubble point temperature
T_{BPC}	bubble point temperature for cold stream
T_{BPH}	bubble point temperature for hot stream
T_D	dew point temperature
T_{DPC}	dew point temperature for cold stream
T_{DPH}	dew point temperature for hot stream
T_i	temperature of stream segment i
u	vector of decision variables (independent variables)
UB	upper bound (may be a subscript)
U_i	heat transfer coefficient
V_C	union of index sets V_T and V_R. The set of all inequality constraints which are actively being used during the current optimization step.
V_R	index set of all inequality constraints whose slack variables (which convert the inequalities to equalities) are being used as decision variables for the current optimization step
V_S	index set of all inequality constraints not in set V_C
V_T	index set of inequality constraints which are being held as equality constraints for current optimization step
x	vector of dependent variables
x_i	element of the vector x

Greek

a	split factor for a stream splitter unit
a_i	slack variable i
i>	objective function

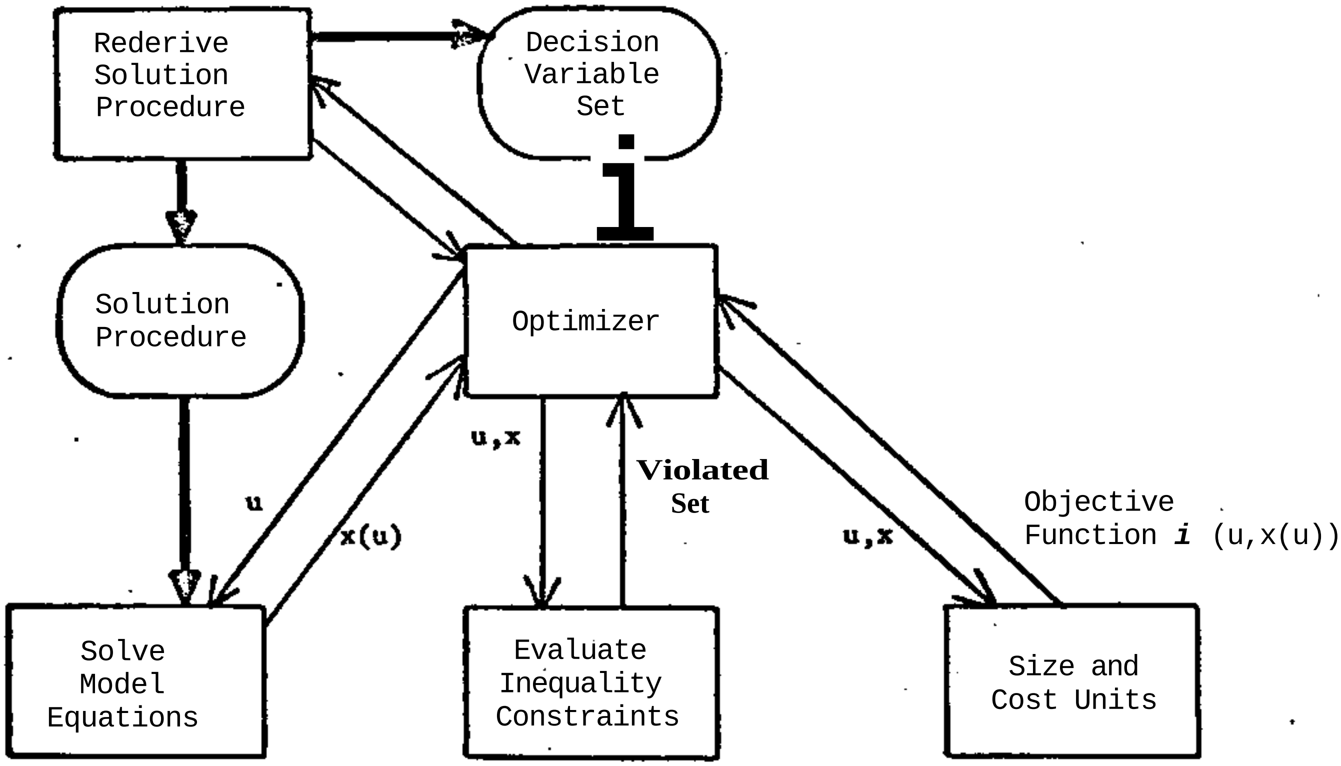
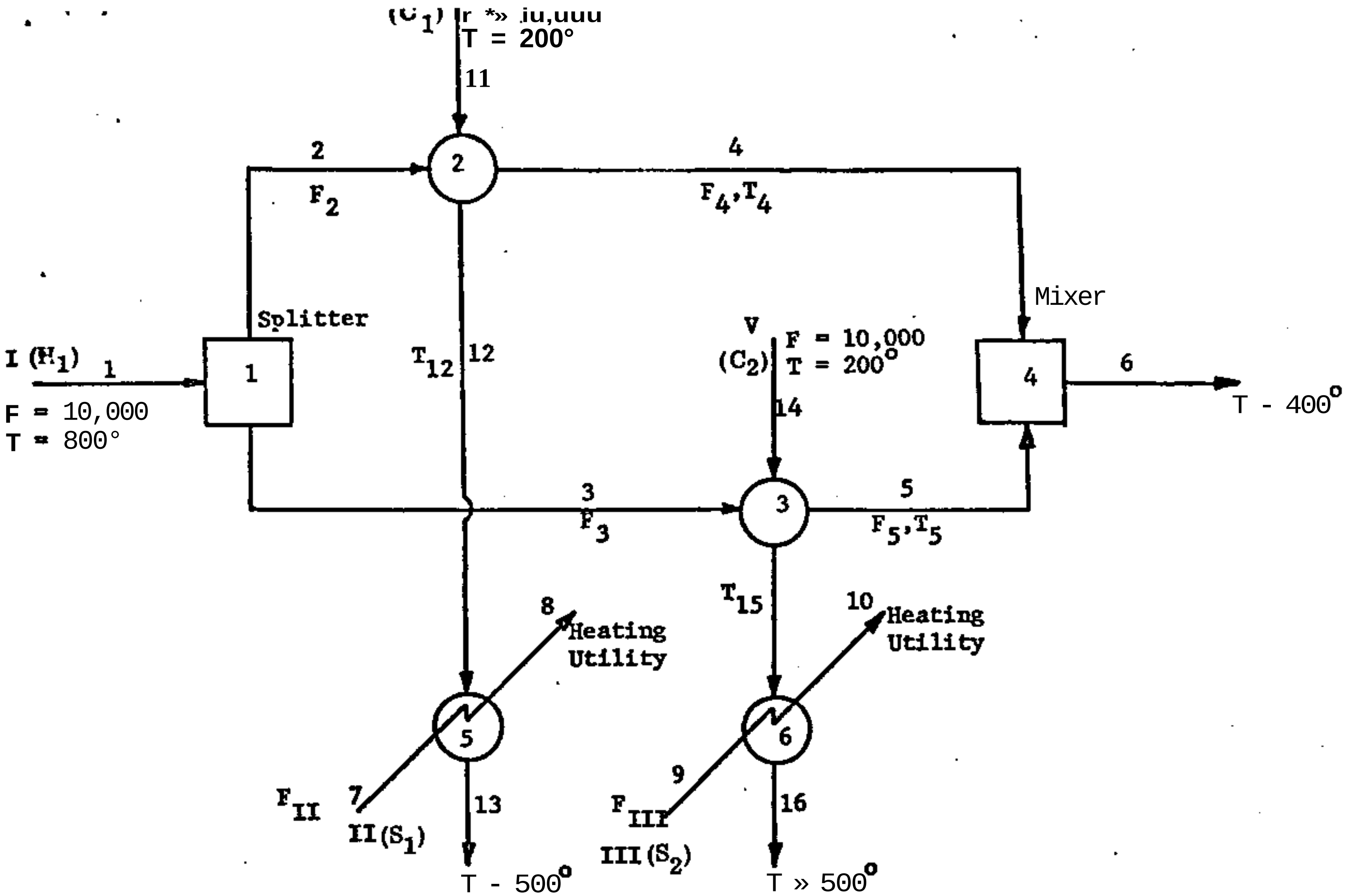


Fig. 1. Structure of an Optimizing System.



