1977

Computer Aided Synthesis Of Steam And Power Plants For Chemical Complexes

Masatoshi Nishio

COMPUTER AIDED SYNTHESIS OF STEAM AND POWER PLANTS FOR CHEMICAL COMPLEXES

by

MASATOSHI NISHIO

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Faculty of Graduate Studies
The University of Western Ontario
London, Canada

April 1977

ABSTRACT

A steam and power plant for a chemical complex is a type of multi-product process. "Product" demands are given, in terms of steam demands at different pressure levels, and electric power demand. Because there is a large number of possible ways to meet the demands there arises a synthesis problem for a steam and power plant.

Two different synthesis cases, i.e., a "grass roots" case and an expansion case, are studied.

The Optimal synthesis problem with constraints, which is an expansion case, is formulated and solved as a multi-time period linear programming problem. A general computer executive system OPES (Optimal Expansion of Energy Systems) is developed to formulate the problem automatically.

The Optimal synthesis problem with no constraints, which is a grass roots case, is formulated and solved using a two-level approach where a linear programming problem is formulated at the upper level and a parameter optimization problem is formulated at the lower level. The direct substitution method is employed for the two-level coordination.

In the parameter optimization problem which includes a simulation of arbitrary energy systems, energy and material

balances are formulated into a set of simultaneous equations having a homogeneous form, systematically by a modular approach and solved by the triangulation method. The Complex method is chosen to seek a set of optimal parameter values.

General computer executive systems OSES (Optimal Synthesis of Energy Systems) and ODES (Optimal Design of Energy Systems) are developed to formulate and solve these problems automatically.

Practical examples are illustrated to show the effectiveness of the methods for both cases of synthesis, and the usefulness of computer executive systems that assist the generalization of the methodologies.

ACKNOWLEDGEMENTS

The author wishes to express his sincere thanks to Dr. A.I. Johnson, Dean and Professor, Faculty of Engineering Science, The University of Western Ontario, for his supervision and guidance in carrying out this study.

The author also owes many thanks to Dr. M.A. Bergougnou, Professor, Faculty of Engineering Science, The University of Western Ontario, for valuable comments on this study and especially his constant guidance in carrying out the design project required for the partial fulfillment of the Ph.D. Comprehensive Examination.

The author extends his appreciation to Mr. J.R. Dickinson, SACDA director, and Mr. S. Komatsu, Manager of Process Technology Groups, Chiyoda Chemical Engineering and Construction Company Limited, Japan for their critical and practical comments on this study.

Special thanks are due to Mr. H.M. Sochan, Manager, Energy Engineering and Conservation, Polysar and Mr. W.N. Brown, Utilities Technologist, Energy Engineering and Conservation, Polysar for their useful suggestions and co-operation which provided the necessary information for this work.

Also the author is indebted to Chiyoda Chemical Engineering and Construction Company Limited, Japan, for granting him a leave of absence to carry out this study.

Lastly, the author wishes to express his gratitude to Mrs. B. Yates for her excellent typing of this thesis.

TABLE OF CONTENTS

PAGE

INDEX OF TABLES

Appendix (contd.) PAGE

INDEX OF FIGURES

FIGURE	DESCRIPTION	PAGE

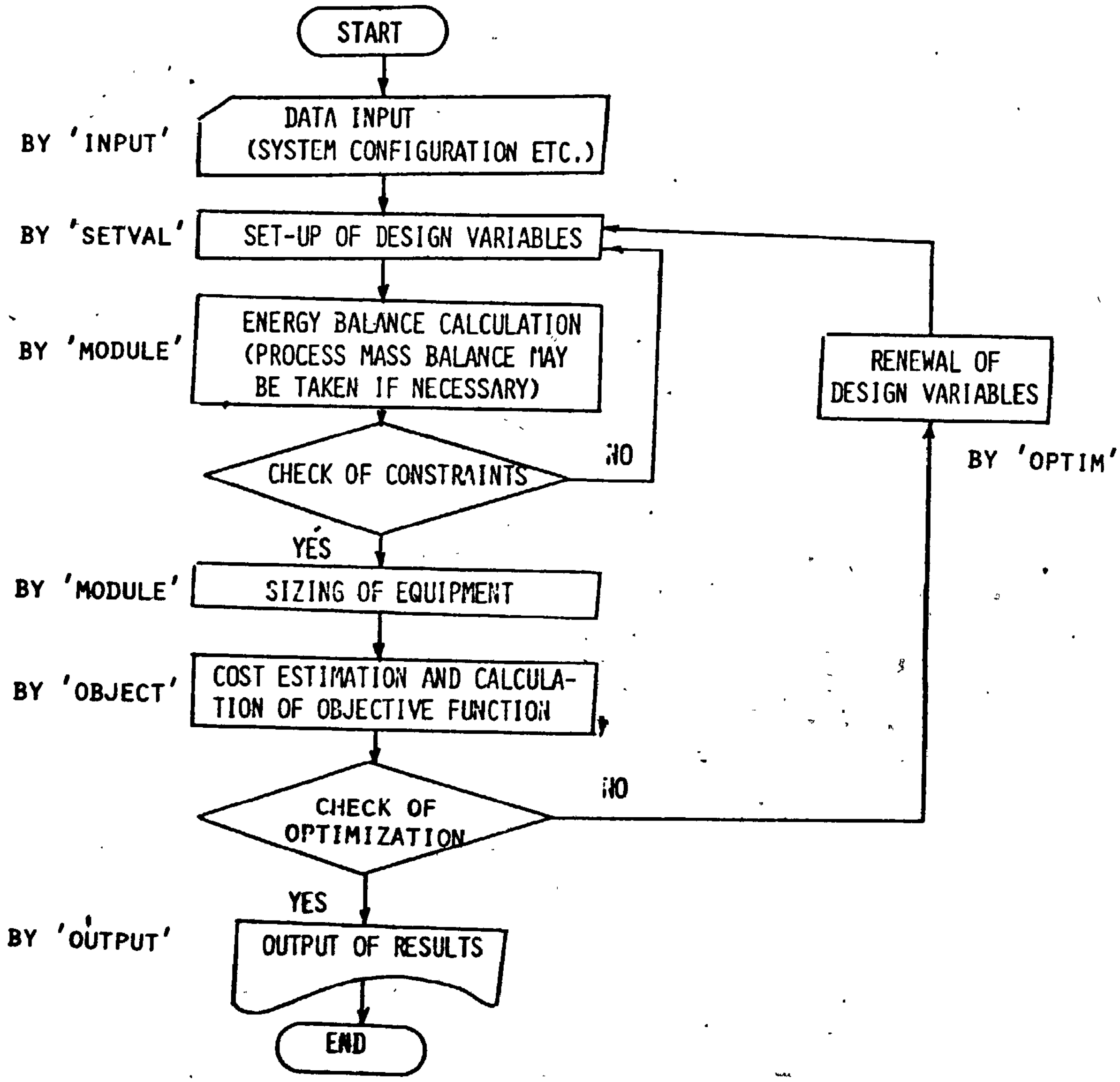


FIG. 2.4 COMPUTATIONAL PROCEDURE FOR AN OPTIMAL DESIGN OF ENERGY SYSTEMS

NOMENCLATURE

$\underline{\underline{A}}$	Coefficient matrix	
a or a_{ij}	Element of matrix $\underline{\underline{A}}$	
$\underline{b}$	Right hand side vector of simultaneous equations	
c_f	Unit conversion factor	(KW/BTU)
Cop	Operating expenses	($)
d	Depreciation rate for tax purposes	
e	Depreciation rate for accounting purposes	
$\underline{g}$	Lower bounds for independent variable	
$\underline{G}$	Lower bounds for implicit constraints	
h	Heating value of fuel	(BTU/lb)
I	Investment on facilities or stream identification number	($)
I_w	Working capital	($)
i	Enthalpy or rate of earnings of the company on its invested capital	(BTU/lb)
i_m	Minimum rate of return considered attractive	
J	Period expanded	
k or K	Number identifying a unit	
L	Period operated	
m	Number of variables including knowns	
M	Sequence number of equation or 10^3	
N	Number of unknowns or equations	
P	Net profit	($)
P_i	Blow-down ratio or power consumption ratio	

Q	Heat flow rate	(BTU/Hr)
R	Gross profit	($)
$\underline{s}$	Upper bounds for independent variables	
$\underline{S}$	Upper bounds for implicit constraints	
t	Income tax rate	
V	Venture profit or venture cost	($)
$\underline{X}$	State variables; mainly represents energy rate or independent variables	
$\underline{X}_o$	Centroid of a complex	
$\underline{X}_w$	Worst point in a complex	
W	Venture worth	
$W_{j,k}$	Demand to unit k required at period j	
$Y_{i,j,k}$	Supply provided at period j by $Z_{i,k}$ the capacity of unit k that is expanded at period i	
Z_k or $Z_{i,k}$	Capacity of unit k to be installed (at period i)	

<u>Greek letters</u>

α	Splitting ratio or reflection factor in the Complex method
β	Heat exchange ratio
γ	Pseudo-random number
η	Turbine or boiler efficiency
$\underline{\phi}$	Implicit constraints

CHAPTER 1

INTRODUCTION

1.1 General

Drastic changes in the energy situation throughout the world have necessitated a re-evaluation of existing technology and provided opportunities for innovations. The need for energy conservation is serious and various methods for energy saving have been tried, especially in those chemical industries that consume great amounts of energy.

A steam and power plant for a chemical complex is one of several systems which should be examined carefully from the above point of view; for steam and power plants not only supply great amounts of energy with different qualities to a chemical complex, but also consume large amounts of energy resources themselves in order to produce "finished" products.

The synthesis of steam and power plants is an important task in industries which need various types of energy. Many alternate sizes, types, and arrangements of energy supply and conversion devices can be conceived for a steam and power plant to serve a complex chemical process; it is thus important to create the best possible arrangement.

In this chapter, design studies for steam and power plants are reviewed for the purpose of identifying problem areas. System synthesis problems are reviewed to search for synthesis techniques that will be effective to synthesize a steam and power plant systematically. The scope of study will be defined in accordance with the works reviewed.

1.2 Review of the Design of Steam and Power Plants

Although a steam and power plant consists of many components, this review is confined to system design problems instead of investigating each component itself in detail. On the other hand, the review on constrained system design problems, i.e., expansion problems for steam and power plants due to growth of demands, is extended to other fields, e.g., chemical process and public power station because of some similarities with respect to expansion problems. The particular situation that motivated the study of expansion problems is the expansion of production of the Polysar plant as it seeks to attain world scale production to become competitive. A plant like Polysar, much of it over thirty years old, can begin to take advantage of new technology both in the supply and in the demand side, for steam and power.

Slack (1969) introduced a new exact method of steam balance by setting up simultaneous equations and Bouilloud (1969) succeeded it by LP solving, however, their approaches were rather specific in the problem formulation.

Kikkawa and Shoji (1968) presented an "optimal design package" dealing only with typical utility systems in refining industries. Also, Dain and Whitlock (1969) introduced a "computer program" for the optimization of total energy systems which are limited to specific configurations. Yamazaki (1970) applied a linear programming technique first and thereafter used separable programming for determining the capacity of a power plant; his approach seems to be of limited use because the effect of design parameters was not taken into account on determining the capacity. Hatakeyama and Symazu (1971) developed a general method for heat balance calculations; their method was only for on-line use on computer control of power stations with heat generation.

There is only one paper (Jackson et al, 1965) which deals with expansion problems for a steam and power plant using stochastic simulations; the best method for dealing with expansion problems remains open to question.

While there are many similarities between the expansion problems for public power stations and the expansion of the energy system for a chemical complex, there is one great difference between them; that is the problem of a determination of installed reserve margin of plant capacity. For public power generation, this is a very significant problem mainly because of scale of investment and also because of the quality of service required. Therefore, much work has been done for the rational determination of generation reserve using probability concepts (Kist and Thomas, 1958; Baldwin et al, 1959) and a criterion of economic choice (Cohen and Jensen, 1958; Schroeder and Wilson, 1958, and so on).

Thus, studies of expansion problems for public power generation have been done frequently but these are not satisfactory for practical full-scale industrial application to a steam and power plant for a chemical complex.

Of the mathematical techniques that make a selection of optimal expansion patterns possible, while all practical candidates of expansion pattern were case-studied at early periods of study in this field, only the dynamic programming technique has been applied thus far to a chemical process (Jeneroso, 1968) and to a public power station (Booth, 1972). However, the dynamic programming technique which is suitable for a serial process can not be used for a steam and power plant.

1.3 Review of System Synthesis Problems

The general techniques (Hendry et. al, 1973) that have been developed for process synthesis have included the heuristic approach based on the use of rules of thumb, algorithmic methods often involving well-known optimization principles, and evolutionary strategies wherein improvements are systematically made to an initially proposed feasible structure. Often these techniques have been used in combination and in conjunction with the decomposition principle.

Masso and Rudd (1969) indicated a means by which the design decomposition approach to process synthesis could be modified to make use of heuristic problem-solving techniques. A heuristic method is one which seeks the discovery of the solution to a problem by means of plausible but fallible guesses.

Heuristic rules of thumb are quite common in chemical engineering practice, examples being the six-tenths-power-law cost approximation, minimum approach temperature in heat exchange, estimates of optimal reflux ratios, and economic fluid velocities. The useful heuristics are well stated by King (1971).

Rudd (1968) proposed an approach to process system synthesis based on decomposition whereby a design problem for which no previous technology existed is broken down into a sequence of sub-design problem until the level of available technology is reached. This approach provides a framework for the systematic synthesis of solutions for quite general design problems. Umeda et. al, (1974) applied this approach to complex processing systems by an extensive use of the task assignment concept.

Ichikawa and co-workers have attempted to apply well-known techniques of optimization and mathematical programming to the synthesis of chemical processes. The approach has been to embed all possible process flow sheets into one combined flow sheet by defining all the inter-connections which might exist between various pieces of equipment. The approach has also been applied to portions of the general chemical process synthesis problem such as heat exchange networks and sequences of component separation equipment. These techniques include linear programming, dynamic programming, non-linear programming, branch-and-bound, etc.

Evolutionary synthesis refers to the synthesis of a new process by modification of previously generated processes. King, Ganz and Barnes (1972) applied this technique as a succession of alternations involving identification of that portion of the most recent process which could be changed to greatest advantage, followed by generation of the appropriate change for that portion of the process and by an analysis of the new process. It should be pointed out that these synthesis procedures are necessarily local in that effect and that processes synthesized by this means depend heavily upon the initial assumed processing concept.

Papers dealing with techniques of process synthesis are classified and summarized in Table 1.1, while papers which include applications of process synthesis techniques to specific chemical processes are classified and summarized in Table 1.2.

TABLE 1.1

PROCESS SYNTHESIS TECHNIQUES

Technique	References
A. Decomposition	Rudd (1968), Nishida et.al. (1971), Kobayashi et. al. (1971) Menzies and Johnson (1972), Umeda et.al. (1974), Umeda and Ichikawa (1975).
B. Heuristics	Lockhart (1947), Herbert (1957), Masso and Rudd (1969), Nishida et. al. (1971), Nishimura and Hiraizumi (1971), King (1971). Siirola et. al. (1971), Siirola and Rudd (1971), Menzies and Johnson (1972), Powers (1972), Thompson and King (1972), Umeda et. al. (1974), Mahalec et. al. (1977), Liapias et. al.(1977), Wells and Hodgkinson (1977).
C. Algorithmic (optimization)	Rod and Marek (1959), Hwa (1965) Kesler and Parker (1969), Lee et.al. (1970), Kobayashi et.al. (1971), Menzies and Johnson (1972), Umeda et.al. (1972), Goto and Matsubara (1972), Hendry and Hughes (1972), Umeda and Ichikawa (1972), Thompson and King (1972), Rathore et.al. (1974), Rathore and Powers (1975), Westerberg and Stephanopoulas (1975), Nishida et.al. (1976,1977) Umeda et.al. (1974), Umeda and Ichikawa (1975), Mahalec et.al. (1977).
D. Evolutionary	King et.al. (1972), McGalliard and Westerberg (1972), Ichikawa and Fan (1972), Nishida et.al. (1977), Mahalec et.al. (1977).

TABLE 1.2

APPLICATIONS OF SYNTHESIS TECHNIQUES

TO CHEMICAL PROCESSES

	Structures	References
A.	Homogeneous: Heat-exchanger networks	Hwa (1965), Kesler and Parker (1969), Masso and Rudd (1969), Lee et.al. (1970), Kobayashi et.al. (1971), Nishida et.al. (1971), McGalliard and Westerberg (1972), Ponton and Donaldson (1974), Rathmore and Powers (1975), Nishida et.al. (1977), Liapins et.al.(1977), Wells and Hodgkinson (1977).
	Multicomponent distillation separation sequences	Lockhard (1947), Harbert (1957), Rod and Marek (1959), Petlyuk et.al, (1965), Nishimura and Hiraizume (1971), King (1971), Hendry and Hughes (1972), Westerberg and Stephanopoulos (1975).
	Reactor Networks	Ichikawa and Fan (1972), Umeda and Ichikawa (1972).
B.	Heterogeneous: Energy-transfer networks	King et.al. (1972), Menzies and Johnson (1972).
	Selection and sequencing of separation processes	Siirola and Rudd (1971), Thompson and King (1972), Powers (1972), Rathore et.al. (1974).
	Entire chemical processes	Siirola et.al. (1971), Siirola and Rudd (1971), Umeda et.al. (1972), Powers (1972), Ichikawa and Fan (1972), Goto and Matsabara (1972), Rudd et.al. (1973), Nishida (1976), Umeda et.al. (1974), Umeda and Ichikawa (1975).

1.4 Scope of Study

As a result of the work reviewed, the following points have become clear:

- There are two different types of synthesis problems. Namely, one is the synthesis problem with constraints; that is, a plant expansion problem. The other is the synthesis problem with no constraints; that is, a "grass roots" plant erection problem. No extensive study has been done on both synthesis problems for steam and power plants.

- A solution method of energy and material balances for arbitrary energy systems will have to be developed to support an extensive study on synthesis problems of steam and power plants.

- There have been few successful applications for the synthesis of a large system with a heterogeneous structure.

- Expansion problems of steam and power plants for process industries are quite different from those of public power stations.

Accordingly in this study, two different synthesis problems of steam and power plants for a chemical complex, which is a quite large system with a heterogeneous structure, will be defined and formulated. A solution method for energy and material balances for arbitrary energy systems will be developed as a matter of necessity. These methods will be applied to practical examples and solved.

Moreover, the methodologies are generalized by the development of the following three computer executive systems:

ODES	Optimal Design of Arbitrary Energy Systems
OPES	Optimal Planning for Expansion of Energy Systems
OSES	Optimal Synthesis of Energy Systems

CHAPTER 2

COMPUTER EXECUTIVE SYSTEM FOR OPTIMAL DESIGN OF ARBITRARY ENERGY SYSTEMS

2.1 Introduction

As pointed out by Fleming et. al. (1974) the recent drastic increase of energy costs has changed the design philosophy which had been followed before the "energy crisis". Since the energy crisis engineers have been obliged to take a completely new look at plants which consume great amounts of energy. This new situation has created the need for the development of a general computer executive system which makes possible a prompt evaluation for arbitrary energy systems.

Figure 2.1 is shown as an example of an energy system of reasonable complexity. Such systems comprise many component units and in the analysis or design of such systems a very large amount of information is involved. This chapter presents the first of a series of executive programs developed in this study for handling large information sets and for easily evaluating arbitrary energy systems.

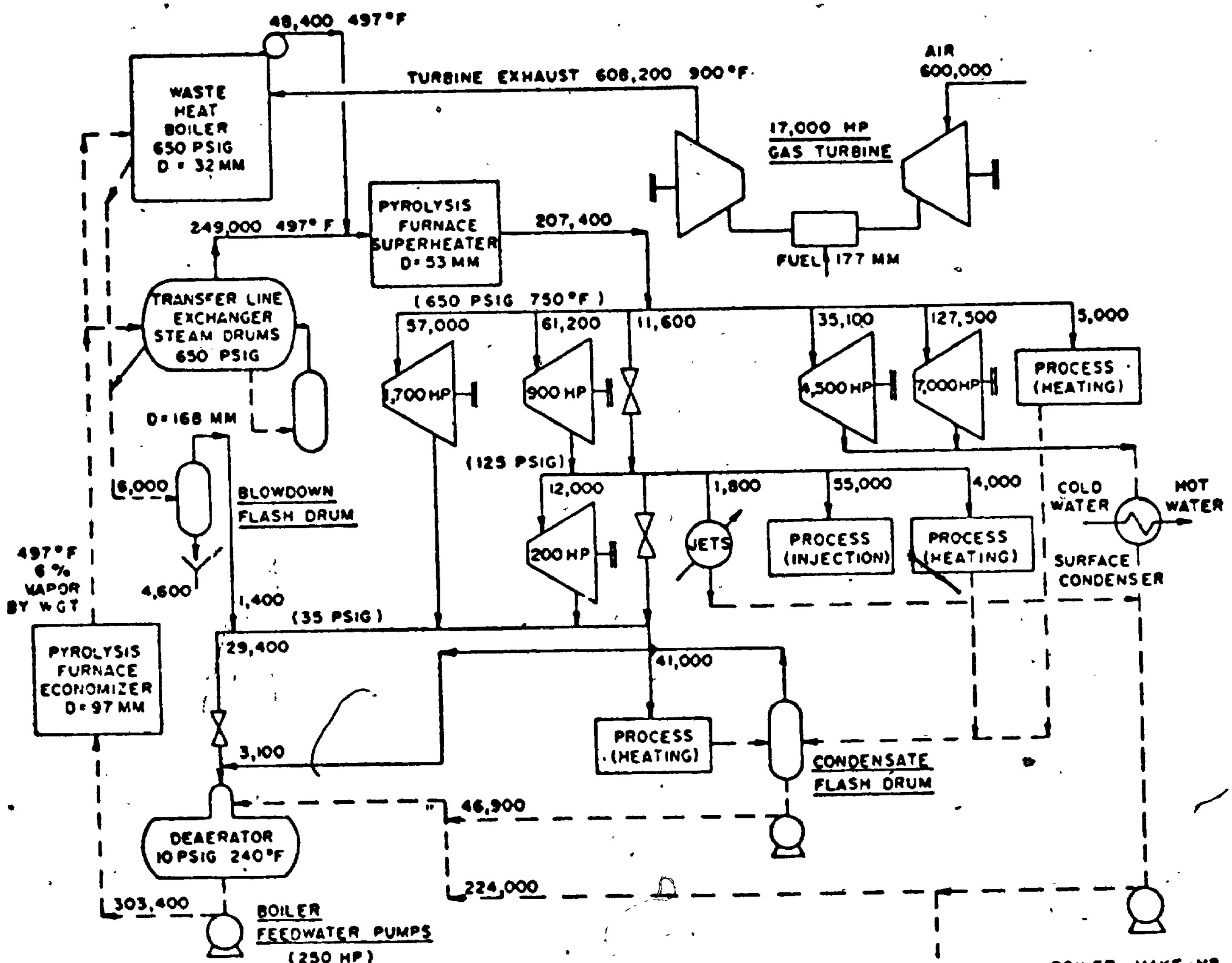


FIGURE 2.1 AN EXAMPLE OF ENERGY SYSTEMS

NOTE: Unit for all material flowrates is lb/hr.

First, a new solution method is developed for the energy and material balance calculation which is a central part in evaluating arbitrary energy systems. Second, a unit modeling method is introduced which uses a modular approach to make an evaluation of arbitrary energy systems easy. Third, an optimization technique is chosen which makes an optimal design attainable. Fourth and finally, the program system and data structureswhich support the generality of the executive system are defined.

2.2 Problem Description

In general the usefulness of a computer executive system depends upon the following three aspects:

- Input-Output (I/O) methodology
- Program system and data structures
- Computational efficiency and stability

The I/O device must be simple for "users" to access. At the same time it should provide an interface with that of the.other important components of the synthesis system; "OPES" (which will be developed in Chapter 3) and "OSES" (which will be developed in Chapter 4).

The data structure should be succinct enough from the viewpoint of programming and achieve efficient storage of data.

The computational efficiency depends largely upon the computing method used for the central part of the simulation. Therefore, an efficient method has to be developed for the energy and material balance calculation.

The choice of the optimization technique is also important from the viewpoint of computational stability since many optimization techniques fail to attain an optimal point for a problem with many independent variables and stiff constraints; such as is usually the case for optimal design of energy systems.

2.3 Solution Method for Energy and Material Balances

Basically there are two methods for solving energy and material balances i.e. sequential and simultaneous methods (Umeda and Nishio, 1972).

Energy systems characteristically have a large number of recycle loops and also a very large number of energy "products" which are specified as system outputs. For these characteristics, a sequential method which is often used in a balance calculation for chemical processes is not advantageous. A sequential method starts the calculation from the system input and proceeds toward the system output sequentially and is effective only when the system specification is input-oriented and the system does not have a large number of recycle loops. Since a simultaneous method is not limited in these respects, it is chosen as the solution method for energy and material balance.

The modular approach in which there are equipment modules, each with a keyword identifier and information streams, is used to represent the systems of energy and material balance equations that can describe the systems. As will be explained in the following section, energy and material balance equations can be set up by a modular approach in an executive system which eventually assembles them into a set of simultaneous linear equations such that:

$$\underline{X}' \underline{\underline{A}} = \underline{0}' \qquad (2.1)$$

where

$\underline{X}' = (x_1, \ldots\ldots, x_m)$

$\underline{0}$ is the zero vector with n zero elements

and $\underline{\underline{A}}$ is a coefficient matrix; assumed non-singular:

$$\underline{\underline{A}} = \begin{pmatrix} a_{11} & \cdots & a_{1k} & \cdots a_{1n} \\ a_{i1} & \cdots & a_{ik} & a_{in} \\ a_{m1} & & a_{mk} & a_{mn} \end{pmatrix} .$$

The variables and coefficients will be explained when examples of unit modeling are presented in the next section.

These homogeneous linear equations (2.1) are converted into equations (2.2) by triangulation of matrix $\underline{\underline{A}}$ as follows:

$$\underline{X}' \quad \underline{\underline{A}} = \underline{0}'$$

$$\underline{X}' \underline{\underline{B}}\,\underline{\underline{C}} = \underline{0}'$$

$$\underline{X}' \underline{\underline{B}}\,\underline{\underline{C}}\,\underline{\underline{C}}^{-1} = \underline{0}'\underline{C}^{-1}$$

$$\underline{X}' \quad \underline{\underline{B}} = \underline{0}' \qquad (2.2)$$

where

$$\underline{\underline{B}} = \begin{pmatrix} b_{11} & 0 & \cdots & 0 \\ \vdots & & & \vdots \\ b_{i1} & & & \vdots \\ \vdots & & & 0 \\ b_{n1} & \cdots b_{nj} & \cdots & b_{nn} \\ \vdots & & & \vdots \\ b_{m1} & \cdots b_{mj} & \cdots & b_{mn} \end{pmatrix}$$

$$\underline{\underline{C}} = \begin{pmatrix} 1 & C_{12} \cdots C_{1k} \cdots & C_{1n} \\ 0 & 1. & \vdots \\ \vdots & \ddots & C_{jn} \\ \vdots & \ddots & \vdots \\ 0 & \cdots\cdots 0 & 1 \end{pmatrix}$$

where elements of $\underline{\underline{B}}$ and $\underline{\underline{C}}$ can be obtained as follows:

$$\left.\begin{aligned} b_{i,k} &= a_{i,k} - \sum_{j=1}^{k-1} b_{i,j} c_{j,k} \quad , \quad (i=k,..m) \\ c_{k,i} &= (a_{k,i} - \sum_{j=1}^{k-1} b_{k,j} c_{j,i}) / b_{k,k} \quad , \quad (i=k+1,..,n) \end{aligned}\right\} (k=1,..n) \qquad (2.3)$$

Equations (2.2) can be solved recursively as follows:

$$x_{n-k+1} = -\sum_{j=n-k+2}^{m} x_j b_{j,n-k+1} / b_{n-k+1,n-k+1} \; , \quad (k=1,.,n) \qquad (2.4)$$

The calculation of energy and material balances occupies a significant part of the computation in the simulation of energy systems and the coefficient matrix in the simultaneous equation set which represents energy and materials balances, is usually sparse as can be seen in an example in Appendix 3. Therefore, the essential elements for the calculation of energy and material balances are extracted and stored for iterative calculation such as optimization, to gain computational efficiency. This can be achieved by decomposing the solution procedure (2.3) and (2.4) into the computation sequence of four rules of arithmetic (Hiraizumi et.al. 1969), which will be described in detail in Appendix 5.

Hatakeyama and Symazu (1971) formulated heat balance equations into simultaneous equations with heterogeneous form such that:

$$\underline{\underline{A}}\ \underline{x} = \underline{b} \tag{2.5}$$

where $\underline{A}$ is a square matrix different from that of equations (2.1).

The Gauss elimination method was used to solve them. This formulation could be extended easily to the general calculation of energy and material balances. In addition the sparse-matrix technique proposed by Bending and Hutchison (1973) could be applied for iterative calculation from the standpoint of computational efficiency. However, it seems that the formulation by a homogeneous form is easier and more consistent than that by a heterogeneous form for energy systems which include a large number of specified variables, i.e. "energy" products; because in the latter approach there is a disadvantage that known variables must come to the right hand side or extra equations have to be defined for known variables.

A comparison of balance formulation by both forms can be found in Appendix 3.

2.4 Unit Modeling by a Modular Approach

As mentioned in the preceding section, if a system configuration is represented using a set of keywords which have "equipment images", and stream numbers which indicate input or output information variables relating to the "equipment", energy and material balances can be formulated systematically into a set of simultaneous linear equations by a modular approach.

A typical example of unit modeling can be used to illustrate the modular approach. Figure 2.2 shows a symbolic figure for a back-pressure turbine unit. From this figure the following equations are derived:

$$x_1 - x_2 = 0 \tag{2.6}$$

$$x_3 - c_f (i_1 - i_2)\, \eta\, x_1 = 0 \tag{2.7}$$

where

c_f ; unit conversion factor

i_1 ; enthalpy at turbine inlet (BTU/lb)

i_2 ; enthalpy after isentropic expansion (BTU/lb)

η ; turbine efficiency which may be a function of steam rate, pressure or temperature

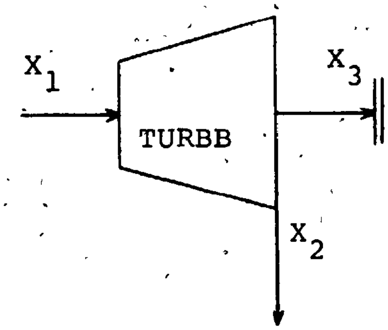


Figure 2-2 Symbolic Figure for a Back-Pressure Turbine Unit

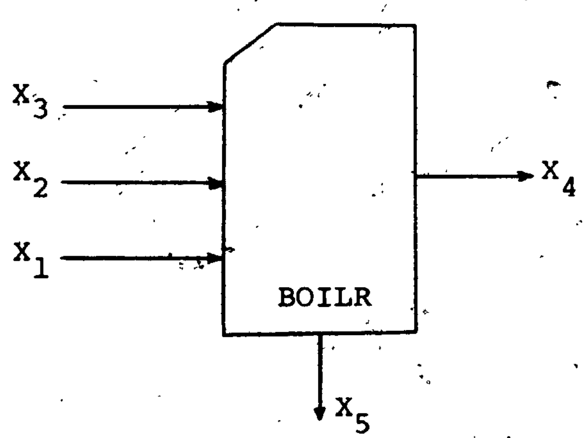


Figure 2-3 Symbolic Figure for a Boiler Unit

x_1, x_2 ; steam rates (lb/hr)

x_3 ; motive power rate generated (KWH)

The above equations are coded actually using data structures defined in the following section. For example, the computation of enthalpy at turbine inlet is accomplished as follows:

```
CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
```

where

ENTH is a subprogram to compute an enthalpy for steam using regression models of the "steam table". The first argument 3 indicates the code for vapor enthalpy. SNEN(IS(I1,2)+2) indicates the temperature at turbine inlet. As will be described in the following section, IS(I1,2) points to the head address where stream data are stored. Similarly, SNEN(IS(I1,2)+3) implies the pressure at turbine inlet. E1 is the enthalpy to be computed by subprogram ENTH.

The complete module for this unit is listed in Listing 2.1. Another example module is for a boiler unit as follows. From the symbolic figure for a boiler unit given in Figure 2.3, the following equations are derived:

LISTING 2.1 MODULE FOR BACK-PRESSURE TURBINE UNIT

```
      SUBROUTINE TYPE11(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** BACK TURBINE(TURBB)
C     I1                      ... STREAM NO. OF INLET STEAM
C     I2                      ... STREAM NO. OF OUTLET STEAM
C     I3                      ... STREAM NO. OF OUTPUT POWER
C     E1                      ... INLET ENTHALPY
C     E2                      ... OUTLET ENTHALPY AFTER ISENTROPIC EXPANSION
C     A                       ... COEFFICIENT MATRIX FOR A SIMULTANEOUS EQUATION SET
C     L-3                     ... SIZING AND COST ESTIMATION
C     L-2                     ... ENERGY AND MASS BALANCE
C     N                       ... SEQUENCE NUMBER OF BALANCE EQUATIONS
C     IUNIT-1                 ... C.G.S UNIT
C     IUNIT-2                 ... BTU-LB UNIT
C     SNEN(IS(I1,2)+2)        ... TEMPERATURE OF INLET STEAM
C     SNEN(IS(I1,2)+3)        ... PRESSURE    OF INLET STEAM
C     ENTS(SUBROUTINE)        ... CALCULATE A CONDITION AFTER ISENTROPIC EXPANSION
C     ENTH(SUBROUTINE)        ... CALCULATE STEAM ENTHALPY
C     EFFTB(FUNCTION)        .... TURBINE EFFICIENCY
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
C     STEAM BALANCE
      N=N+1
      A(I1,N)= 1.
      A(I2,N)=-1.
C     MODEL FOR POWER GENERATED BY STEAM EXPANSION
      N=N+1
      A(I3,N)= 1.
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1          T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      A(I1,N)=-(E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1          SNEN(IS(I1,2)+3))
C     UNIT CONVERSION
      IF(IUNIT.EQ.1) A(I1,N)=A(I1,N)*3.968
      A(I1,N)=A(I1,N)*0.293
      IF(A(I1,N).GT.0.) ICH=-1
      RETURN
C     COSTIMATION FOR TURBINE
10    CALL COST11(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

$$x_4 - (1 - p_1)\ x_1 = 0 \tag{2.8}$$

$$x_5 - p_1 x_1 = 0 \tag{2.9}$$

$$i_5 x_5 + i_4 x_4 - i_1 x_1 - \eta_1 h x_3 = 0 \tag{2.10}$$

$$x_2 - p_2 x_1 = 0 \tag{2.11}$$

where

x_1 ; boiler feed water rate (lb/hr)

x_2 ; power consumed for a draft fan driver (KWH)

x_3 ; fuel rate consumed (lb/hr)

x_4 ; steam rate generated (lb/hr)

x_5 ; blow-down rate (lb/hr)

p_1 ; blow-down ratio

p_2 ; power consumption ratio (kw/hr)

η_1 ; boiler efficiency

h ; heating value of fuel (BTU/lb)

Other unit modules can be built-in a similar way; program listings for other unit modules and physical property modules are attached in Appendix 9.

2.5 Choice of an Optimization Technique

The choice of an optimization technique is a key to the success of an optimal design by a computer executive system, for arbitrary energy systems may have many independent variables and stiff inequality constraints. Generally an optimal design problem for arbitrary energy systems can be regarded as a non-linear optimization problem which has no unique characteristics of which one can take advantage in choosing a particular method such as mathematical programming, variational methods and so on. Therefore, a search method seems to be suitable for this type of problem. The Complex method (Box, 1965), which is the extension of the Simplex method (Nelder and Mead, 1965) to the problem with inequality constraints, is chosen as an optimization technique. This method is extremely suitable from the viewpoint of computational stability for attaining an optimal point, whereas many other search techniques fail when the problem has many independent variables and inequality constraints. This has been confirmed through numerical experiments by Lopez (1975) and by the author's own experiences. The description as to implementation of the Complex method can be found in Appendix 6.

2.6 Program System and Data Structures

A general computational procedure for an optimal design of energy systems can be represented as shown in Figure 2.4. The program system and data structures

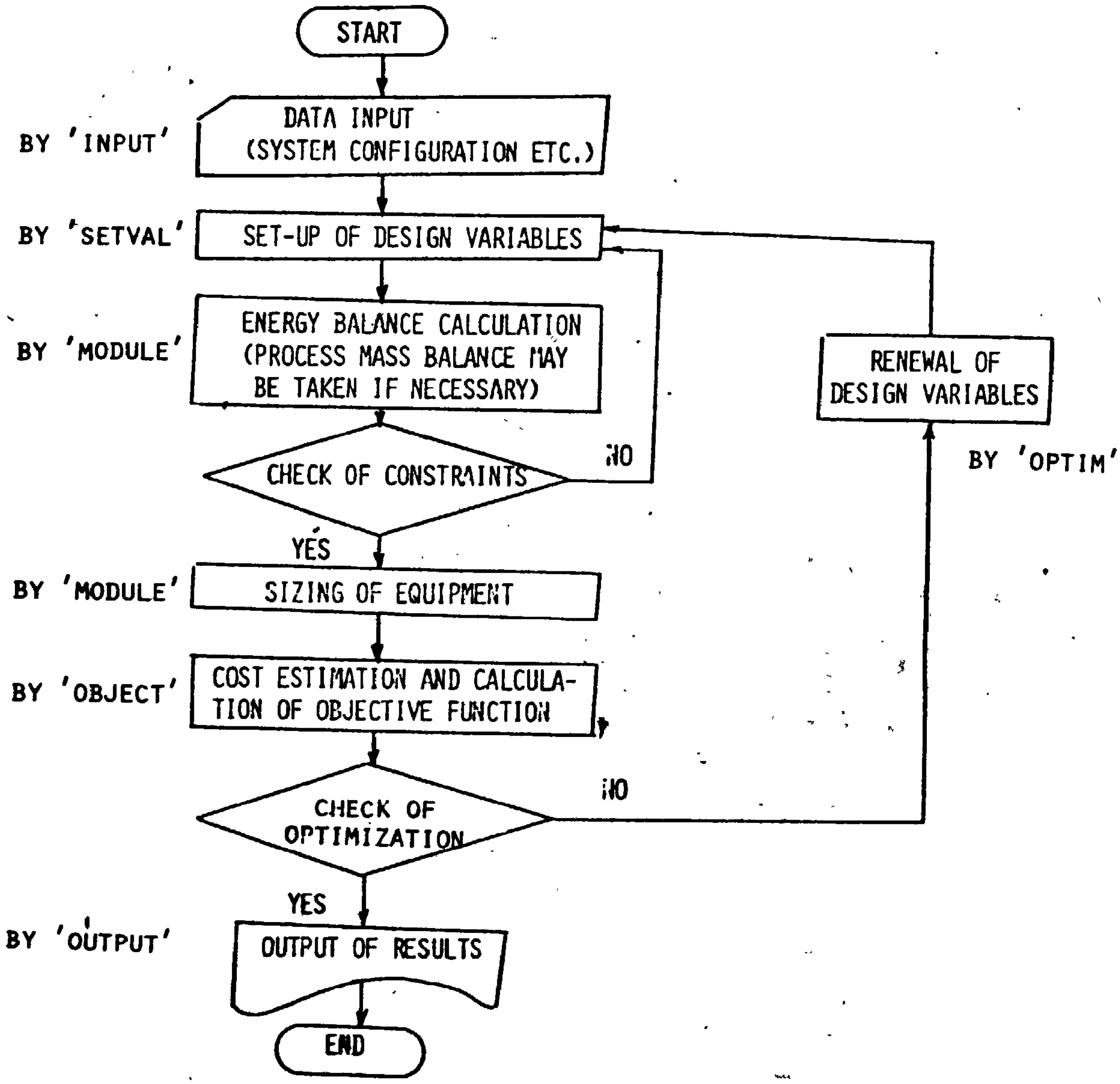


FIG. 2.4 COMPUTATIONAL PROCEDURE FOR AN OPTIMAL DESIGN OF ENERGY SYSTEMS

are designed for the generalization of the computer executive system which implements the computational procedure shown in Figure 2.4.

Figure 2.5 indicates the program structure and the main components of "ODES". This program structure serves not only to make the organization of functions involved in an evaluation for energy systems clear, but also to keep future expansion of the program system open. The groups of subprograms are:

1) The Design and Optimizing Group

- OPTIM - Optimization module which implements the Complex method
- SETVAL - sets up independent variables from design variables
- MODULE - calls subprograms for unit modules
- PREPAR - pre-treatment to solve the simultaneous equation set
- TRIAN - solves the simultaneous equation set
- CALO - computes essential operations for solving the simultaneous equation set
- OBJECT - calculates the objective function based on an economic criterion chosen optionally from venture profit, venture cost or operating cost.

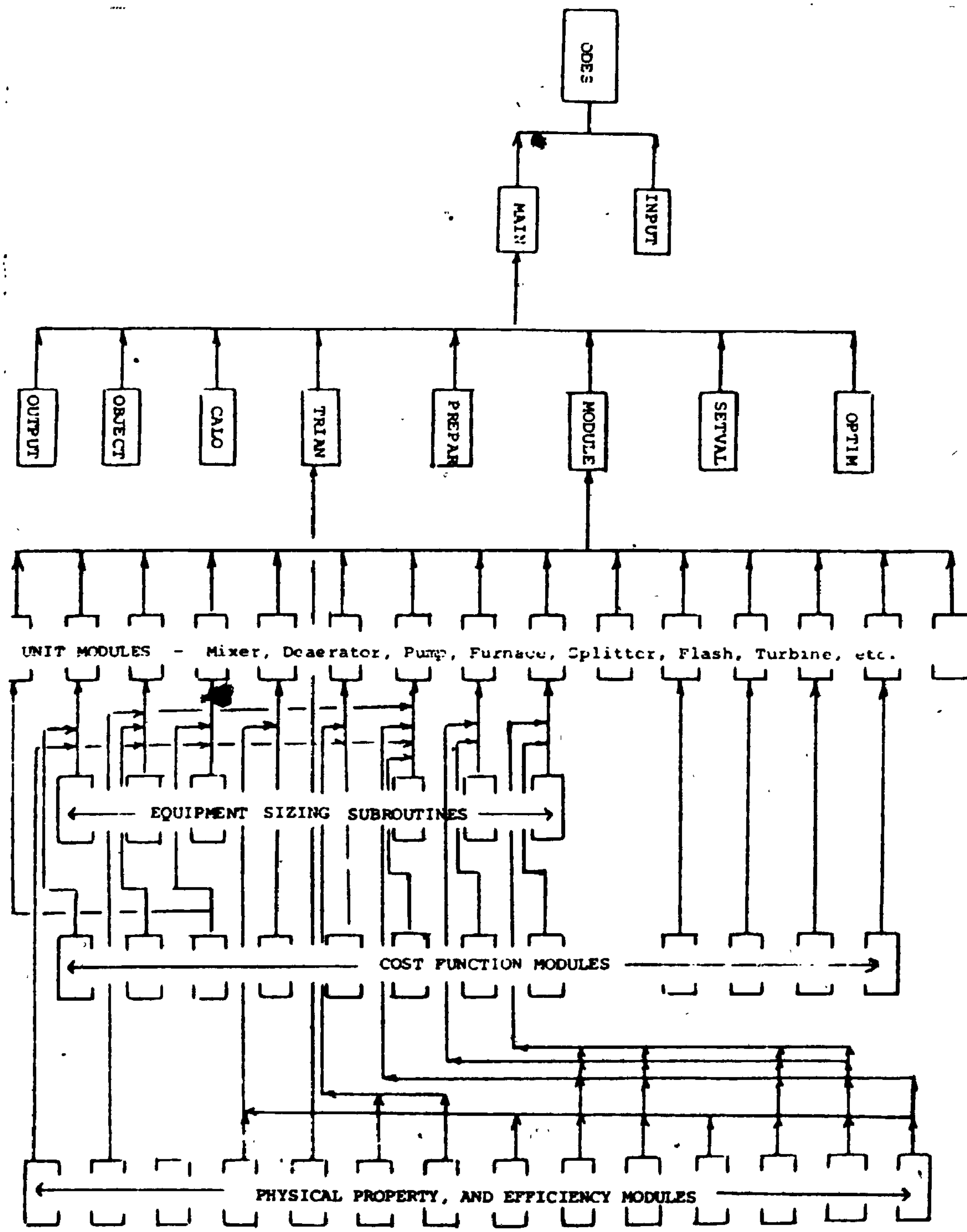


Figure 2.5 Computer Program Structure for an Optimal Design of Energy Systems

The subprogram which organizes subprograms in this group to implement the computational procedure given in Figure 2.4 is "MAIN".

2). The Unit Modules

These include - mixer, receiver of electric power, deaerator, pump, boiler, splitter etc. Each module involves balance formulations, equipment sizing and cost estimation according to the simulation level.

3) Subprograms for equipment sizing

4) Cost functions

5) Physical Properties, turbine efficiencies and numerical techniques

6) Subprograms for program I/O

INPUT - reads input data

OUTPUT - prints out computational results

Data access by the "user" can be made easily owing to data sophistication described below.

All program listings for the above subprograms are given in Appendix 9.

The data used in the program system can be classified mainly into the following four groups:

A. Data relating to energy systems

B. Data relating to the objective function

C. Data relating to solving the simultaneous equation set

D. Data relating to the optimization technique

From the standpoint of physical data structure, data with two dimensional arrays are transferred through arguments of the subprogram with variable dimensions while data with one dimensional array are transferred through "labelled common". This consideration makes a centralization of dimensioning possible as seen in an example dimensioning given in Listing 2.2

Stream data and equipment data which are major data in group A listed above are defined in Tables 2.1 and 2.2.

Stream codes are used for identifying each stream. Stream codes assign data length for each stream with different properties so that the address pointer for each stream can point to a head address which allocates the exact data length required. Stream codes are also used effectively to obtain aggregate utility consumptions and costs. Each equipment data length is specified by the "user" through a keyword so that the address pointer can prepare the data length necessary for each equipment having different data length. As seen in Table 2.2, all stream properties and equipment parameters are stored in the

LISTING 2.2 A CENTRALIZATION OF DIMENSIONING