$C_p = 1$ $\quad U_2 = 25$ $\quad U_3 = 20$ $\quad U_5 = U_6 = 50$

For Stream II $\quad T_{in} = T_{out} = 600°$ $\quad$ Heat of Vaporization = 800

Cost multiplier per unit flow rate = 0.4

For Stream III $\quad T_{in} = T_{out} = 600°$ $\quad$ Heat of Vaporization = 800

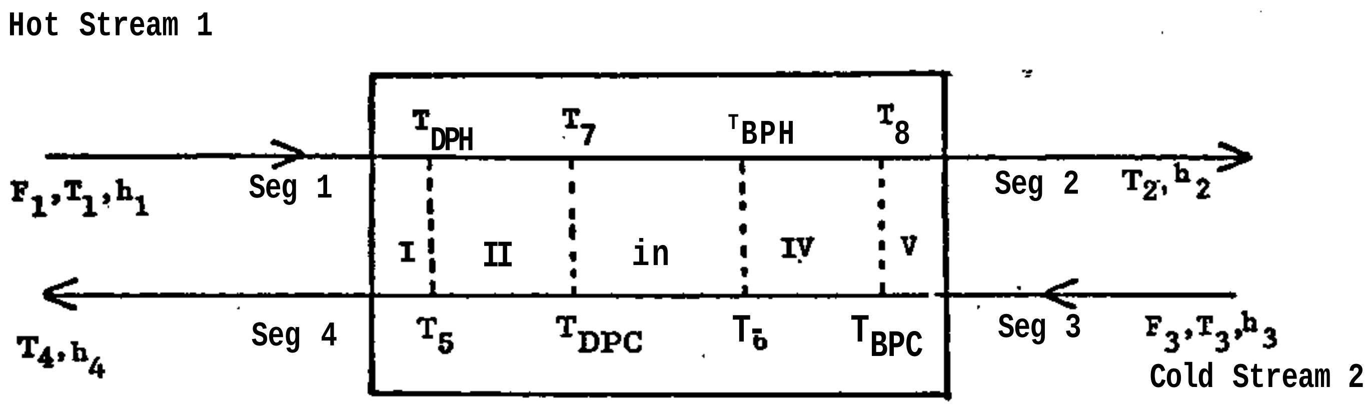
Cost multiplier per unit flow rate = 0.2

Constraints: $\quad 2500 \le F_2, F_3 \le 7500$, $\quad 300° \le T_{12}, T_{15} \le 500°$

Aim: To minimize $\sum_{i=2,\, i\neq 4}^{6} Cost_i + 0.4\, F_{II} + 0.2\, F_{III}$

where $Cost_i = 35\,(Area_i)^{0.6}$. $Cost_i$ and $Area_i$ are associated with exchanger i.

Fig. 2. An Example Problem



T_{DPH} = Dew point temperature, hot stream

T_{BPH} = Bubble point temperature, hot stream

T_{DPC} = Dew point temperature, cold stream

T_{BPC} = Bubble point temperature, cold stream

T_5 = Temperature of the cold stream at a location where the hot stream temperature is T_{DPH}

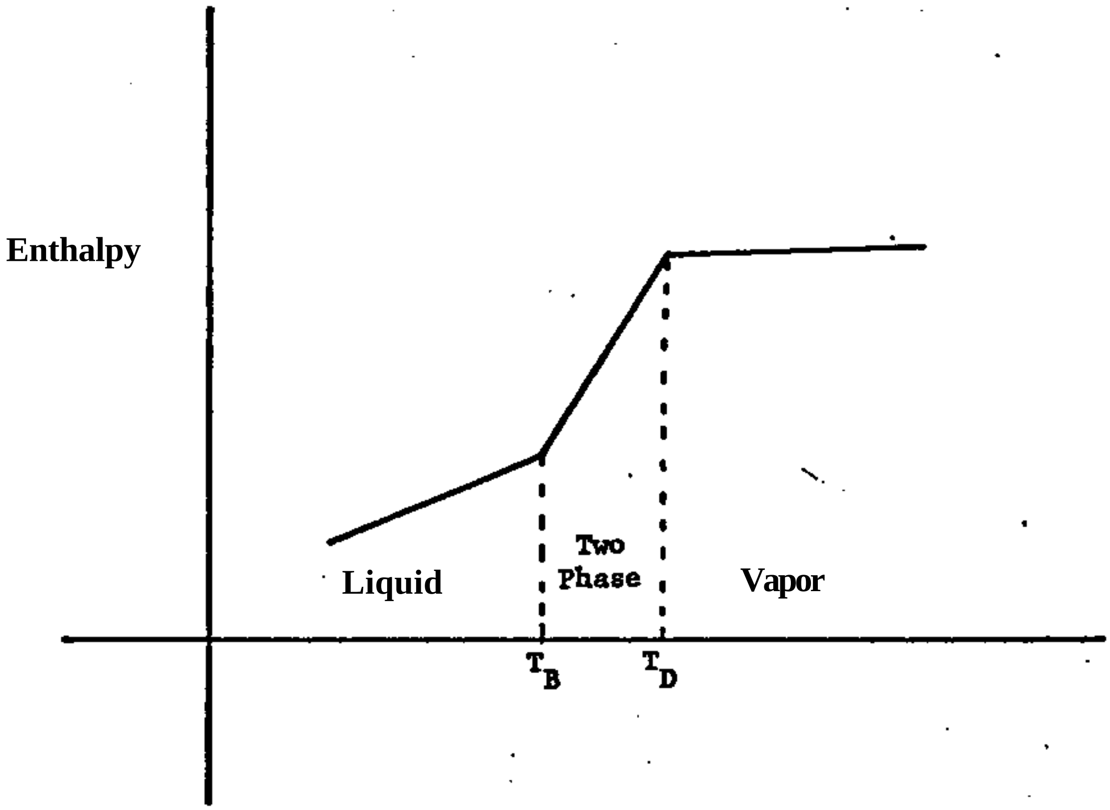
T_6 = Temperature of the cold stream at a location where the hot stream temperature is T_{BPH}

T_7 = Temperature of the hot stream at a location where the cold stream temperature is T_{DPC}

T_8 = Temperature of the hot stream at a location where the cold stream temperature is T_{BPC}

0 - Minimum allowable approach temperature

Fig. 3. Partitioning a Heat-exchanger into Zones where Phase Changes Occur



T_B • Bubble point temperature

T_D • Dew point temperature

Fig. 4. A Typical Cooling Curve

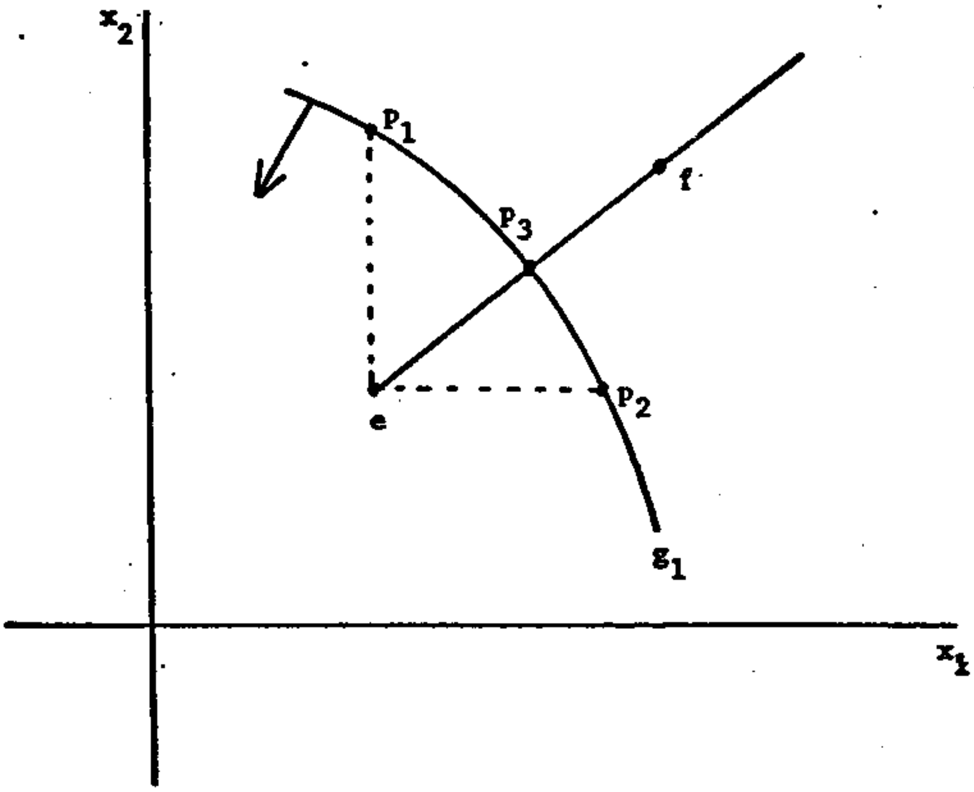
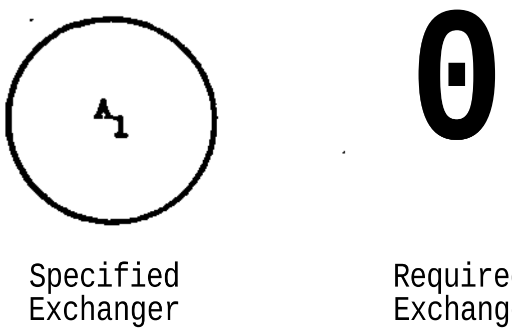