```
      PROGRAM OOES(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C     PROGRAM FOR OPTIMAL DESIGN OF ARBITRARY ENERGY SYSTEMS
C
      COMMON /SYS1/SNEN(400),IS(80,2),IE(40,16)/SYS2/LENGS(16)
      COMMON /SYS3/NM,KYWRD(30)/OBJ1/JIN,IIN(10)/OBJ2/JOUT,IOUT(15)
      COMMON /OBJ3/COST(16)/OBJ4/ECON(20)/OBJ5/CRIT
      COMMON /SIM1/F(80),WORK(2),A(80,60)/SIM2/IL(4120)/SIM3/II(4120)
      COMMON /SIM4/IJ(4120)/SIM5/IK(4120)/SIM6/NI(80)/SIM7/NJ(80)
      COMMON /SIM8/NK(80)/SIM9/NS,ISP(15)
      COMMON /OPT1/MAXM,X(13)/OPT2/RMIN(13)/OPT3/RMAX(13)/OPT4/RLI(13)
      COMMON /OPT5/RUI(13)/OPT6/G,FUNC(14),XA(13,14)/OPT7/XO(13)
      COMMON /OPT8/XR(13)/OPT9/R(13)/OPT10/IX(13)/OPT11/NOD,CNST(20)
     1       ,ICON(20,3)
C     .
C     .
C     GENERAL DATA
      DATA NST,NEN,NEQ,NIS,NIE,NM,NC,NN,G /80,40,60,2,16,30,10, 5,0.001/
C     .
C     .
      CALL      MAIN  (A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICON,NC,XA,NV,NN)
      STOP
      END

      SUBROUTINE MAIN  (A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICON,NC,XA,NV,NN)
C     SUBPROGRAM TO ORGANIZE BASIC MODULES FOR AN OPTIMAL DESIGN
C     OF ARBITRARY ENERGY SYSTEMS
      DIMENSION      A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE),XA(NV,NN)
      DIMENSION      ICON(NC,3)
      COMMON /SIM1/   F(1)/SIM2/IL(1)/SIM6/NI(1)/SIM7/NJ(1)/SIM8/NK(1)
      COMMON /SYS1/SNEN(1)/OBJ4/ECON(1)
      COMMON /OPT1/MAXM,X(1)/OPT2/RMIN(1)/OPT3/RMAX(1)/OPT4/RLI(1)
      COMMON /OPT5/RUI(1)/OPT6/G,FUNC(1)/OPT7/XO(1)/OPT8/XR(1)/OPT9/R(1)
C     .
C     .
      RETURN
      END
```

K	ATTRIBUTE OF STREAM I (IS(I,K))
1	STREAM CODE*
2	ADDRESS POINTER FOR REAL STREAM DATA

K	ATTRIBUTE OF EQUIPMENT J (IE(J,K))
1	EQUIPMENT NUMBER
2	EQUIPMENT CODE
3	ADDRESS POINTER FOR REAL EQUIPMENT DATA
4	NO. OF INPUT STREAMS
5	NO. OF OUTPUT STREAMS
.	.
.	.
.	.

* 1 : LIQUID } PROCESS STREAM
2 : MIXTURE } PROCESS STREAM
3 : VAPOR } PROCESS STREAM
4 : LIQUID FUEL
5 : GAS FUEL
6 : INDUSTRIAL WATER
7 : COOLING WATER
8 : POWER
9 : ELECTRICITY
10 : LIQUID (PURE WATER)
11 : MIXTURE (CONDENSED STEAM)
12 : STEAM 1 (LOW PRESSURE STEAM)
13 : STEAM 2
14 : STEAM 3
15 : STEAM 4
16 : STEAM 5 (HIGH PRESSURE STEAM)

TABLE 2.1 IS AND IE DATA

K	ATTRIBUTE OF STREAM I (SNEN(K))
IS(I,2)+1	STREAM FLOW RATE
IS(I,2)+2	PROPERTY 1 (E.G. TEMPERATURE)
IS(I,2)+3	PROPERTY 2 (PRESSURE)
IS(I,2)+4	PROPERTY 3 (VAPOR RATIO OF MIXTURE)
. . .	. . .

K	ATTRIBUTE OF EQUIPMENT J (SNEN(K))
IE(J,3)+1	EQUIPMENT COST
IE(J,3)+2	PARAMETER 1
IE(J,3)+3	PARAMETER 2
IE(J,3)+4	PARAMETER 3
. . .	. . .

TABLE 2.2 SNEN (REAL STREAM AND EQUIPMENT) DATA

'SNEN' array. Thus, these data structures make possible a flexible and efficient storage of parameters such as stream and equipment data with variable length. Also this data sophistication makes data access by "users" easy because most of the information which is needed is found in stream and equipment data which have their addresses in a sequence order.

2.7 Summary

A new solution method for energy and material balances was proposed and a computer executive system "ODES" was developed to automate the method for the optimal design of arbitrary energy systems. Such devices as a modular approach for unit modeling, a design of flexible program and data structures and an implementation of optimization technique sufficiently stable to attain an optimal solution, have supported the generalization of the methodology.

The concept developed here can be applied to an optimal design of energy systems interacting strongly with process systems.

CHAPTER 3

OPTIMAL EXPANSION OF STEAM AND POWER PLANTS

3.1 Introduction

In solving a plant expansion problem involving complex energy systems there are two major factors that must be considered. One is the effect of the cost-estimate error and the demand-forecast error on the choice of an optimum system expansion plan. The other is the solution technique to obtain an optimum system expansion plan, which may be defined as the solution technique for a synthesis problem with constraints.

At the moment, there are two ways to treat the demand-forecast. One, which is sometimes seen in chemical plant design, is to describe the growth pattern by the classical S-shaped curve and treat the uncertainty in the demand-forecast by risk analysis (Coleman, 1964). Alternatively, stochastic treatment is often used in a determination of the installed reserve margin for public power generation (AIEE Committee Report, 1961; Brennan, 1958; Fitzpatrick, 1962; Kist, 1958; Reps, 1959). Neither method seems adequate when the problem at hand involves a steam and power plant for a chemical complex.

The solution technique to obtain an optimum system expansion plan, that is, the solution technique for a synthesis problem (Hendry, 1973) may take one of four basic approaches: decomposition, heuristic, algorithmic (or optimization), and evolutionary. A combination of these basic techniques is often used for the treatment of large systems. There have been relatively few successful applications for the synthesis of a large system with a heterogeneous structure (Menzies and Johnson, 1972).

In this chapter, the problem of the expansion of a steam and power plant for a chemical complex is formulated as a synthesis problem with constraints, and a choice of an optimal system expansion case is made by solving a multi-time period linear programming (LP) problem. A general program system which automatically formulates the LP models is developed so that quick case studies or parametric studies can support a final decision over uncertainties in the cost estimate or demand forecast.

3.2 Problem Statement

A steam and power plant for a chemical complex supplies electric power and steam at several possible levels, which are required for satisfying on-site and off-site demands, as well as for the steam and power plant itself (Jackson et al. 1965).

The planning of expansion of a steam and power plant is closely related to that of the expansion of the process system, especially in this age of energy shortage. Therefore, it is not realistic to merely describe the growth pattern of energy demand by S-shaped curves or regard it as a stochastic process. Instead, it has been common recently to regard it as a deterministic process and follow a trial and error approach aiming at an optimal expansion planning from the overall point of view. It would be desirable to provide an optimal expansion plan under whatever demand pattern there may be. The question to be answered for an expansion of steam and power plant, given arbitrary demands of the type illustrated in Figure 3.1 as an example, are the following:

1. What types of units should be installed to most economically meet growing needs?
2. When should these new units be installed?
3. When should old existing units be retired?

In other words, what is an optimum configuration of the steam and power plant during project life, given energy demands predicted, existing configuration and operating conditions?

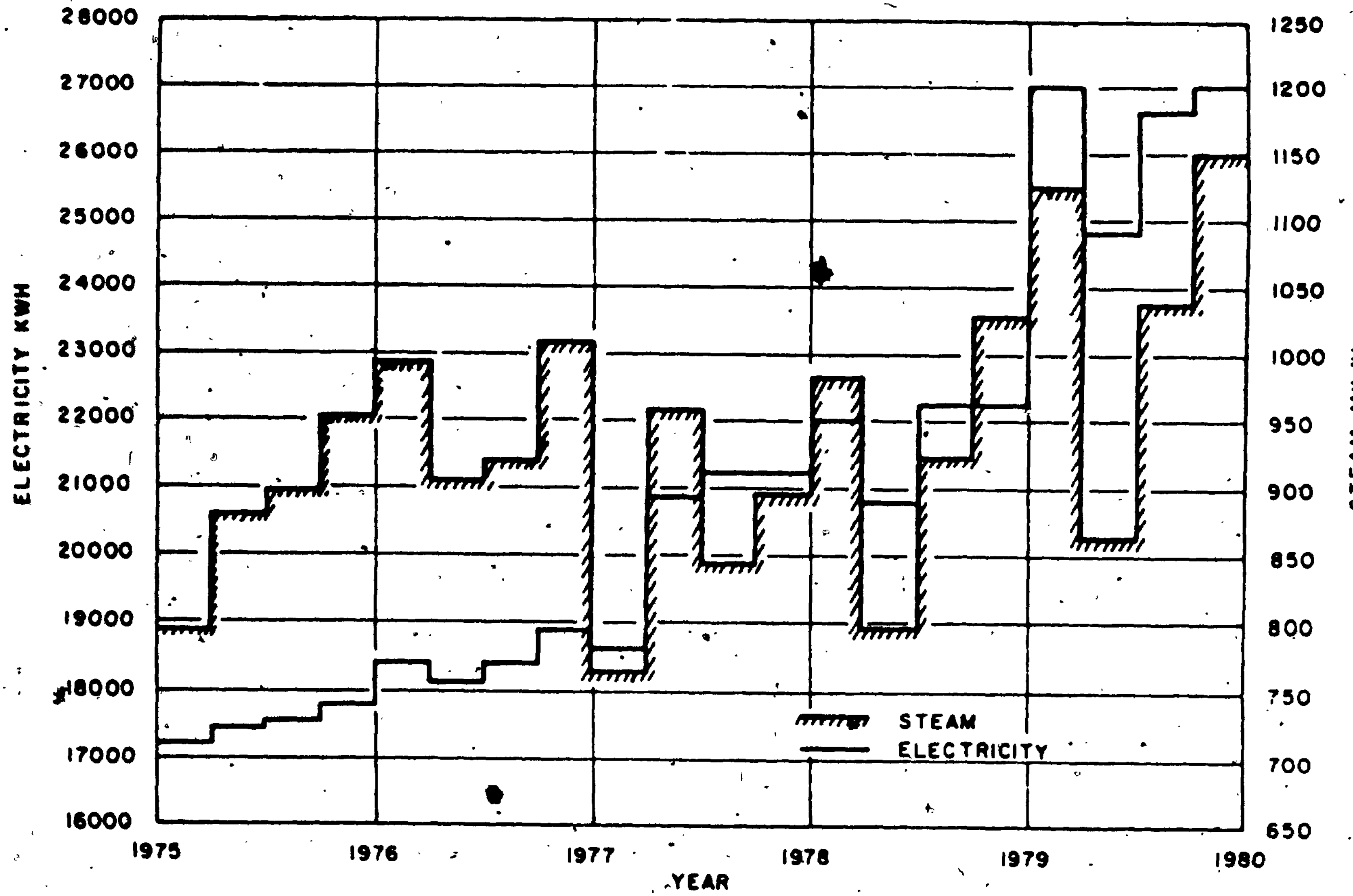


FIGURE 3.1 DEMAND FORECAST OF STEAM AND POWER

This can obviously be defined as a synthesis problem with constraints.

Inasmuch as use of the venture profit concept seems to be common in evaluating the attractiveness of a venture in a chemical complex, the same criterion should be applied to a steam and power plant that is a "decomposed" sub-system in the complex. For an expansion problem, it would be reasonable to choose as a criterion the venture worth (Happel, 1958; Generoso, 1968) which judges the profitability of a project by the present value of venture profit obtained during each year of the project's expected life. The detailed derivation is shown in Appendix 1.

3.3 Choice of an Optimal Expansion Case

As described in the preceding section, the relation between supply and demand can be approximated as a deterministic process, and venture worth is used as an economic criterion for the choice of an optimal expansion case. Operating conditions cannot be changed extensively because of restrictions on the existing plant; therefore, the energy balance can be expressed in terms of linear relations including those linking the multi-time period, which are generally expressed as follows:

$$W_{j,k} = \sum_{i=o}^{j} Y_{i,j,k} \quad (j=1, \ldots, n \text{ period}) \qquad (3.1)$$

$$Y_{i,j,k} \leq Z_{i,k} \quad ((j=i, \ldots n), i=1, \ldots n) \qquad (3.2)$$

$$Y_{o,j,k} \leq Z_{o,k} \quad (j=1, \ldots n) \qquad (3.3)$$

where

- k : the number identifying a unit
- $W_{j,k}$: demand to unit k required at period j
- $Y_{i,j,k}$: supply provided at period j by $Z_{i,k}$, the capacity of unit k that is expanded at period i
- $Z_{o,k}$: existing capacity of the unit k

Then, a preliminary choice of an optimum system expansion case may be achieved using a linear programming technique. Figure 3.2 outlines a computational procedure for the choice of an optimal expansion case. The LP formulation for the multi-time period becomes tedious as the period considered becomes long. Therefore, automatic formulation and set-up of LP models are desired, as described in the following section. At the same time, the size of the LP problem requires a commercial LP package from the standpoint of computation time and core requirement, hence employed is MPSX of IBM as an LP "solver".

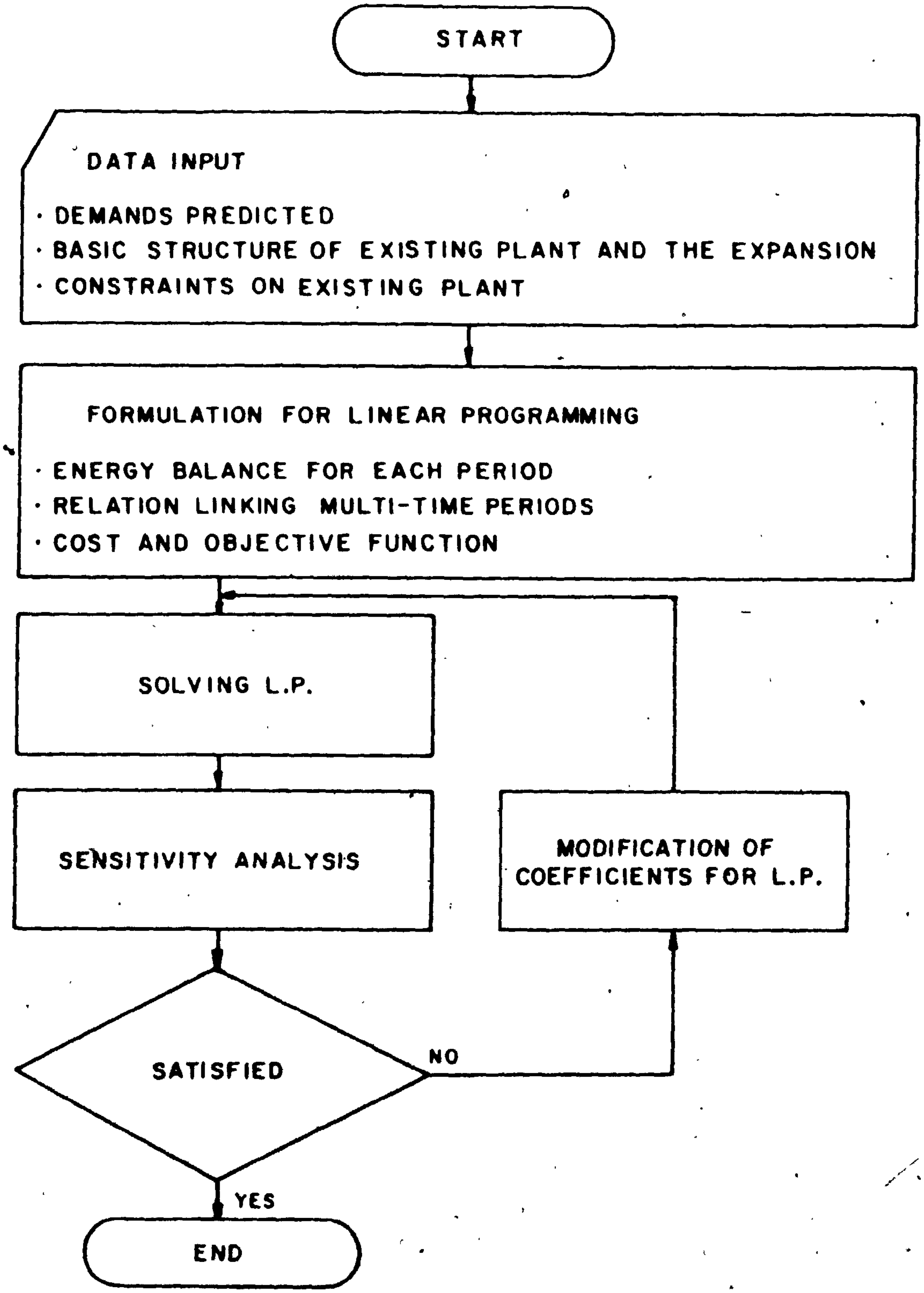


FIGURE 3.2 COMPUTATIONAL PROCEDURE FOR THE CHOICE OF AN OPTIMAL EXPANSION PLAN

3.4 Computer Executive System for LP formulation of Expansion Problem

Listing 3.1 shows an input example for a simple expansion problem, which will be taken up as the first example of the applications in Chapter 6. The use of keywords and stream numbers is not only convenient for the input of basic information such as demands predicted for the multi-time period, basic structure of existing plant, and constraints on existing plant (e.g., capacity, operating conditions), but also assists the LP formulation of the expansion problem. Such sophistication of input information makes case study easy and LP formulation by a modular approach possible.

The computational procedure for the automatic generation of LP models is summarized in Figure 3.3, provided that the LP package of MPSX by IBM is available to solve the LP problem.

Figure 3.4 shows the program system structure which makes the execution of the computational procedure possible. The structure is quite flexible and open-ended so that new unit modules may be added for an evaluation of arbitrary energy systems.

The data structure to support the program system is basically the same as that of the program system "ODES"