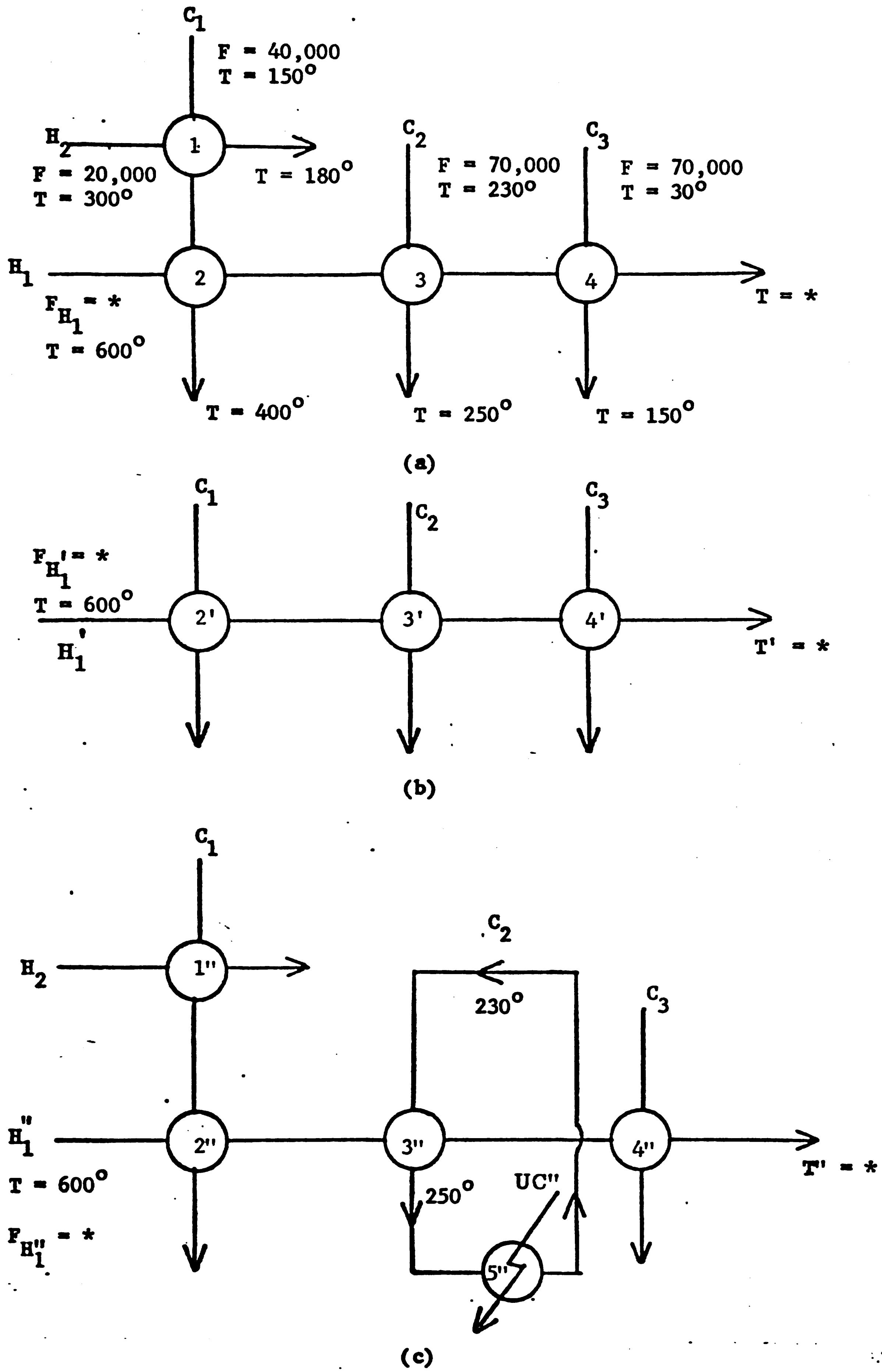
Fig. 5. Keeping Track of the Best Point 'e'



1) If $A_2 \le A_1$ Cost = 0

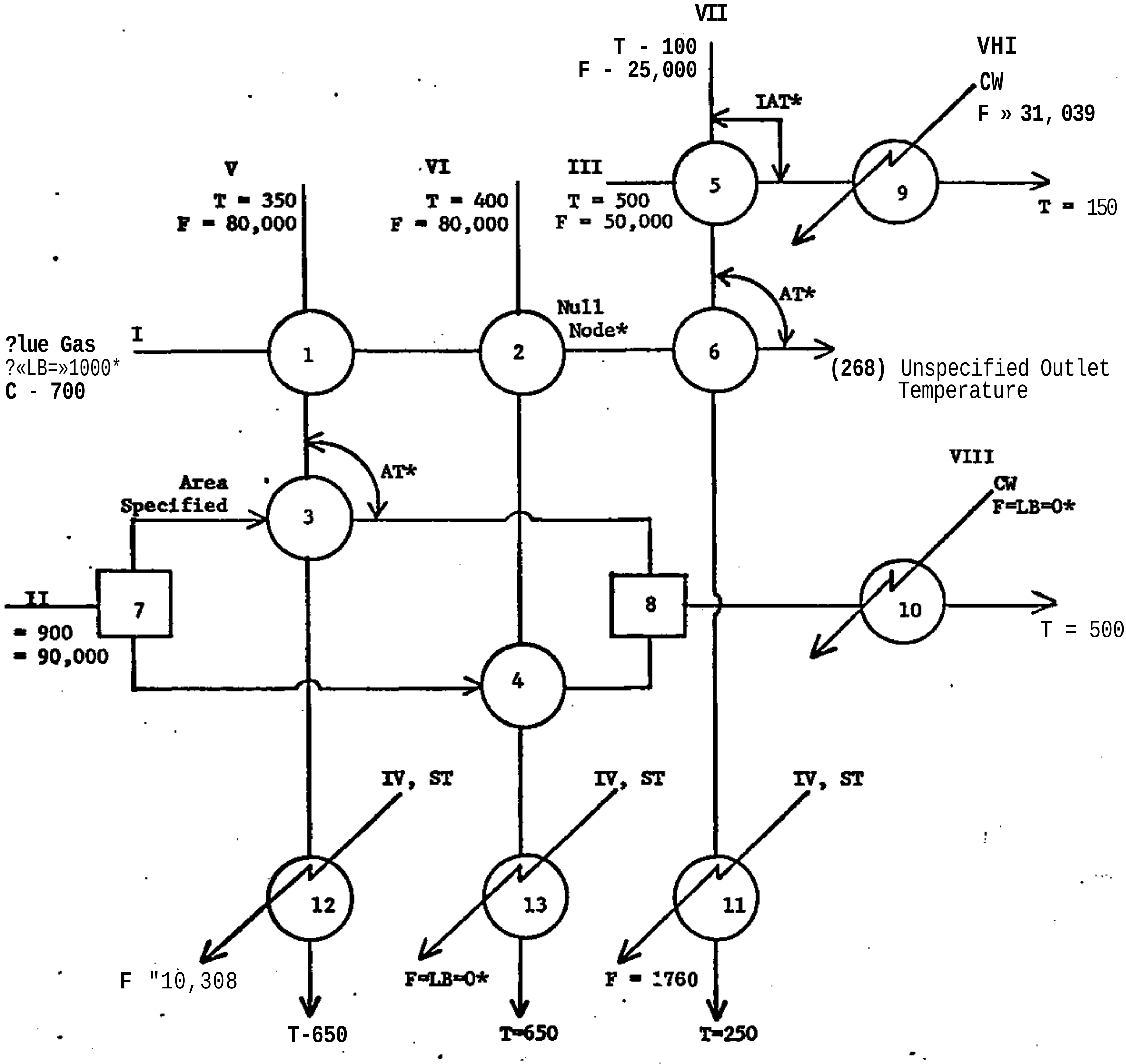
2) If $A_2 > A_1$ Cost = Cost associated with an exchanger having the area $(A_2 - A_1)$