```
   MAXST    MAXEQ      IUNIT     IPRIOD
      24       10          2          3
STREAM CODE
 10   12   10   10    8   10   13   10   10    4    8   14   13   12    9    9
  9    9   13   13   12   12   12   -0
                          NO. OF I/O STREAMS   STREAM NO. RELATING TO 'EQUIPMENT'
SYSTEM NETWORK
DEAER      1    3    3    1    1    2    3    4   -0   -0   -0   -0   -0   -0
PUMP       2    3    2    1    4    5    6   -0   -0   -0   -0   -0   -0   -0
HEATR      3    4    2    2    7    6    3    8   -0   -0   -0   -0   -0   -0
BOILR      4    6    3    2    8   11   10   12    9   -0   -0   -0   -0   -0
GTURBI     5    4    1    3   12   13   14   15   -0   -0   -0   -0   -0   -0
ERECIV     6    3    2    2   15   16   17   18   -0   -0   -0   -0   -0   -0
MOTOR      7    3    1    1   18   11   -0   -0   -0   -0   -0   -0   -0   -0
SPLTR      8    1    1    4   13    7   20   19   24   -0   -0   -0   -0   -0
TURBB      9    3    1    2   20    5   21   -0   -0   -0   -0   -0   -0   -0
JUNC      10    1    2    3   14   21    2   23   22   -0   -0   -0   -0   -0
STREAM DATA     11 PERIOD-   1
STREAM     1    2      122.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM     3    2      365.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM     4    3       29.700    2    250.000   -0     -0.000   -0      -0.000
STREAM     6    3      500.000    2    250.000   -0     -0.000   -0      -0.000
STREAM     8    3      474.700    2    350.000   -0     -0.000   -0      -0.000
STREAM    12    3      474.700    2    650.000   -0     -0.000   -0      -0.000
STREAM    13    3      179.700    2    520.000   -0     -0.000   -0      -0.000
STREAM    14    3       29.700    2    300.000   -0     -0.000   -0      -0.000
STREAM    17    1    35100.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM    19    1     1284.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM    22    1       91.000   -0     -0.000   -0     -0.000   -0      -0.000
EQUIPMENT DATA     9
DEAER      1    2     1600.000    3      4.000    1      0.030   -0      -0.000
PUMP       2    2     1600.000    3      6.000    1      0.006   -0      -0.000
HEATR      3    2     3000.000    3     -0.000    4    500.000    1       0.050
BOILR      4    2     1900.000    3     12.000    4      0.040    5   19000.000
BOILR      4    6        1.471    1      6.330   -0     -0.000   -0      -0.000
GRURBI     5    2    33000.000    3     15.000    1      0.100   -0      -0.000
ERECIV     6    2    40000.000    3     -0.000    1      0.200   -0      -0.000
MOTOR      7    2     3320.000    3     11.000    1      0.041   -0      -0.000
TURBB      9    2     1640.000    3      5.000    1      0.037   -0      -0.000
STREAM DATA      3 PERIOD-   2
STREAM    17    1    35100.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM    19    1     1430.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM    22    1      119.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM DATA      3 PERIOD-   3
STREAM    17    1    41700.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM    19    1     1563.000   -0     -0.000   -0     -0.000   -0      -0.000
STREAM    22    1      178.000   -0     -0.000   -0     -0.000   -0      -0.000
```

LISTING 3.1 INPUT DATA FOR AN EXPANSION PROBLEM

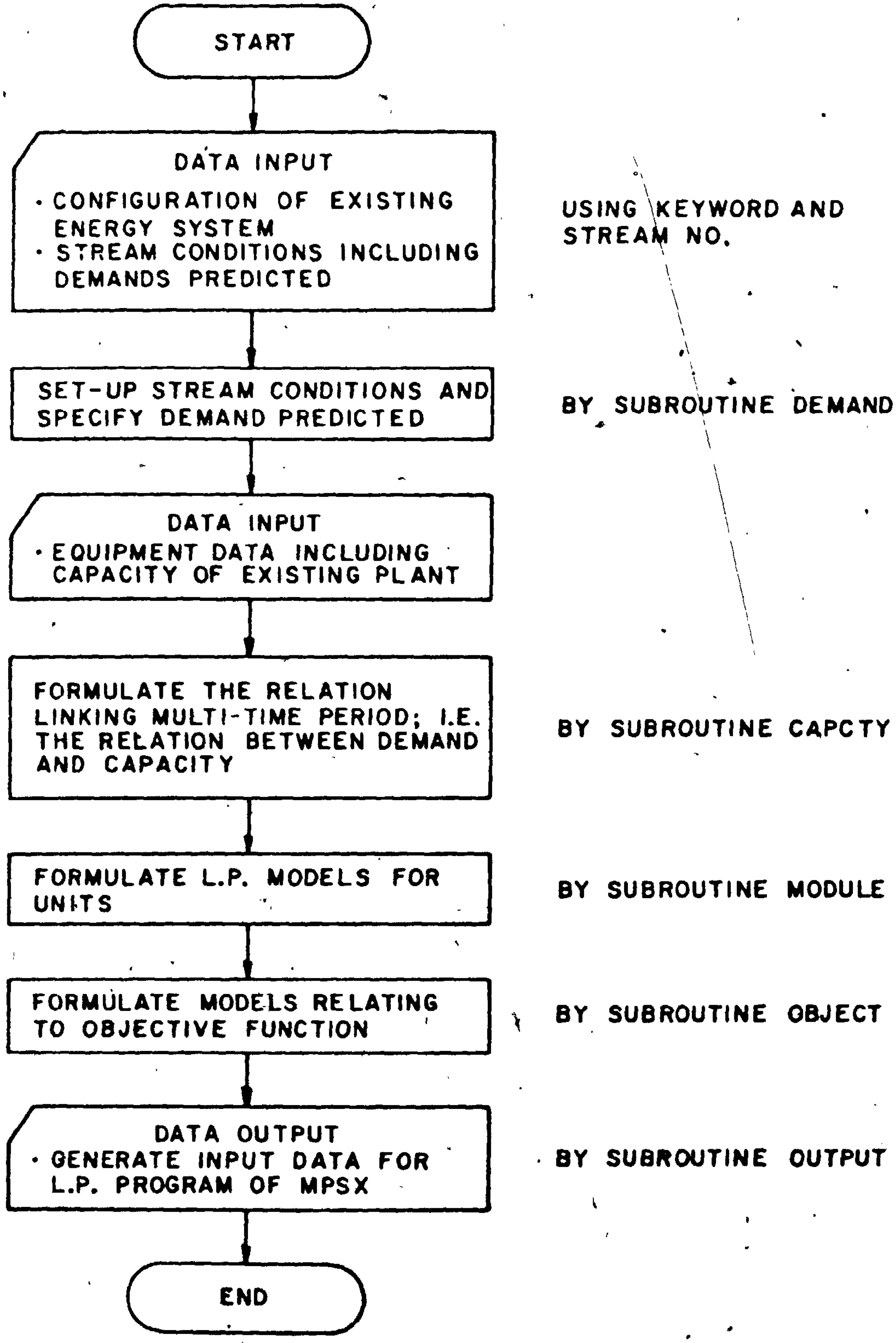


FIG. 3.3 A COMPUTATIONAL PROCEDURE FOR GENERATING INPUT DATA TO THE L.P. PROGRAM OF MPSX.

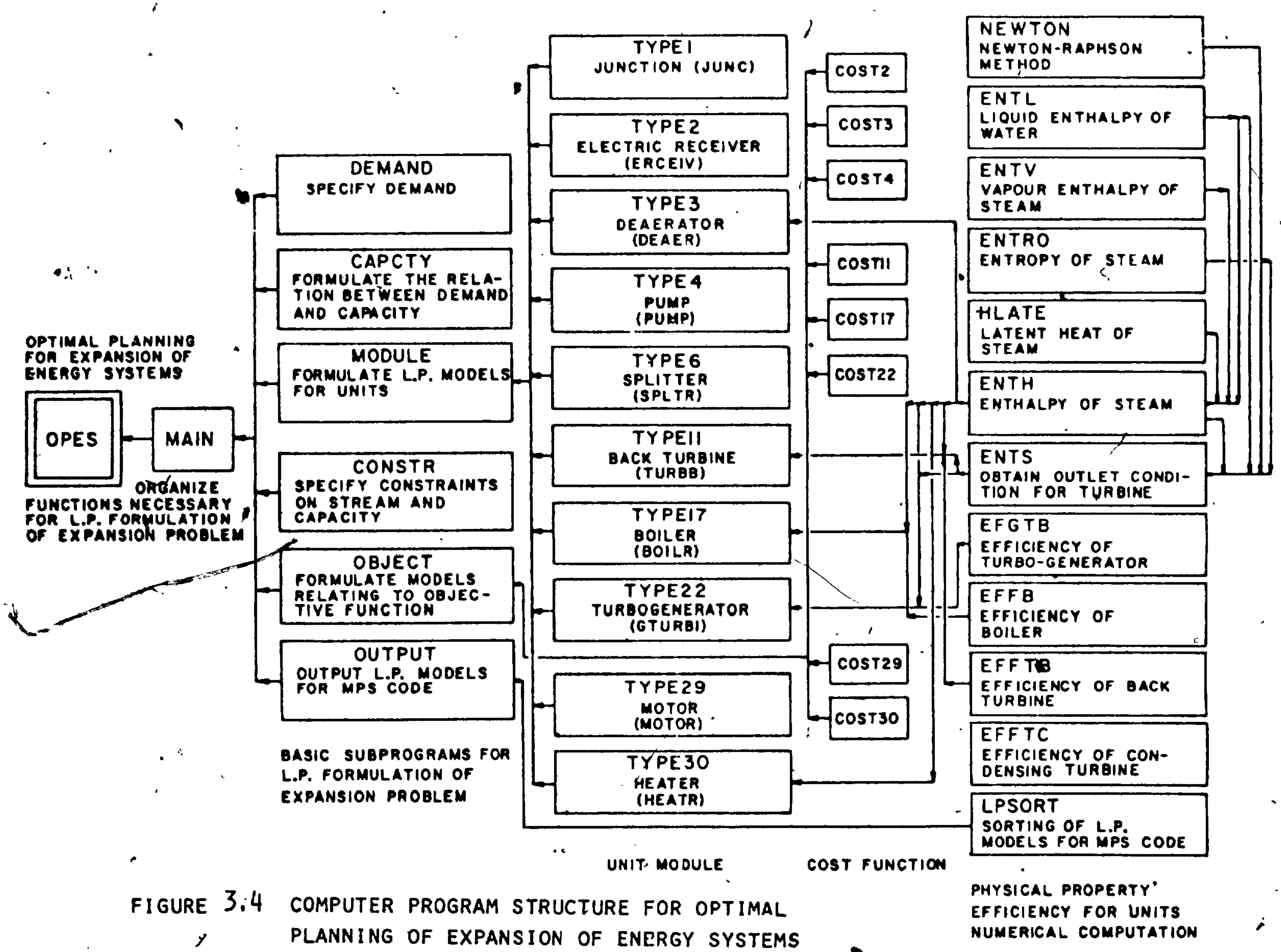


FIGURE 3.4 COMPUTER PROGRAM STRUCTURE FOR OPTIMAL PLANNING OF EXPANSION OF ENERGY SYSTEMS

and has the same features which were described in Section 6, Chapter 2. Additional data necessary for LP formulation are defined in Table 3.1. This data sophistication is a key to formulate LP models systematically and makes computational results easy to analyse. This will be explained in detail in Appendix 7.

3.5 LP Modeling by a Modular Approach

As mentioned before, the use of keyword identifiers and stream numbers to represent a system configuration serves to formulate LP models systematically using a modular approach.

A typical example of LP modeling by a modular approach is described below.

For a back-pressure turbine unit for which a symbolic figure is shown in Figure 3.5, the equations are:

For steam flows $$X_1 - X_2 = 0 \qquad (3.4)$$

An energy balance yields

$$X_3 - c_f\,(i_1 - i_2)\eta X_1 = 0 \qquad (3.5)$$

where notations have the same meaning as those in unit modeling in Chapter 2. These equations are coded actually

Table 3.1 Basic Data for LP Formulation of the Expansion Problem

Entity No. / Attribute	Element N for LP formulation
IROW(N)	$M + L*10^2 + (K*10^4) + JCODE*10^6$
ICOL(N)	I or $K + L*10^2 + (J*10^4) + ICODE*10^6$
RDATA(N)	Real Data

Entity No. / Attribute	Sequence Number MEQ for LP equation
NEQ(MEQ)	$M + L*10^2 + (K*10^4) + JCODE*10^6$

where M ; Sequence no. of equation

I ; Stream identification no.

K ; Equipment identification no.

IROW ; Information for row identification

ICOL ; Information for column identification

NEQ ; Same information as IROW, for row specification such as 'greater', 'equal' or 'less'.

L ; Period operated

J ; Period expanded

ICODE ; 1 : Variable for energy balance (X)

2 : Variable for capacity demand (W)

3 : Variable for capacity supply (Y)

4 : Variable for capacity (Z)

5 : Variable for cost (C)

(codes 1–5: COLUMNS)

Table 3.1 Basic Data for LP Formulation of the Expansion Problem

(continued)

ICODE ;	6 : Variable for RHS (right hand side)	(R)	RHS
	7 : Variable for energy balance	(X)	BOUNDS
	8 : Variable for capacity demand	(W)	BOUNDS
	9 : Variable for capacity	(Y)	BOUNDS
JCODE ;	1 : Equation for energy balance		
	2 : Equation for demand-supply		
	3 : Equation for supply-capacity		
	4 : Equation for cost evaluation		
	5 : Lower bound		
	6 : Upper bound		
	7 : Objective function		

using data structures defined in the preceding section. The complete program listing for this unit module is shown in Listing 3.2. Also, program listings for other unit modules are attached in Appendix 10.

After defining all LP models, a set of data, IROW(N),ICOL(N) and RDATA(N), which are essential LP data, are sorted out by subprogram LPSORT listed in Appendix 10 in order to follow the input form to MPSX as shown in Appendix 8.

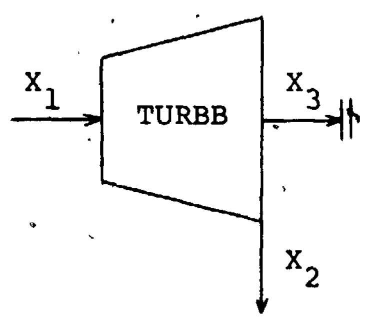


Figure 3.5 Symbolic Figure for a Back Pressure Turbine Unit

3.6 Summary

The expansion problem of a steam and power plant for a chemical complex was formulated as a synthesis problem with constraints, and solving a multi-time period linear programming problem was proposed to choose an optimal

LISTING 3.2 MODULE FOR BACK-PRESSURE TURBINE UNIT FOR AN EXPANSION PROBLEM