Fig. 6. Using an Existing Exchanger



* Unspecified

Fig. 7. Reliability Analysis



ST - STEAM, CW - COOLING WATER

LB - LOWER BOUND

AT - MIN. ALLOWABLE APPROACH TEMP.

IAT - INTERNALLY HELD MIN. APPROACH TEMP.

* - ACTIVE' CONSTRAINTS AT OPTIMUM

Fig. 8. An Example Exchanger Network

TABLE 1

CONSTRAINT REPRESENTATION FOR AN EXAMPLE NODE 3 IN FIGURE 4

Comment	Exterior Constraints	Code
approach	T_1 ^ T_4 + D	(NODE * 10) + 1 = 31
approach		(NODE * 10) + 2 = 32
	$T_1 \geq T_2$	(NODE * 10) + 3 = 33
	V T 3	(NODE * 10) + A= 34
F_{1LB} = lower bound for flow F_1	*T^m	(NODE * 10) + 5 = 35
F_{1UB} = upper bound for flow F_1	$F_{1UB} \geq F_1$	(NODE * 10) + 6 = 36
F_{3LB} = lower bound for flow F	$F_3 \geq F_{3LB}$	(NODE * 10) + 7 = 37
F_{3UB} = upper bound for flow F	$F_{3UB} \geq F_3$	(NODE * 10) + 8 = 38

Type	Interior Constraints	Representation
approach	o j3.rO	(NODE * 1000) + 1 = 3001
approach	T_{BPH}^ T_6 + D	(NODE * 1000) + 2 = 2002
approach	T_7 * ^ pc + D	(NODE * 1000) + 3 = 3003
approach	T_{DPH} ^ T_5 + D	(NODE * 1000) + A = 3004

In addition there could be constraints associated with any stream segment.

Example: Segment 3, Node 3

Comment	Constraints	Code
H_{3LB} = lower bound for enthalpy H_3	J 3 T_{LB}	- (SEG * 1000 + NODE * 10 + 1) = -3031
H_{3UB} = upper bound for enthalpy H_3	$H_{3UB} \geq H_3$	- (SEG * 1000 + NODE * 10 + 2) = -3032

TABLE 2

THE INCIDENCE MATRIX

Variables / Equations	F 1	F F	F_2, F_4	F_3, F_5	h_{12}	h_4	h_5	h_{15}	F_6	α
(i)			X							X
(2)				X						X
(7)			X		X	X				
(10)				X			X	X		
(11)			X	X					X	
(12)			X	X		X	X		X	
(15)	X				X					
(18)		X						X		
(19)	X									
(23)							X			

TABLE 3

SOLUTION PROCEDURE FOR THE PROBLEM IN FIGURE 2

Decision Variables	σ_{55}, σ_{33}
Variable	Equation
i. $y=F_8)$	(19)
2. h ^	(15)
3. $h_\$$	(23)
4-. Guess $F_2(=F_4)$	
5. a	(1)
6. $F_3(=F_5)$	(2)
7. h^	(7)
8. F_6	(11)
9. $F_2(=F$^)	(12)
10. H_{15}	(10)
11. $F_9(=F_{10})$	(18)

TABLE 4

ADDITIONAL USER SPECIFICATION FOR THE PROBLEM IN FIGURE 4

Stream	H_1	H_1'	H_1''	UC"
Flow Rate	*	*	*	*
Lower Bound on Flow	50,000	50,000	50,000	0.
Upper Bound on Flow	200,000	200,000	200,000	100,000
Inlet Temp	600°	600°	600°	100.
Outlet Temp	*	*	*	150.
Lower Bound on Outlet Temp	190°	190°	190°	150. *
Upper Bound on Outlet Temp	600°	600°	600°	150.
Cost Coefficient	0.10	0.015	0.015	0.008

Unspecified.