```
      SUBROUTINE TYPE11(L,IS,IE,NST,NEN,NIS,NIE)
C     PROGRAM FOR BACK TURBINE (TURBH)
C     I1                  .... STREAM NO. OF INLET STEAM
C     I2                  ... STREAM NO. OF OUTLET STEAM
C     I3                  ... STREAM NO. OF OUTPUT POWER
C     E1                  ... INLET ENTHALPY
C     E2                  ... OUTLET ENTHALPY AFTER ISENTROPIC EXPANSION
C     N                   ... SEQUENCE NUMBER OF LP DATA
C     M                   ... NO. OF ENERGY AND MASS BALANCE EQUATIONS
C     MEQ                 ... SEQUENCE NO. OF LP EQUATION
C     IROW(N)             ... INFORMATION FOR ROW IDENTIFICATION
C     ICOL(N)             ... INFORMATION FOR COLUMN IDENTIFICATION
C     NEQ(MEQ)            ... SAME INFORMATION AS IROW,ROW SPECIFICATION
C     SNEN(IS(I1,2)+2)    ... TEMPERATURE OF INLET STEAM
C     SNEN(IS(I1,2)+3)    ... PRESSURE    OF INLET STEAM
C     IUNIT               ... 1 - C.G.S. UNIT , 2 - BTU-LB UNIT
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
C     STEAM BALANCE
      J1=IE(IEQ,1)*10000
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=I1+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+L+1000000
      RDATA(N)=-1.
C     MODEL FOR POWER GENERATED BY STEAM EXPANSION
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=I3+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+L+1000000
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(I9(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1          SNEN(IS(I2,2)+2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)=-(E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1         SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      RETURN
      END
```

Table 4.2 Types of Generator Turbines and Drivers

1. Extractive back-pressure turbine (HPS to MPS and LPS)
2. Back-pressure turbine (HPS to MPS)
3. Back-pressure (HPS to LPS)
4. Back-pressure turbine (MPS to LPS)
5 Extractive condensing turbine (HPS to MPS and vacuum)
6. Extractive condensing turbine (HPS to LPS and vacuum)
7. Condensing turbine (MPS to vacuum)
8. Motor (for driver selection only)

CHAPTER 4

OPTIMAL SYNTHESIS OF STEAM AND POWER PLANTS

4.1 Introduction

Studies concerned with the synthesis of chemical processes were reviewed by Hendry et al. (1973). It seems thus far that four basic approaches have been used as solution techniques for a synthesis problem; i.e., decomposition, heuristic, algorithmic (or optimization) and evolutionary although there also exist rather inefficient approaches such as the trial and error approach (Umeda, 1972). Combinations of these basic techniques seem to be often used for the treatment of large systems.

In this chapter, the combination of three basic approaches i.e., decomposition, heuristic and optimization is used as a solution technique for an optimal synthesis of steam and power plant for a chemical complex which is a large system with a heterogeneous structure. The solution method as a whole is generalized by the development of two computer executive systems, OSES and ODES.

First, the synthesis problem is defined. Second, the basic procedure is introduced for choosing an optimal configuration. Third, the program system and data structure to support the computer executive system OSES are defined. Fourth and finally, unit modeling by a modular approach is described as a general method to formulate the sub-systems which make up large energy systems.

4.2 Problem Statement

A steam and power plant for a chemical complex is a type of multi-product process. The product demands are given in terms of steam demands with different pressure levels, and electric power demands including electricity for lighting, instrumentation etc. Table 4.1 is an illustrative set of steam and power demands. It should be noted that not only are there many demands but there are two types of power demands, i.e., the internal demands needed in the steam and power plant so that the plant can be operated, and the external demands which are required outside of the plant. The internal power demand depends entirely upon the configuration of the plant which has to be synthesized. And also this consideration of internal demands is based on a set of heuristics for the synthesis of the steam and power plant for a chemical complex.

Table 4.1 Steam and Power Demands for the Synthesis Problem

Demand Items		Stream Number	Demands
Electricity		71	26140 kwH
High Pressure Steam		25	75.6 Mlb/H
Medium Pressure Steam		74	1100 Mlb/H
Low Pressure Steam		94	139.3 Mlb/H
Condensate		81	26.3 Mlb/H
By-pass steam		26	> 410 Mlb/H
		80	> 10 Mlb/H
Internal			
power demand	1	4	(pump eq.2)
" "	2	4	(Pump eq.4)
" "	3	4	(Boiler eq.8)
" "	4	4	(Pump eq.26)
External			
power demand	5	4	184 KwH
" "	6	4	1288 KwH
" "	7	4	257.5 KwH
" "	8	4	225 KwH
" "	9	4	456 KwH
" "	10	4	368 KwH

In general, in a steam and power plant for a chemical complex, the basic structure of the plant can be proposed easily according to "external" steam and power demands and the circumstances of the situation. For instance, there is no question that boiler facilities, deaerator and water treatment facilities are all necessary at appropriate juxtapositions. Also, a boiler-feed-water (BFW) pump is a requisite to pump the treated water up to boiler feed conditions. In addition, some drivers, types of which are unknown at first, must be provided for power supplies to boiler draft fan, BFW pump and water supply pump to deaerator. A flash drum may be needed for the recovery of blow-down boiler water.

Thus, by making use of heuristics derived from the particular conditions of the situation, the basic configuration of steam and power plant can be quite clearly structured. For example, the basic structure for demands given in Table 4.1 is shown in Figure 4.1, where the imaginary network for driver selection is represented only for a single power demand although many power demands actually exist. This will be treated properly by a network processing technique at a later stage of model formulation.

Then, the practical questions to be answered are as follows:

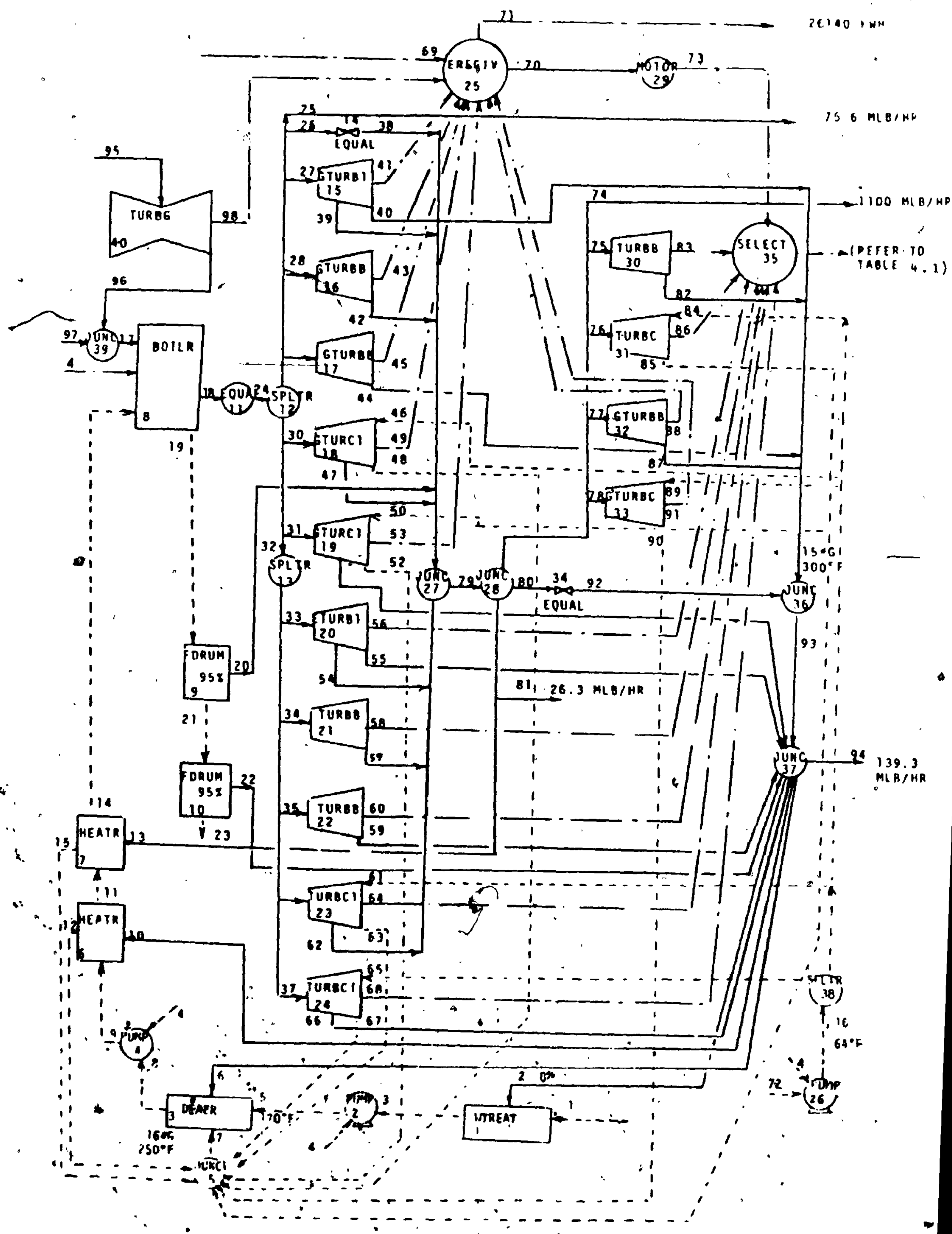


Figure 4.1 Information Flow Diagram of a Steam and Power Plant

For large number of motive power demands, should motor-driven or turbine-driven schemes be employed? If turbines are used, what type of turbine should be selected, i.e., a back-pressure turbine, extractive turbine or condensing turbine? For electric power demand including electricity for lighting and instrumentation etc., what is an electric power generating system like, in other words, a gas turbine generator or steam turbine generator or combination between them? In the case of steam turbine use, what type of turbine should be selected? What is an optimum energy balance to satisfy power and steam demands of different pressure levels?

To summarize the problem statement, under the energy demands given in Table 4.1, the problem of optimal synthesis is to determine in Figure 4.1 the arrangement and type selection of generators and drivers listed in Table 4.2 in order to minimize the venture cost (including the fixed cost and operating cost) which will be well-defined in Appendix 1. The above determination should be made under the condition of optimal design conditions which make an optimum energy balance possible.

Table 4.2 Types of Generator Turbines and Drivers

1. Extractive back-pressure turbine (HPS to MPS and LPS)

2. Back-pressure turbine (HPS to MPS)

3. Back-pressure (HPS to LPS)

4. Back-pressure turbine (MPS to LPS)

5 Extractive condensing turbine (HPS to MPS and vacuum)

6. Extractive condensing turbine (HPS to LPS and vacuum)

7. Condensing turbine (MPS to vacuum)

8. Motor (for driver selection only)

4.3 Choice of an Optimal Configuration

As described in the preceding section, the determination of an optimal configuration is not independent from that of optimal design conditions which make an optimum energy balance attainable. They are closely related in the sense that the determination of optimal configuration is made with optimum parameters assumed and the determination of optimum design conditions is made under the given configuration assumed to be optimal. While these items should be dealt with simultaneously, the problem of arrangement and selection of equipment differs in solution technique from that of parameter optimization. Therefore, a two-level approach is used in this work as a solution method.

The computational procedure to implement a two-level approach is shown in Figure 4.2. Namely, an optimal configuration is determined at the upper level by solving a linear programming (LP) problem and parameter adjustments are made at the lower level by using a parameter optimization technique. The co-ordination between two levels is carried out by iteration since decisions concerning the upper level affect the decisions for the lower level and vice versa.

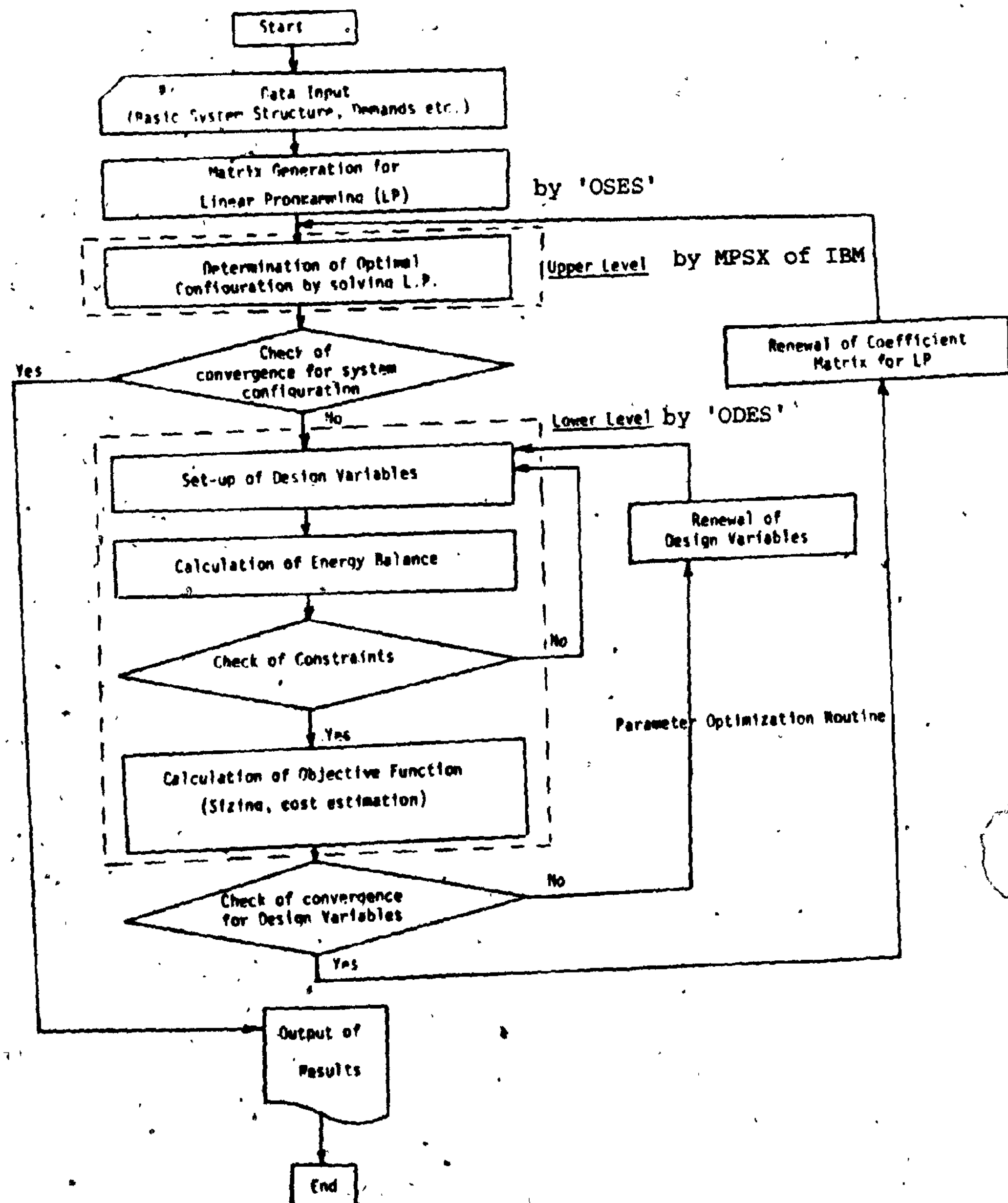


Figure 4.2 Computational Procedure for an Optimal Synthesis of a Steam and Power Plant

Convergence is not guaranteed in this method, but there was no difficulty encountered in all of the cases tested, for the tolerances set.

Although the upper level should be formulated theoretically in terms of a mixed integer program, it has been determined in this study through numerical tests that the formulation by LP is satisfactory from the viewpoint of computational efficiency and accuracy required.

The solution method proposed above is generalized within two computer executive systems OSES and ODES as described in the following section.

4.4 Computer Executive Systems "OSES" and "ODES"

It would normally be a difficult task to set up an LP problem at the upper level every time renewal of the coefficient matrix for the LP is required or studies on different cases are needed. Similarly it is not practical either to write a specific program for parameter optimization at the lower level every time a new configuration has to be evaluated. Therefore, the LP formulation at the upper level is automated by the general computer executive system called "OSES", and the parameter optimization at the lower level is automated in another computer executive called "ODES".

A computational procedure for the LP formulation is shown in Figure 4.3, and Figure 4.4 represents the program system structure of "OSES" which is the same as that of the "OPES" program system for expansion problems which was described in Chapter 3. The program data structure is basically the same as that of "OPES" except the different indication for information code J, i.e., the period expanded in Table 3.2 was replaced by the identification number for driver selection as shown in Table 4.3.

The "ODES" program system for parameter optimization implements the automation for the lower part of the computational procedure shown in Figure 4.2. The system has a similar structure to "OSES" and is believed flexible enough to evaluate any configuration. Energy and material balances are formulated into a set of homogeneous linear equations and efficiently solved by a triangulation method. The complex method is employed as an optimization technique. The details for "ODES" have been described in Chapter 2.

4.5 Unit Modeling by a Modular Approach

Again, a modular approach is used to build LP models in the upper level. The unit modeling by a modular approach is exactly the same as that for the expansion problem which

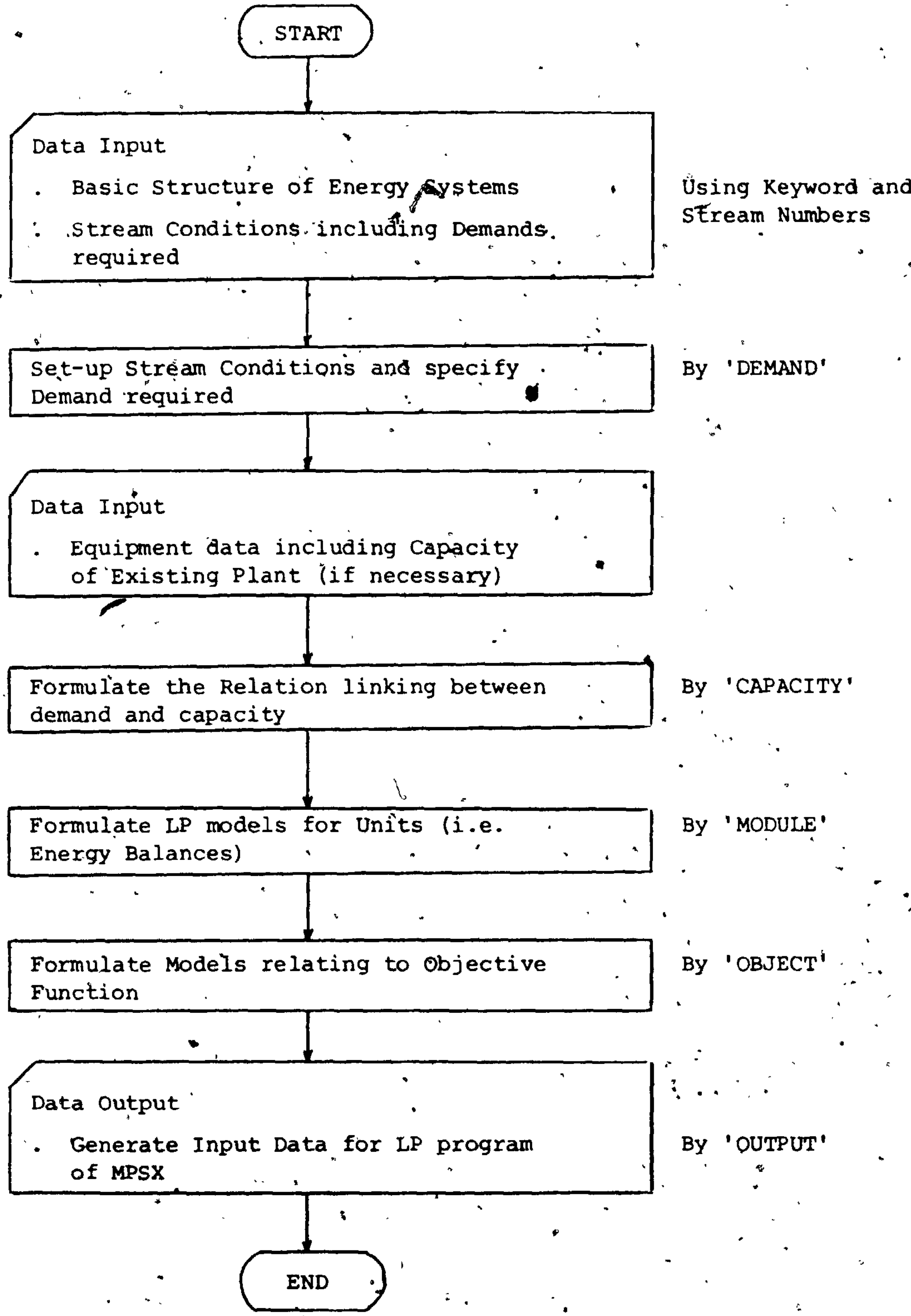


FIGURE 4.3 A COMPUTATIONAL PROCEDURE FOR MATRIX GENERATION FOR LP

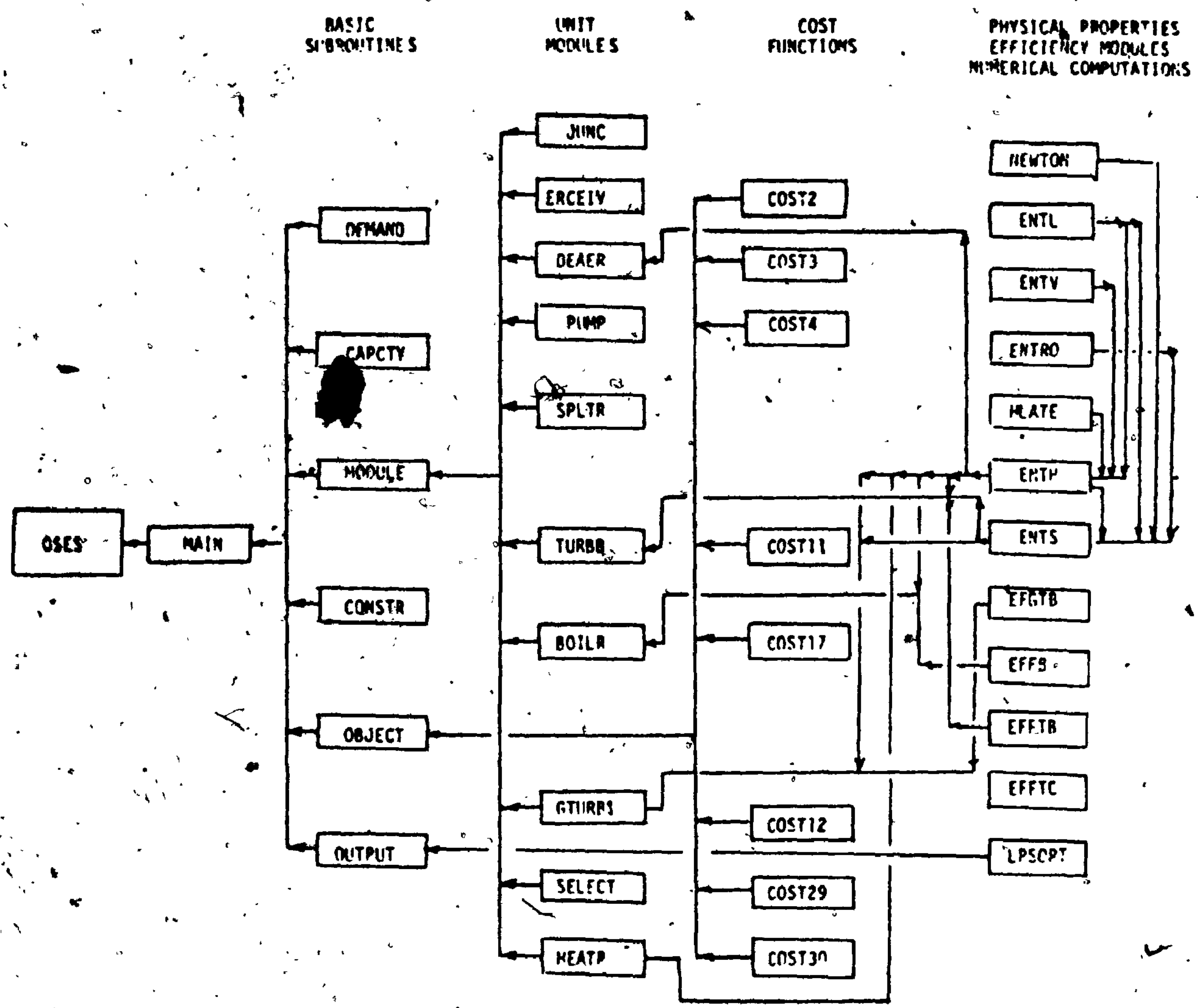


Figure 4.4 Computer Program Structure for Optimal Synthesis of Energy Systems

Table 4.3 Basic Data for LP Formulation of the Synthesis Problem

Attribute \ Entity No.	Element M for LP formulation
IROW(M)	$M + L*10^2 + (K*10^4) + JCODE*10^6$
ICOL(M)	I or $K + L*10^2 + (J*10^4) + ICODE*10^6$
RDATA(M)	Real Data

Attribute \ Entity No.	Sequence number MEQ for LP equation
NEQ(MEQ)	$M + L*10^2 + (K*10^4) + JCODE*10^6$

where M ; Sequence number of equation

I ; Stream identification number

K ; Equipment identification number

IROW ; Information for row identification

ICOL ; Information for column identification

NEQ ; Same information as IROW, for ROW specification ($\geq$ $\leq$ $=$)

L ; Period operated

J ; Identification number for driver selection

ICODE ; 1 : Variable for energy balance (X)

2 : Variable for capacity demand (W)

3 : Variable for capacity supply (Y)

4 : Variable for capacity (Z)

5 : Variable for cost (C)

(1–5: COLUMNS)

Table 4.3 Basic Data for LP Formulation of the Synthesis Problem (continued)

ICODE ;	6 : Variable for RHS (right hand side)	(R)		RHS
	7 : Variable for energy balance	(X)	}	BOUNDS
	8 : Variable for capacity demand	(W)	}	BOUNDS
	9 : Variable for capacity	(Y)	}	BOUNDS
JCODE ;	1 : Equation for energy balance			
	2 : Equation for demand-capacity			
	3 : Equation for cost evaluation			
	4 : Lower bound			
	5 : Upper bound			
	6 : Objective function			

was described in Chapter 3. Therefore, only the modules particular to the synthesis problem will be described here.

One module which is unique for the synthesis problem of steam and power plants is for driver selection and having the keyword identifier 'SELECT'. As seen in Figure 4.1 and Table 4.2, all types of drivers are connected with equipment 'SELECT'. This special module creates all equations necessary for the 'connected' driver candidates for each power demand. From a structural point of view, an imaginary network is made up for driver selection, according to the number of power demands required. A flow chart for processing the network for driver selection is shown in Figure 4.5.

Whereas the module 'CAPCTY' in the expansion problem sets up the relation between demands and, existing and expansion capacities, here in the synthesis problem it formulates the relation between demand and newly installed capacity which is expressed as follows:-

$$Z_k \geq W_{j,k} \quad (j = 1, ..n) \tag{4.1}$$

where

- k : the number identifying a unit
- Z_k : capacity of unit k to be installed
- $W_{j,k}$: demand to unit k required at period j

Where seasonal variation of demands is large, several

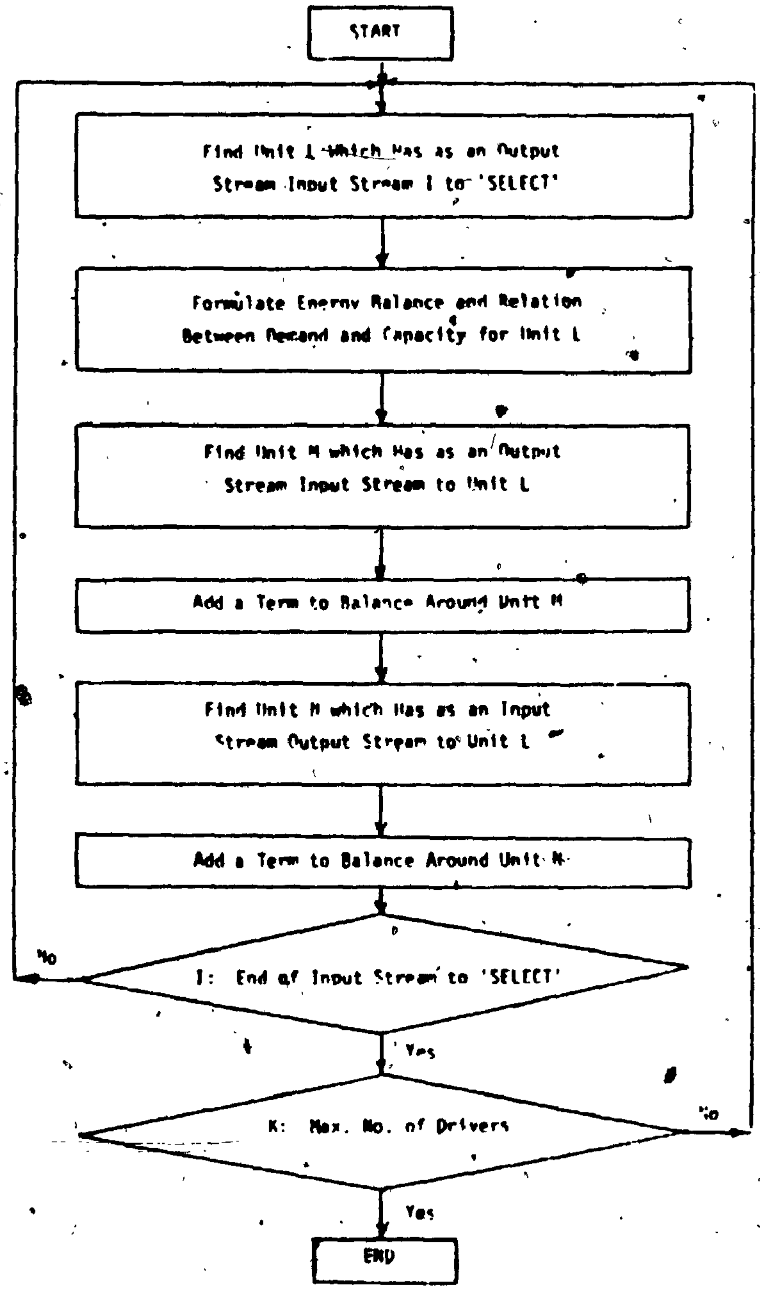


Figure 4.5 Flow Charts for Processing Networks for Driver Selection

periods, e.g., summer and winter terms, can be taken into consideration.

Program listings for modules are shown in Appendix 10.

4.6 Summary

The synthesis problem of steam and power plants was defined and a two-level approach, where optimal configuration is determined at the upper level by solving a linear programming problem and parameter adjustments made at the lower level by using a parameter optimization technique, was basically used as a solution technique. Generalization of methodology is supported by two computer executive systems "ODES" and "OSES".

CHAPTER 5

APPLICATIONS FOR COMPUTER EXECUTIVE SYSTEM "ODES"

5.1 Economic Evaluation of Gas Turbine Plus Medium Pressure Steam Cycle

The first example of the use of "ODES" is an economic evaluation for a gas turbine plus medium pressure steam cycle for an ethylene plant. The system is given in Figure 5.1, with utility requirements listed in Table 5.1. Figure 5.1 is one of four power cycles which were evaluated by Arstein and O'Connell (1968) to select the optimum heat cycle under requirements from 500,000,000 lbs/yr ethylene plant. However, the selection of the optimum heat cycle is not a goal in this section although a case study comparing alternate cases can be easily done by the use of "ODES" as the paper by Arstein and O'Connell did. Instead, the aim here is to illustrate how usefully the computer executive system "ODES" can be utilized to evaluate an arbitrary system.

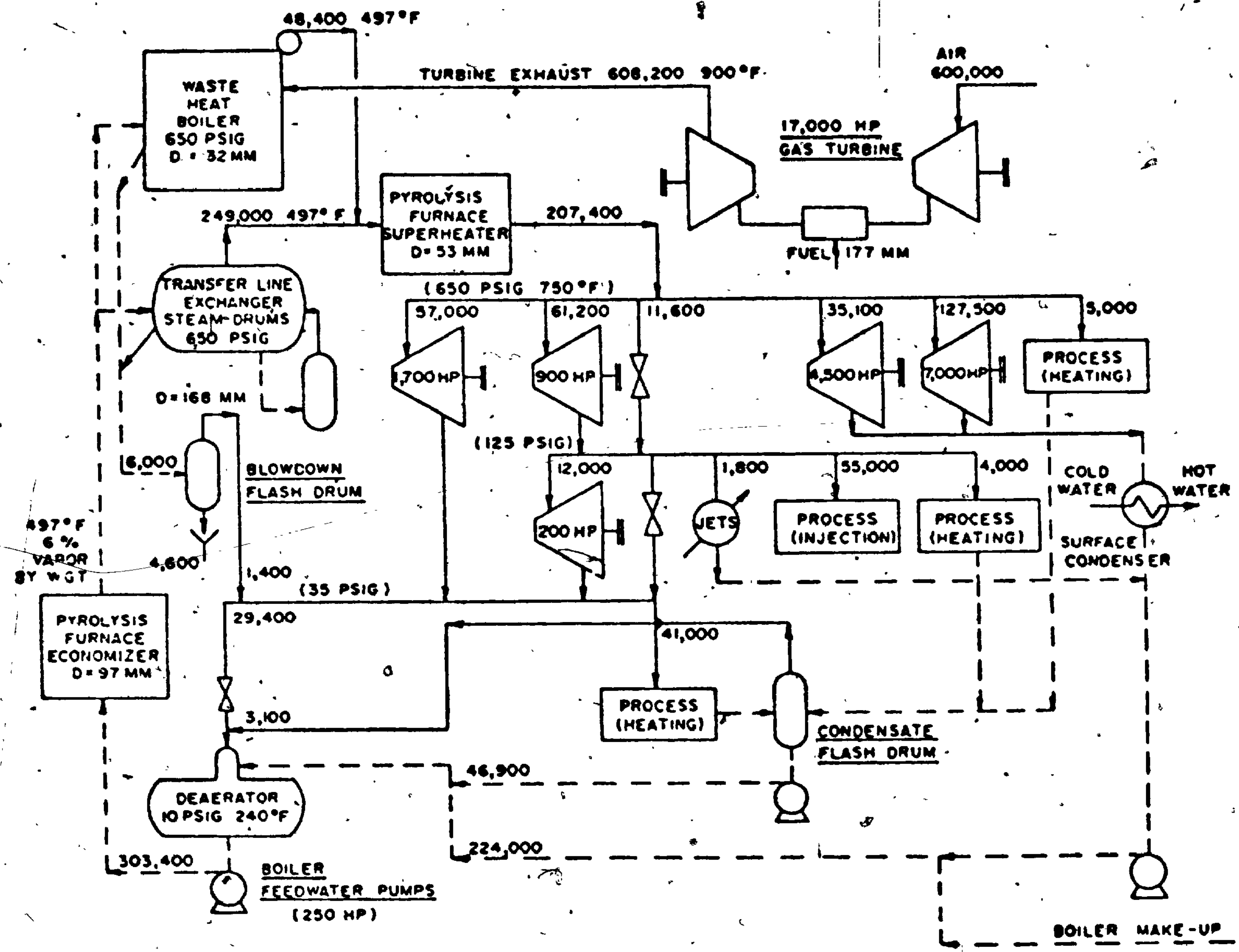


FIGURE 5.1 GAS TURBINE PLUS MEDIUM PRESSURE STEAM CYCLE

1. DRIVER HORSEPOWERS

A) PROCESS GAS COMPRESSOR	17,000 HP
B) PROPYLENE REFRIGERATION COMPRESSOR	17,000 HP
C) ETHYLENE REFRIGERATION COMPRESSOR	4,500 HP
D) PROCESS PUMPS	1,500 HP
E) COOLING WATER PUMPS	900 HP
F) BOILER FEEDWATER PUMPS - SET BY STEAM BALANCE	
G) MISC. SMALL DRIVERS (PUMPS, FANS, ETC.)	200 HP

2. PROCESS HEATING REQUIREMENTS (CONDENSATE RETURNED)

A) 600 PSIG STEAM	5,000 LBS./HR.
B) 125 PSIG STEAM	4,000 LBS./HR.
C) 35 PSIG STEAM	41,000 LBS./HR.
TOTAL HEATING STEAM	50,000 LBS./HR.

3. PROCESS STEAM REQUIREMENTS (NO CONDENSATE RETURNED)

A) STEAM TO PYROLYSIS FURNACES (125 PSIG)	55,000 LBS./HR.

4. HEAT AVAILABLE FROM PROCESS

A) HIGH LEVEL FROM FURNACE PROCESS EFFLUENT	168 MM BTU/HR.
B) WASTE HEAT FROM FURNACE FLUE GASES	150 MM BTU/HR.
OVER-ALL FURNACE EFFICIENCY = 85 PERCENT (LHV)	
C) LOW LEVEL HEAT	14 MM BTU/HR.

TABLE 5.1 UTILITY REQUIREMENTS FOR A 500,000,000 LBS/YR ETHYLENE PLANT

First of all Figure 5.1 has to be converted into the information flow diagram shown in Figure 5.2 which involves 69 streams and 38 units, for the preparation of the input to the computer executive system "ODES". Figure 5.2 becomes a "design" information flow diagram using the library modules of "ODES' and with the calculational modules and streams numbered. This conversion of figure is necessary for a consistent treatment of design information including the system configuration.

Then the next step is to make up the input data to "ODES" based on the information flow diagram Figure 5.2 and utility requirements listed in Table 5.1. The input form for "ODES' showing complete data format is presented in Appendix 6. The print-out of an input data set is shown in Listing 5.1 where basic input information consists of a network representation by the aid of keyword identifiers and stream numbers, stream properties and equipment parameters.

For example, the first line in the data set of network representation in Listing 5.1 reads as follows:

The keyword is 'MIXER1'

'Equipment'number is 1

Data length of 'Equipment' MIXER1 is 1

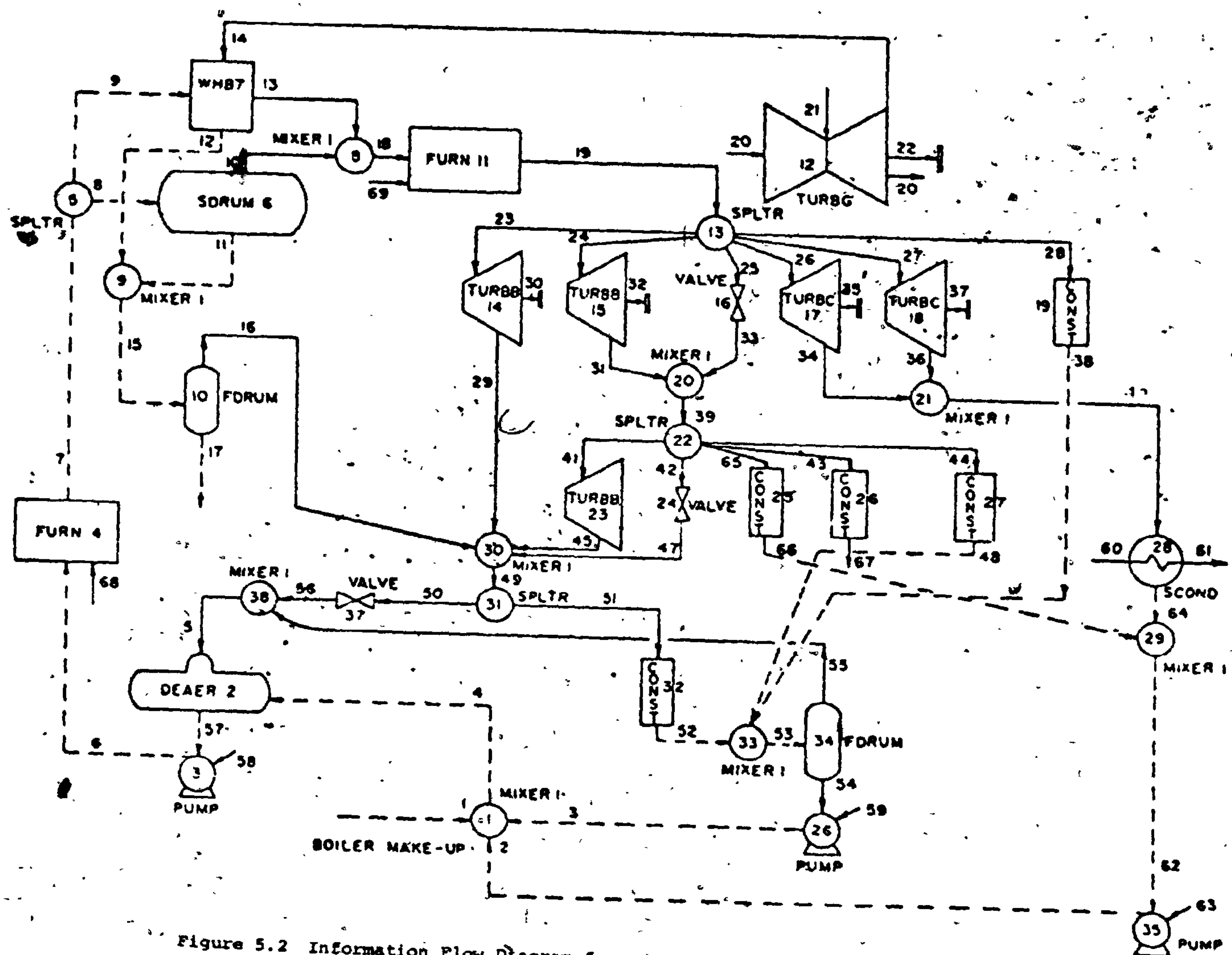


Figure 5.2 Information Flow Diagram for Figure 5.1

LISTING 5.1 INPUT DATA TO "ODES"

** ECONOMIC EVALUATION FOR UTILITY SYSTEM OF ETHYLENE PLANT **

ISIM	MAXST	MAXEQ	IUNIT
3	69	38	7

STREAM CODE

10	10	10	10	12	10	11	11	11	10	12	10	12	5	10	10
12	12	12	8	5	6	12	12	12	12	12	12	12	8	12	8
12	11	8	11	8	12	12	11	12	12	12	12	12	8	12	12
12	12	12	12	12	10	12	12	10	6	8	7	7	10	8	10
12	10	12	5	5											

SYSTEM NETWORK

			NO. OF I/O STREAMS		STREAM NO. RELATING TO 'EQUIPMENT'										
MIXER1	1	1	3	1	1	2	3	4	0	0	0	0	0	0
DEAER	2	1	2	1	4	5	57	0	0	0	0	0	0	0
PUMP	3	2	2	1	57	58	6	0	0	0	0	0	0	0
FURN	4	2	2	1	6	66	7	0	0	0	0	0	0	0
SPLTR	5	1	1	2	7	8	9	0	0	0	0	0	0	0
SDRUM	6	3	1	2	8	10	11	0	0	0	0	0	0	0
WHB	7	3	2	2	9	14	13	12	0	0	0	0	0	0
MIXER1	8	1	2	1	10	13	18	0	0	0	0	0	0	0
MIXER1	9	1	2	1	12	11	15	0	0	0	0	0	0	0
FDRUM	10	2	1	2	15	16	17	0	0	0	0	0	0	0
FURN	11	2	2	1	18	69	19	0	0	0	0	0	0	0
TURBG	12	4	2	3	21	20	22	14	20	0	0	0	0	0
SPLTR	13	1	1	6	19	23	24	25	26	27	28	0	0	0
TURBB	14	2	1	2	23	29	30	0	0	0	0	0	0	0
TURBB	15	2	1	2	24	31	32	0	0	0	0	0	0	0
VALVE	16	1	1	1	25	33	0	0	0	0	0	0	0	0
TURBC	17	2	1	2	26	34	35	0	0	0	0	0	0	0
TURBC	18	2	1	2	27	36	37	0	0	0	0	0	0	0

LISTING 5.1 - CONTINUED

CONST	19	3	1	1	28	38	0	0	0	0	0	0	0	0
MIXER1	20	1	2	1	31	33	39	0	0	0	0	0	0	0
MIXER1	21	1	2	1	34	36	40	0	0	0	0	0	0	0
SPLTR	22	1	1	5	39	41	42	65	43	44	0	0	0	0