Let U_i represent the heat transfer coefficient for exchanger i

$U^\wedge = U_x'' = 700$

$U_2 = U_2' = U_2'' = 477.27$

$U_3 = U_3« = U_3'' = 562.5$

$U_4 = U_4' = U_4« = 700$

$U_5« = 225$

TABLE 5

RESULTS FOR THE PROBLEM IN FIGURE 9

Exchanger	Area
1	131.50
2	30.01
3	413.24
4	103.24
2'	41.80
3*	462.31
4*	105.06
1"	131.50
2"	36.10
3"	0.00
4"	64.67
5"	0.00

$F_{H_1} = 170,341$ $F_{H_1'} = 180,878$ $F_{H_1''} = 50,148$

$T = T^* = T'' = 190°$

0 = 23451.25 SUI-

TABLE 6

STREAM SPECIFICATIONS FOR THE EXAMPLE IN FIGURE 9

DESCRIPTION	STREAM							
	I	II	III	IV	V	VI	VII	VIII
FLOW	*	90,000.	50,000.	*	80,000.	80,000	25,000.	*
INLET TEMPERATURE	100.0	900.0	500.0	756.0	350.	400.0	100.0	80.0
OUTLET TEMPERATURE	*	500.0	150.0	756.0	650.	650.0	250.0	130.0
INLET VAPOR FRACTION	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0
OUTLET VAPOR FRACTION	0.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0
LIQUID HEAT CAPACITY	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
VAPOR HEAT CAPACITY	1.0	1.0	*	1.0	1.0	1.0	1.0	1.0
HEAT OF VAPORIZATION	100.0	100.0	75.	768.0	100.0	100.0	200.0	100.0
DEW POINT	800.0	400.0	500.0	756.0	700.0	900.0	250.0	400.0
BUBBLE POINT	800.0	200.0	250.0	756.0	700.0	900.0	250.0	400.0
LIQUID PHASE HTC	1500.0	300.0	300.0	1500.0	300.0	300.0	300.0	300.0
VAPOR PHASE HTC	1500.0	300.0	300.0	1500.0	300.0	300.0	300.0	300.0
TWO PHASE HTC	1500.0	300.0	300.0	1500.	300.0	300.0	300.0	300.0
COST COEFFICIENT	10.0	0.0	0.0	8.5	0.0	0.0	0.0	1.0
LOWER BOUND ON FLOW	1000.	100.0	50,000.	0	80,000.	80,000.	25,000.	0.0
UPPER BOUND ON FLOW	90,000.	90,000.	50,000.	90,000.	80,000.	80,000.	25,000.	90,000.

*Unspecified

HTC - Heat Transfer Coefficient

Minimum Allowable Approach Temperature = 18°

TABLE 7

RESULTS

Example	1	2	3	4	5	6	7*	8**	9	10
Process Streams	3	4	3	4	2	2	7	5	5	6
Utility Stream	2	1	2	3	2	3	0	3	3	5
Exchangers	4	4	5	7	5	5	9	13	13	14
Decision Variables	2	2	3	4	4	5	6	8	9	13
Iterations to Feasible Point	4	2	2	13	4	9	18	8	8	***
New Solution Procedures After Feasible Time (seconds)	0 0.00	0 0.00	3 0.22	0 0.00	0 0.00	6 3.39	4 1.88	7 7.32	4 0.14	12 11.08
Iterations After Feasible Time (seconds)	35 3.43	59 6.31	72 4.51	108 38.28	149 16.89	107 10.85	706 163.25	320 71.62	119 17.13	819 306
Total Time (seconds)	7.31	8.24	6.48	48.12	36.63	27.82	180.57	108.60	23.49	477

* From Takamatsu, Hashimoto and Ohno (1970).

** Example illustrated in Figure 8.

*** Feasible starting point provided as an input.