TURBN	23	2	1	2	41	45	46	0	0	0	0	0	0	0
VALVE	24	1	1	1	42	47	0	0	0	0	0	0	0	0
CONST	25	3	1	1	65	66	0	0	0	0	0	0	0	0
CONST	26	3	1	1	43	67	0	0	0	0	0	0	0	0
CONST	27	3	1	1	44	48	0	0	0	0	0	0	0	0
SCOND	28	2	2	2	40	60	64	61	0	0	0	0	0	0
MIXER1	29	1	2	1	66	64	62	0	0	0	0	0	0	0
MIXER1	30	1	4	1	16	29	45	47	49	0	0	0	0	0
SPLTR	31	1	1	2	49	50	51	0	0	0	0	0	0	0
CONST	32	3	1	1	51	52	0	0	0	0	0	0	0	0
MIXER1	33	1	3	1	52	48	38	53	0	0	0	0	0	0
FDRUM	34	2	1	2	53	55	54	0	0	0	0	0	0	0
PUMP	35	2	2	1	62	63	2	0	0	0	0	0	0	0
PUMP	36	2	2	1	54	59	3	0	0	0	0	0	0	0
VALVE	37	1	1	1	50	56	0	0	0	0	0	0	0	0
MIXER1	38	1	2	1	56	55	5	0	0	0	0	0	0	0
**	0	0	0	0	0	0	0	0	0	0	0	0	0	0

STREAM DATA 23

STREAM	4	2	230.00	0	0.00	0	0.00	0	0.00
STREAM	6	2	240.00	3	685.00	0	0.00	0	0.00
STREAM	7	2	497.00	3	664.70	4	.06	0	0.00
STREAM	10	2	497.00	3	664.70	0	0.00	0	0.00
STREAM	13	2	497.00	3	664.70	0	0.00	0	0.00
STREAM	19	2	750.00	3	664.70	0	0.00	0	0.00
STREAM	29	3	49.70	0	0.00	0	0.00	0	0.00
STREAM	30	1	1270.00	0	0.00	0	0.00	0	0.00
STREAM	31	3	139.70	0	0.00	0	0.00	0	0.00
STREAM	32	1	671.00	0	0.00	0	0.00	0	0.00
STREAM	34	3	3.50	0	0.00	0	0.00	0	0.00
STREAM	35	1	3360.00	0	0.00	0	0.00	0	0.00
STREAM	36	3	3.50	0	0.00	0	0.00	0	0.00

LISTING 5.1 - CONTINUED

STREAM	37	1	12700.00	0	0.00	0	0.00	0	0.00
STREAM	39	2	550.00	3	139.70	0	0.00	0	0.00
STREAM	45	3	49.70	0	0.00	0	0.00	0	0.00
STREAM	46	1	149.00	0	0.00	0	0.00	0	0.00
STREAM	62	3	14.70	0	0.00	0	0.00	0	0.00
STREAM	2	2	200.00	3	100.00	0	0.00	0	0.00
STREAM	54	3	20.00	0	0.00	0	0.00	0	0.00
STREAM	3	3	100.00	0	0.00	0	0.00	0	0.00
STREAM	5	2	260.00	3	25.00	0	0.00	0	0.00
STREAM	57	2	240.00	3	24.70	0	0.00	0	0.00
EQUIPMENT DATA		22							
DEAER	2	1	18.00	0	0.00	0	0.00	0	0.00
PUMP	3	1	15.00	0	0.00	0	0.00	0	0.00
FURN	4	2	97.00	1	66.00	0	0.00	0	0.00
SDRUM	6	2	.98	3	168.00	1	253.00	0	0.00
WHB	7	2	.96	3	32.00	1	73.00	0	0.00
FDRUM	10	2	.23	1	1.00	0	0.00	0	0.00
FURN	11	2	53.00	1	44.00	0	0.00	0	0.00
TURBG	12	2	12700.00	3	8.95	1	850.00	0	0.00
TURBB	14	1	15.00	0	0.00	0	0.00	0	0.00
TURBB	15	1	9.00	0	0.00	0	0.00	0	0.00
TURBC	17	1	90.00	0	0.00	0	0.00	0	0.00
TURBC	18	1	240.00	0	0.00	0	0.00	0	0.00
CONST	19	2	5.00	0	0.00	0	0.00	0	0.00
TURBB	23	1	6.00	0	0.00	0	0.00	0	0.00
CONST	25	2	1.80	0	0.00	0	0.00	0	0.00
CONST	26	2	55.00	0	0.00	0	0.00	0	0.00
CONST	27	2	4.00	0	0.00	0	0.00	0	0.00
SCOND	28	2	50.00	1	90.00	0	0.00	0	0.00
CONST	32	2	41.00	0	0.00	0	0.00	0	0.00
FDRUM	34	2	.06	1	5.00	0	0.00	0	0.00
PUMP	35	1	1.50	0	0.00	0	0.00	0	0.00
PUMP	36	1	1.50	0	0.00	0	0.00	0	0.00

'MIXER1' has three input streams (stream numbers 1, 2 and 3) and

One output stream (stream number 4)

As for stream data, the first line reads as follows:

Stream number 4 has a temperature of 230°F as the second attribute

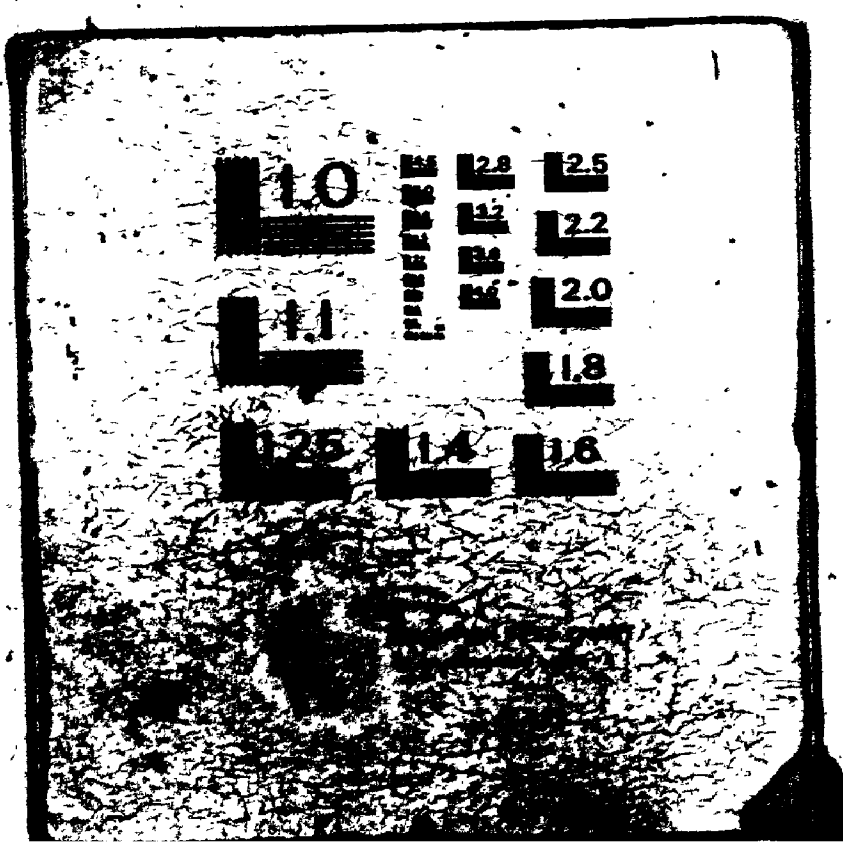
Although this data preparation appears cumbersome, it is not only straight forward once one gets used to it but also helpful to grasp basic conditions clearly, because of the clear data input system relating to the program data structure.

Once the necessary information is gathered, the next step is the execution of computation which includes balance formulation, equipment sizing and cost estimation according to the simulation level.

Computational results are shown in Listing 5.2.

It should be noted that in the paper by Arstein and O'Connell the construction of the balances was done by a trial and error fashion whereby certain quantities are assumed and then corrected in subsequent calculations by hand, whereas in the use of "QDES" engineers have no need to worry about the balance calculation because it can be automatically done by the systematic way which was described in Chapter 2.

2



LISTING 5.2 COMPUTATIONAL RESULTS FOR THE FIRST EXAMPLE

TOTAL ERECTED COST	4130.3	$\times 10^3$ \$
OPERATING COST	806.9	
FUEL GAS	593.0	
FUEL OIL	0.0	
COOLING WATER	121.6	
ELECTRICITY	16.7	
INDUSTRIAL WATER	0.0	
BOILER FEED WATER	5.7	
LABOR $ SUPERVISION	70.0	

LISTING 5.2 - CONTINUED

STREAM DATA FLOW RATE

1	56.65	0.00	0.00	0.00
2	122.45	200.00	100.00	0.00
3	47.00	0.00	100.00	0.00
4	226.10	230.00	0.00	0.00
5	76.05	260.00	25.00	0.00
6	302.15	240.00	685.00	0.00
7	302.15	497.00	664.70	.06
8	253.81	497.00	664.70	.06
9	48.34	497.00	664.70	.06
10	248.73	497.00	664.70	0.00
11	5.08	0.00	0.00	0.00
12	.97	0.00	0.00	0.00
13	47.38	497.00	664.70	0.00
14	610.64	0.00	0.00	0.00
15	6.04	0.00	0.00	0.00
16	1.39	0.00	0.00	0.00
17	4.65	0.00	0.00	0.00
18	296.11	0.00	0.00	0.00
19	296.11	750.00	664.70	0.00
20	5465.19			
21	610.64	0.00	0.00	0.00
22	12700.00			
23	31.08	750.00	664.70	0.00
24	33.86	750.00	664.70	0.00
25	105.52	750.00	664.70	0.00
26	28.01	750.00	664.70	0.00
27	92.64	750.00	664.70	0.00
28	5.00	750.00	664.70	0.00
29	31.08	0.00	49.70	0.00
30	1270.00			
31	33.86	0.00	139.70	0.00
32	671.00			
33	105.52	0.00	0.00	0.00
34	28.01	39.20	3.50	0.00
35	3360.00			

LISTING 5.2 - CONTINUED

36	92.64	39.20	3.50	0.00
37	12700.00			
38	5.00	0.00	0.00	0.00
39	139.38	550.00	139.70	0.00
40	120.65	0.00	0.00	0.00
41	12.20	550.00	139.70	0.00
42	66.38	550.00	139.70	0.00
43	55.00	550.00	139.70	0.00
44	4.00	550.00	139.70	0.00
45	12.20	0.00	49.70	0.00
46	149.00			
47	66.38	0.00	0.00	0.00
48	4.00	0.00	0.00	0.00
49	111.05	0.00	0.00	0.00
50	70.05	0.00	0.00	0.00
51	41.00	0.00	0.00	0.00
52	41.00	0.00	0.00	0.00
53	50.00	0.00	0.00	0.00
54	47.00	0.00	20.00	0.00
55	3.00	0.00	0.00	0.00
56	70.05	0.00	0.00	0.00
57	302.15	240.00	24.70	0.00
58	264.52			
59	4.99			
60	6032.50	0.00	0.00	0.00
61	6032.50	0.00	0.00	0.00
62	122.45	0.00	14.70	0.00
63	13.85			
64	120.65	0.00	0.00	0.00
65	1.80	550.00	139.70	0.00
66	1.80	0.00	0.00	0.00
67	55.00	0.00	0.00	0.00
68	392.16	0.00	0.00	0.00
69	214.27	0.00	0.00	0.00

LISTING 5.2 - CONTINUED

EQUIPMENT DATA	38	COST			
MIXER1	1	0.00			
DEAER	2	18.00			
PUMP	3	302.15	302.15		
FURN	4	66.00	97.00		
SPLTR	5	0.00			
SDRUM	6	253.00	.98	168.00	
WHB	7	73.00	.98	32.00	
MIXER1	8	0.00			
MIXER1	9	0.00			
FDRUM	10	1.00	.23		
FURN	11	44.00	53.00		
TURBG	12	850.00	12700.00	8.95	0.00
SPLTR	13	0.00			
TURBB	14	15.00	1270.00		
TURBB	15	9.00	671.00		
VALVE	16	0.00			
TURBC	17	90.00	3360.00		
TURBC	18	240.00	12700.00		
CONST	19	0.00	5.00	0.00	
MIXER1	20	0.00			
MIXER1	21	0.00			
SPLTR	22	0.00			
TURBB	23	6.00	149.00		
VALVE	24	0.00			
CONST	25	0.00	1.80	0.00	
CONST	26	0.00	55.00	0.00	
CONST	27	0.00	4.00	0.00	
SCOND	28	90.00	50.00		
MIXER1	29	0.00			
MIXER1	30	0.00			
SPLTR	31	0.00			
CONST	32	0.00	41.00	0.00	
MIXER1	33	0.00			
FDRUM	34	5.00	.06		
PUMP	35	1.50	122.45		
PUMP	36	1.50	47.00		
VALVE	37	0.00			
MIXER1	38	0.00			

Computation time for this simulation was about 0.1 minutes on a CDC CYBER 73.

As seen in Listing 5.2, all basic information for the case evaluation is listed as final results, e.g. stream properties, equipment parameters and aggregate utility costs. It is quite easy to see which item is significant in an economic evaluation. It is also a great advantage that once a basic case is set up, an evaluation of an alternate case such as modifying a basic configuration, or changing parameters can be achieved, promptly and with little effort.

5.2 Optimal Design of a Steam and Power Plant

The second example is an optimal design of a steam and power plant for a chemical complex for which the information flow diagram is shown in Figure 5.3, which involves 70 streams and 32 units.

Again this section is to demonstrate how "ODES" can be utilized to perform an optimal design of an arbitrary energy system.

Table 5.2 shows the demands required for the energy system given in Figure 5.3 and the venture cost including the fixed cost and operating cost is taken as a criterion for the optimal design.

DEMAND ITEM	STREAM NUMBER	DEMANDS
ELECTRICITY	61	26140 KWH
HIGH PRESSURE STEAM	25	75.6 MLB/HR
MEDIUM PRESSURE STEAM	63	1100 "
LOW PRESSURE STEAM	69	139.3 "
CONDENSATE	66	26.3 "
BY-PASS STEAM	26	> 410 "
BY-PASS STEAM	65	> 10 "
INTERNAL POWER DEMAND		
1	4	TO BE CALCULATED FOR PUMP 2
" 2	16	" PUMP 4
" 3	46	" BOILER 8
EXTERNAL POWER DEMAND		
1	42	456 KWH
" 2	44	368 "
" 3	48	184 "
" 4	50	1288 "
" 5	52	257.5 "
" 6	54	225 "

TABLE 5.2 STEAM AND POWER DEMANDS

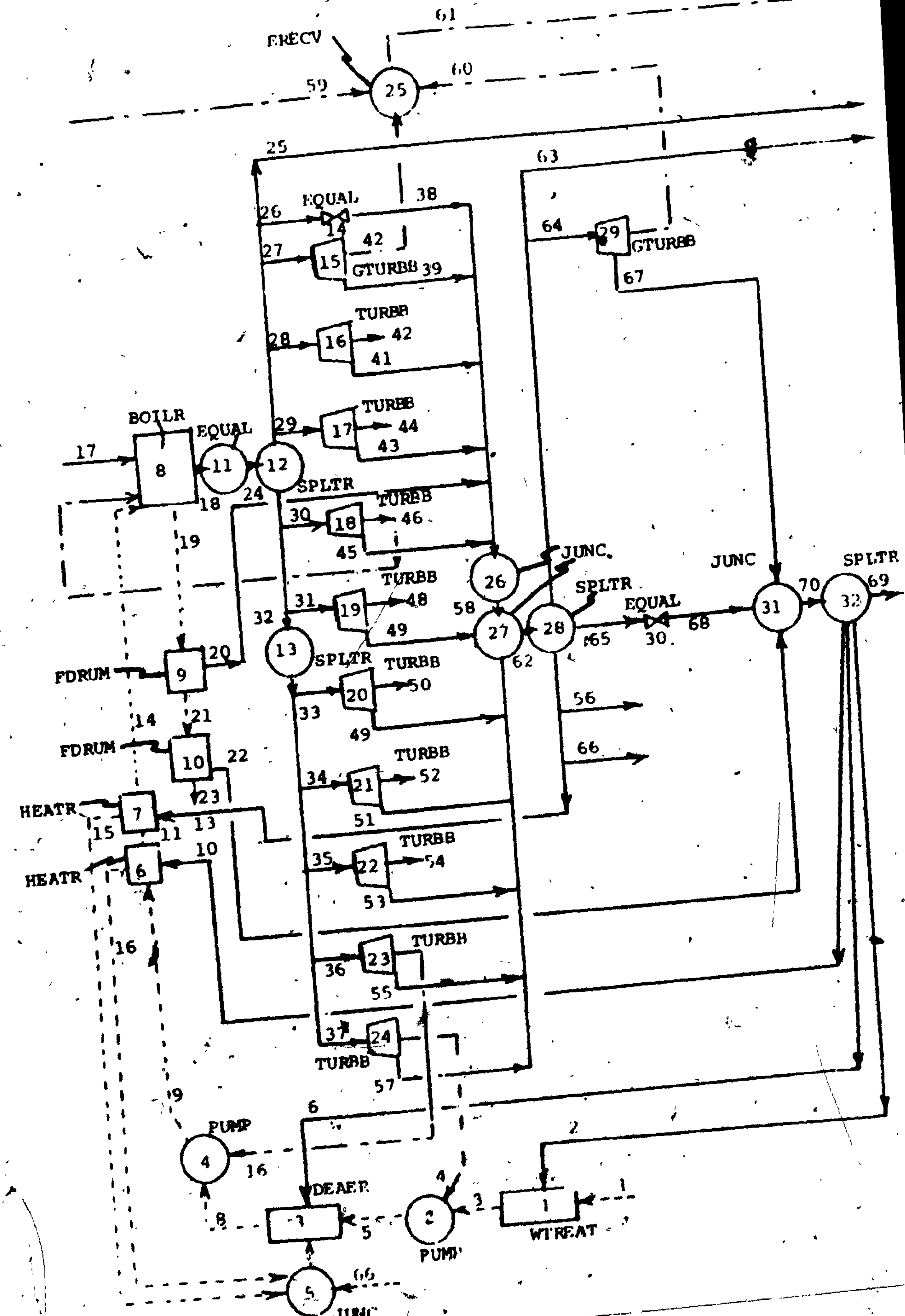


Figure 5.3   Information Flow Diagram For a Steam and Power Plant

In "ODES" the "users" choice of independent variables for the parameter optimization is important. Although engineers can choose independent design variables making use of their experience, they must make a final choice of independent variables taking care to satisfy certain computational needs. For instance, in this problem, choices of by-pass flow rates and M.P. steam loss as independent variables have the advantage of avoiding a negative energy balance which will happen frequently, otherwise. Table 5.3 shows a final choice of independent variables and constraints. In choosing design variables, one of the very helpful features of "ODES" is determination of the degree of freedom in the design system. In other words, since the balancing of the variables with the equations representing the system takes place in "ODES", it is clearly diagnosed whether extra variables were specified or not enough variables were specified.

The program input data can be prepared easily based on the information flow diagram and the demands required, referring to the input format instructions in the same way as the preceding example.

Computational results for this example are summarized in Listing 5.3 and Table 5.4. In particular the cost summary is shown in Table 5.4 where the optimal case 1 is compared with two non-optimal cases, i.e., case 2,

I.V. NO.	VARIABLE DESCRIPTION	LOWER BOUND	UPPER BOUND
1	TEMPERATURE OF STREAM 11	260°F	300
2	TEMPERATURE OF STREAM 14	350°F	430
3	TEMPERATURE OF STREAM 18	600°F	900
4	PRESSURE OF STREAM 18	450 PSIA	900
5	BY-PASS FLOW RATE OF STREAM 26	410 MLB/HR	500
6	BY-PASS FLOW RATE OF STREAM 65	10 MLB/HR	100
7	M.P. STEAM LOSS (STREAM 56)	0 MLB/HR	30
CONSTRAINTS NO.			
1	TEMPERATURE OF STREAM 18 >	TEMPERATURE OF STREAM 14	
2	TEMPERATURE OF STREAM 14 >	TEMPERATURE OF STREAM 11	
3	PRESSURE OF STREAM 18 =	PRESSURE OF STREAM 9-25.3	
4	PRESSURE OF STREAM 18 =	PRESSURE OF STREAM 24+40.0	
5	TEMPERATURE OF STREAM 18 =	TEMPERATURE OF STREAM 24	

TABLE 5.3 INDEPENDENT VARIABLES AND CONSTRAINTS

LISTING 5.3 COMPUTATIONAL RESULTS FOR THE SECOND EXAMPLE

STREAM DATA	FLOW RATE	TEMP.	PRESS.	
1	1429.48	0.00	0.00	0.00
2	114.36	300.00	29.70	0.00
3	1429.48	0.00	14.70	0.00
4	66.33			
5	1429.48	170.00	49.70	0.00
6	105.84	300.00	29.70	0.00
7	244.07	300.00	0.00	0.00
8	1779.39	250.00	29.70	0.00
9	1779.39	250.00	510.79	0.00
10	64.56	300.00	29.70	0.00
11	1779.39	283.42	0.00	0.00
12	64.56	286.00	0.00	0.00
13	153.22	500.00	179.70	0.00
14	1779.39	361.55	0.00	0.00
15	153.22	365.00	0.00	0.00
16	1134.95			
17	97.00	0.00	0.00	0.00
18	1690.42	603.02	465.49	0.00
19	88.97	0.00	0.00	0.00
20	84.52	500.00	179.70	0.00
21	4.45	0.00	0.00	0.00
22	4.23	300.00	29.70	0.00
23	.22	0.00	0.00	0.00
24	1690.42	603.02	445.49	0.00
25	75.60	603.02	445.49	0.00
26	410.00	603.02	445.49	0.00
27	690.72	603.02	445.49	0.00
28	33.90	603.02	445.49	0.00
29	27.36	603.02	445.49	0.00
30	218.25	603.02	445.49	0.00
31	13.68	603.02	445.49	0.00
32	220.91	603.02	445.49	0.00
33	95.75	603.02	445.49	0.00

NOTE - VALUES UNDERLINED ARE FINAL RESULTS OF INDEPENDENT VARIABLES

LISTING 5.3 - CONTINUED

34	19.14	603.02	445.49	0.00
35	16.73	603.02	445.49	0.00
36	84.37	603.02	445.49	0.00
37	4.93	603.02	445.49	0.00
38	410.00	500.00	179.70	0.00
39	690.72	401.48	179.70	0.00
40	12702.65			
41	33.90	401.48	179.70	0.00
42	456.00			
43	27.36	401.48	179.70	0.00
44	368.00			
45	218.25	401.48	179.70	0.00
46	2935.99			
47	13.68	401.48	179.70	0.00
48	184.00			
49	95.75	401.48	179.70	0.00
50	1288.00			
51	19.14	401.48	179.70	0.00
52	257.50			
53	16.73	401.48	179.70	0.00
54	225.00			
55	84.37	401.48	179.70	0.00
56	.00	500.00	179.70	0.00
57	4.93	401.48	179.70	0.00
58	1464.75	500.00	179.70	0.00
59	665.92			
60	12771.43			
61	26140.00			
62	1699.34	500.00	179.70	0.00
63	1100.00	500.00	179.70	0.00
64	409.83	500.00	179.70	0.00
65	10.00	500.00	179.70	0.00
66	26.30	500.00	179.70	0.00
67	409.83	173.62	29.70	0.00
68	10.00	300.00	29.70	0.00
69	139.30	300.00	29.70	0.00
70	424.05	300.00	29.70	0.00

LISTING 5.3 - CONTINUED

EQUIPMENT DATA	32	COST	CAPACITY				
WTREAT	1	16.72	1429.48	0.00	.08		
PUMP	2	5.63	1429.48	0.00			
DEAERI	3	11.95	1779.39	0.00			
PUMP	4	6.42	1779.39	0.00			
JUNC1	5	0.00					
HEATR	6	6.84	142.08	0.00	500.00		
HEATR	7	1.84	33.10	0.00	500.00		
BOILR	8	10513.78	1690.42	0.00	.05	18900.00	1.65
FDRUM	9	0.00	0.00	0.00	.05		
FDRUM	10	0.00	0.00	0.00	.05		
EQUAL	11	0.00					
SPLTR	12	0.00					
SPLTR	13	0.00					
EQUAL	14	0.00					
GTURBB	15	1198.55	12702.65	0.00	.02		
TURBB	16	38.07	456.00	0.00			
TURBB	17	32.77	368.00	0.00			
TURBB	18	140.20	2935.99	0.00			
TURBB	19	20.17	184.00	0.00			
TURBB	20	78.75	1288.00	0.00			
TURBB	21	25.52	257.50	0.00			
TURBB	22	23.22	225.00	0.00			
TURBB	23	72.08	1134.95	0.00			
TURBB	24	9.88	66.33	0.00			
ERECIV	25	5208.67	26140.00	0.00			
JUNC	26	0.00					
JUNC	27	0.00					
SPLTR	28	0.00					
GTURBB	29	1203.22	12771.43	0.00	.02		
EQUAL	30	0.00					
JUNC	31	0.00					
SPLTR	32	0.00					

COST SUMMARY x $1000

	CASE 1	CASE 2	CASE 3
OBJECTIVE FUNCTION	-52787.1	-53796.4	-57324.9
TOTAL ERECTED COST	37228.5	37468.0	40684.3
OPERATING COST	32497.6	33376.3	35152.0
FUEL GAS	0.0	0.0	0.0
FUEL OIL	32289.1	33243.7	35019.3
COOLING WATER	0.0	0.0	0.0
ELECTRICITY	75.8	.0	.0
INDUSTRIAL WATER	62.6	62.6	62.6
LABOR $ SUPERVISION	70.0	70.0	70.0

NOTE: MAJOR DIFFERENCES OF CONDITIONS AMONG THREE CASES

	OPTIMAL CASE	CASE 2	CASE 3	
TEMPERATURE AT BOILER OUTLET	603.0	650	750	°F
PRESSURE AT BOILER OUTLET	485.5	474.7	574.7	PSIA

TABLE 5.4 COST SUMMARY FOR EVERY STEP OF THE TWO LEVEL COORDINATION

where the conditions were estimated from those of a similar existing plant, and case 3, where the conditions were arbitrarily taken within the possible variable ranges. The convergence of major variables in the optimization is shown in Figure 5.4. The stopping criterion for convergence is based on the relative change of objective functions in the complex which will be described in Appendix 6, and the tolerance of 0.001 was used for the criterion in this case.

Computation time was about a half minute on a CDC CYBER 73 for this problem. It is noteworthy that the computation time for re-solving the simultaneous equation set was reduced to one-thirtieth by extracting the essential operations and computing them after the first step of iteration in the optimization.

5.3 Summary

The computer executive system "ODES" was tested for an economic evaluation of gas turbine plus medium pressure steam cycle for an ethylene plant (38 units and 69 streams) and an optimal design of steam and power plant for a petrochemical complex (32 units and 70 streams) with 7 independent variables and 16 inequality constraints.

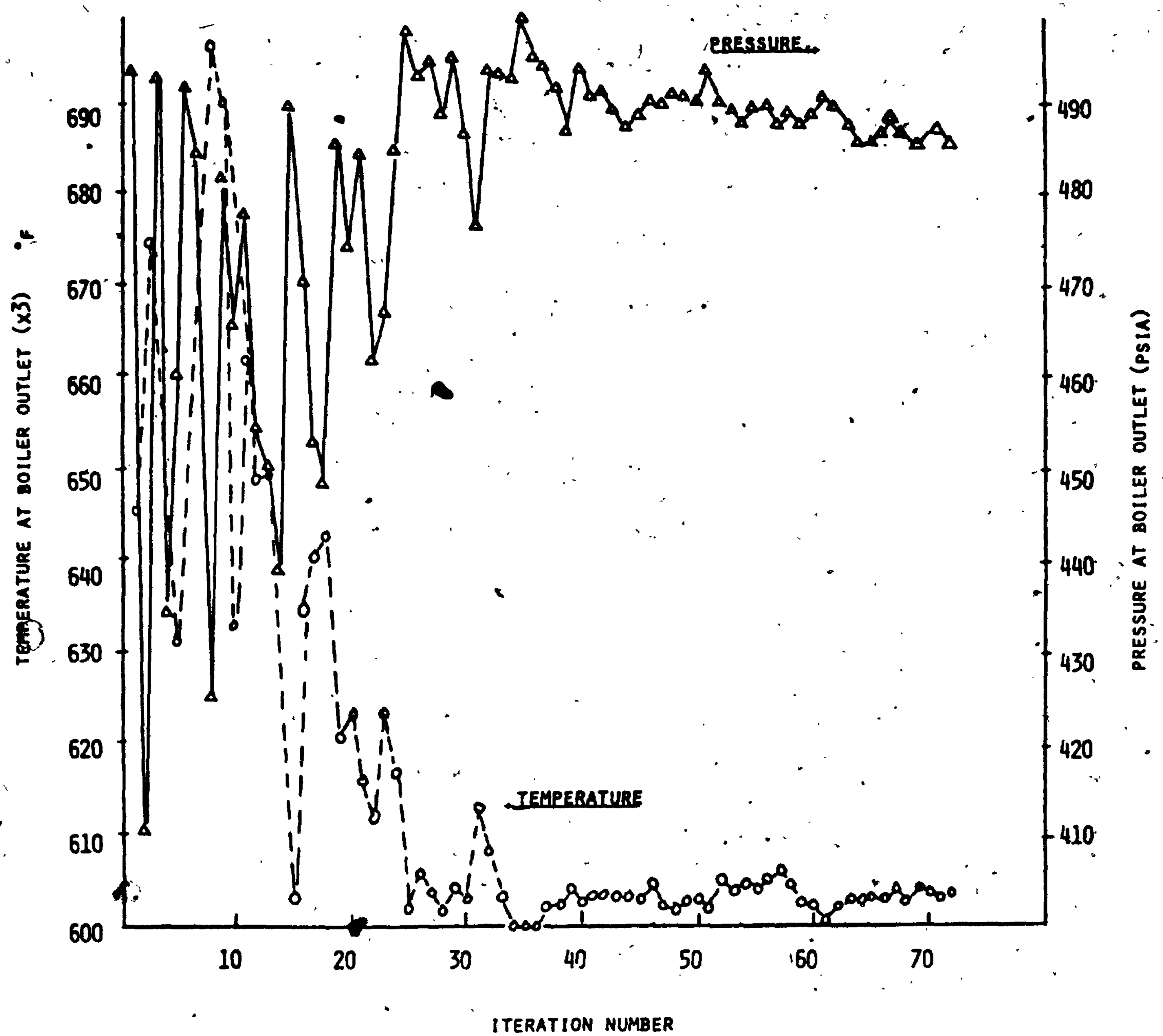


Figure 5.4 Convergence of Major Variables

The system "ODES" also can be applied to an evaluation of a system other than a steam and power plant, e.g. heat recovery systems, refrigeration systems and so on.

CHAPTER 6

APPLICATIONS FOR OPTIMAL EXPANSION OF A STEAM AND POWER PLANT

6.1 Introduction

In chapter 3, an expansion problem of a steam and power plant was defined as a synthesis problem with constraints, a linear programming technique was proposed as a solution technique for a synthesis problem and the computer executive system "OPES" was developed for formulating the LP problem automatically.

In this chapter, the methodology developed in chapter 3 is applied to two examples, i.e., simple and complex cases. Again the practical answers to be expected by solving the LP problem are: What units should be expanded? When? What is the optimal energy balance?

6.2 Simple Example

Energy demands for three periods, existing plant, capacities and operating conditions are given directly in Figure 6.1 for a simple steam and power plant. This information is transformed into input data to the computer executive system "OPES" by referring to the input format

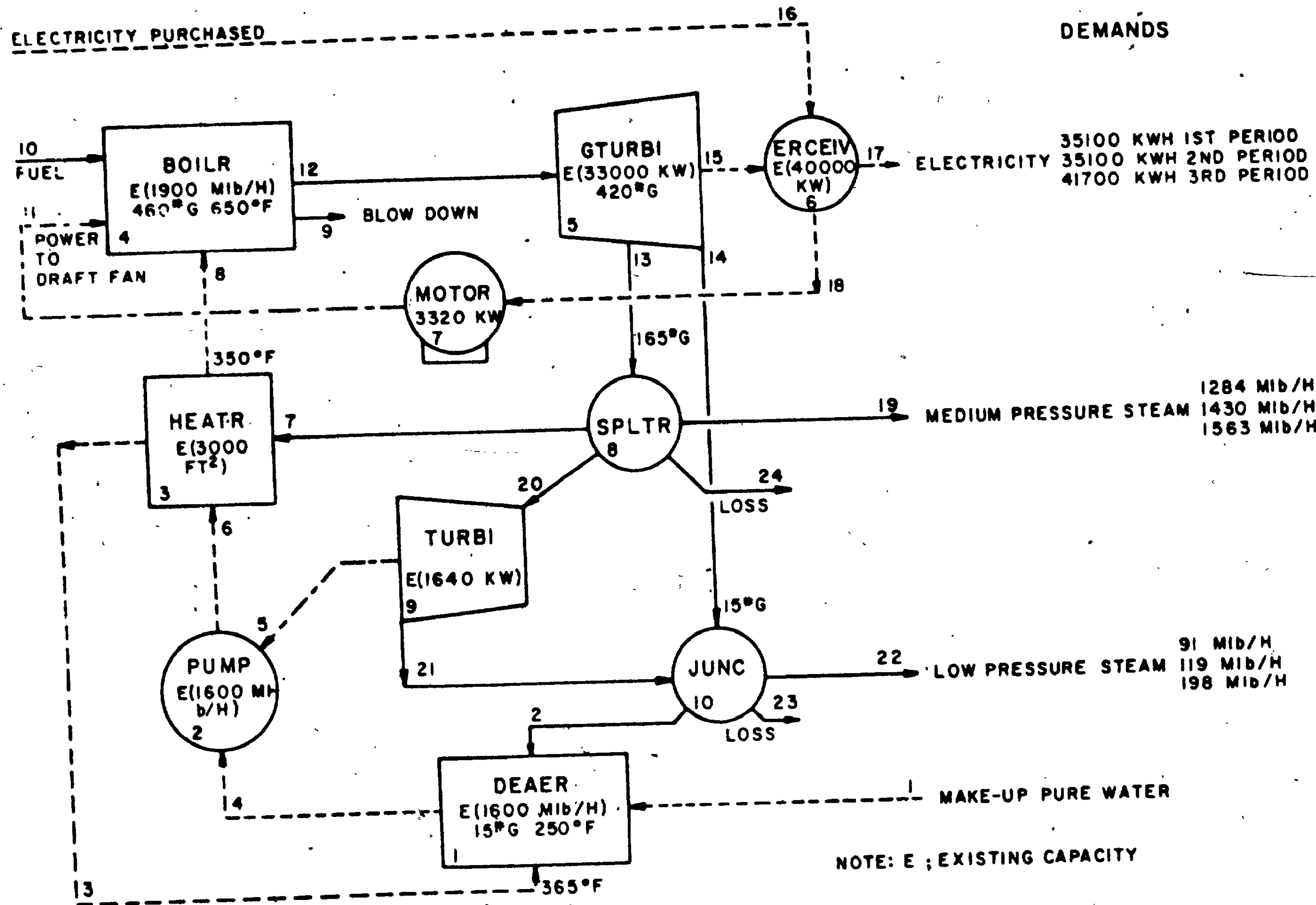


FIGURE 6.1 A SIMPLE STEAM AND POWER SYSTEM

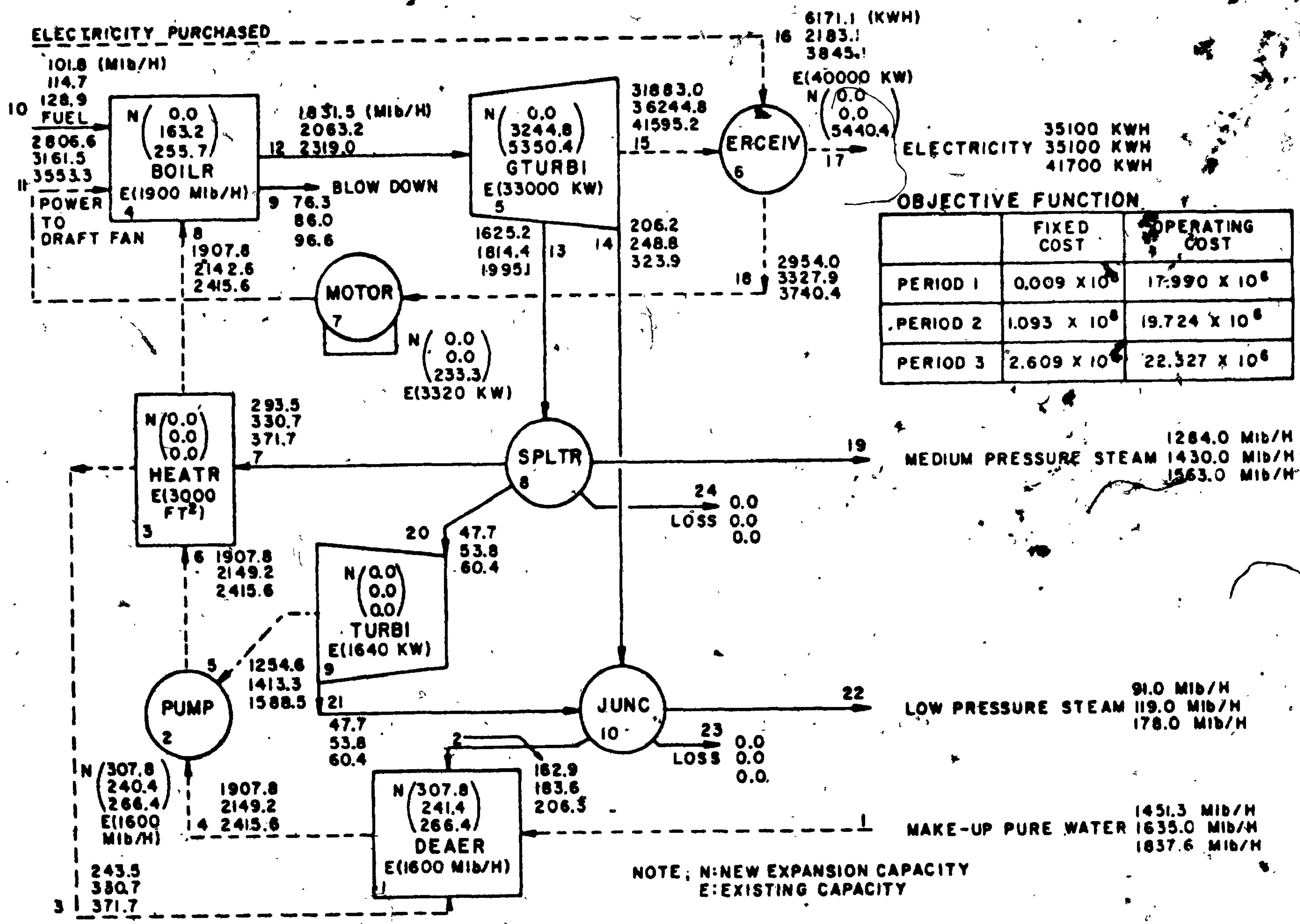
instructions attached in Appendix 8. The print-out of the input data set was shown in Listing 3.1 where basic information comprise a network representation of an existing plant by the aid of keyword identifiers and stream numbers, stream properties including demand specifications, and equipment parameters including plant capacity.

After LP formulation by "OPES" the problem was solved by using MPSX on an IBM 370. Computation time was 0.1 min. for LP formulation by CDC CYBER 73 and 0.1 min. for LP solving by the IBM 370/165.

Numerical results for a standard case are summarized in Figure 6.2. The optimal energy balance for current demands and the result for the case with a high price of purchased electricity are presented in Figures 6.3 and 6.4 for the purpose of comparison with the standard case.

The results show that for current demand requirements, existing facilities have enough capacity, except for the deaerator and the boiler feedwater (BFW) pump as seen in Figure 6.3.

For future predicted demand, the suggested expansions are as follows and as can be seen in Figure 6.2 (although more care would have to be taken in actual practice for a final choice of an expansion plan, as described in the following case, with longer periods considered):



OBJECTIVE FUNCTION

	FIXED COST	OPERATING COST
PERIOD 1	0.009×10^6	17.990×10^6
PERIOD 2	1.093×10^6	19.724×10^6
PERIOD 3	2.609×10^6	22.327×10^6

Figure 6.2 An Optimal Expansion Plan for Future Demands Predicted

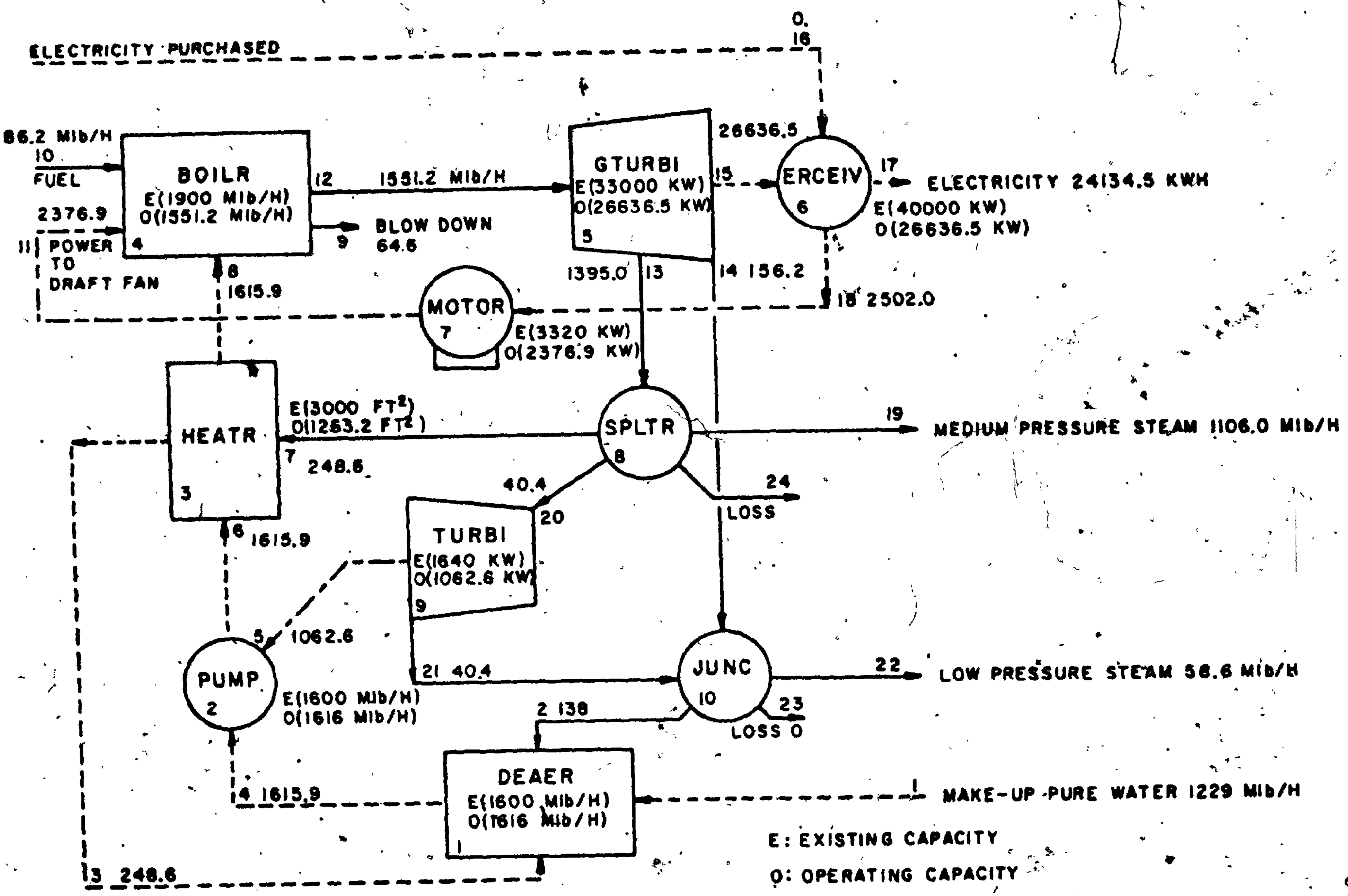


Figure 6.3 An Optimal Energy Balance for Current Demands

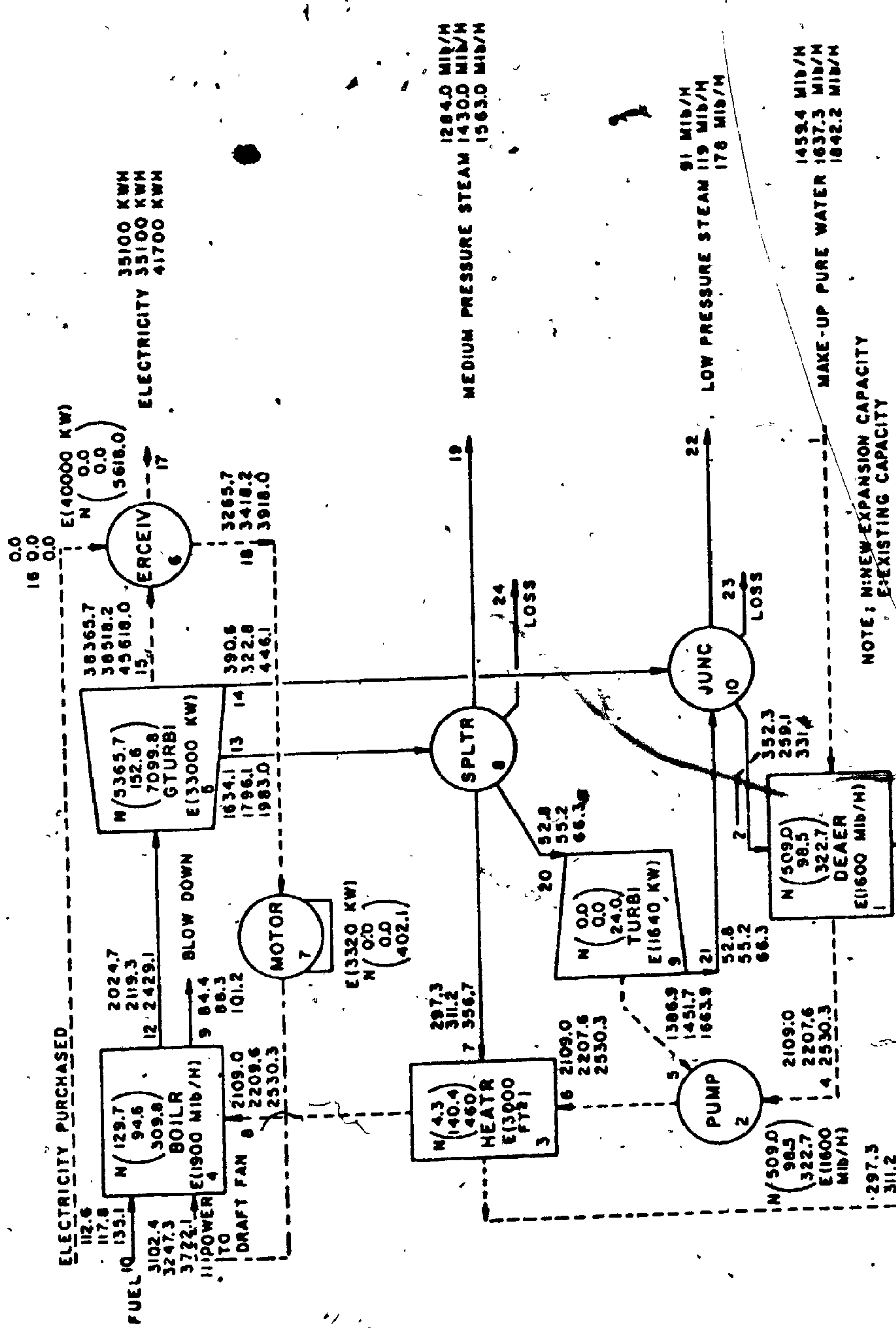


Figure 6.4 An Expansion Plan for Future Demands Predicted, with High Price of Purchased Electricity

- The deaerator and the BFW pump must be expanded during the first period and further expanded in succeeding periods.

- Boiler and generator-turbine units must be added during the second and third period expansion.

- Motors for the boiler draft fan and the power system capacity must be increased in power in the third period.

Moreover, for the case with a high price of purchased electricity seen in Figure 6.4, more expansion of generated electricity should be planned (with no further purchase of outside electricity).

The same example was solved for energy demands over 13 periods. Computation time was 0.2 min. for LP formulation and 1 min. for LP solving. Results are summarized in Table 6.1.

From a practical point of view, too small an expansion rate and consecutive expansion, such as seen in Table 6.1 would not be desirable. Therefore, the original solution must be modified by allowing early installation of larger capacities and the propriety of the modification should be confirmed by running the case where additional constraints are imposed for the minimum capacity which may be installed.

	UNIT 1 (DEAERATOR)	UNIT 2 (B.F.W. PUMP)	UNIT 3 (STAGE. HEATER)	UNIT 4 (BOILER)	UNIT 5 (GENERATOR TURBINE)	UNIT 6 (POWER RECEIVING – DISTRIBUTING STATION)	UNIT 7 (MOTOR)	UNIT 9 (TURBINE)
EXISTING CAPACITY	Mlb/hr 1600.	Mlb/hr 1600.	ft^2 3000.	Mlb/hr 1900.	KW 33000.	KW 40000.	KW 3320.	KW 1640.
PERIOD 1	369.7	369.7	-	-	-	-	-	-
PERIOD 2	-	-	-	-	-	-	-	-
PERIOD 3	-	-	-	-	-	-	-	-
PERIOD 4	16.9	16.9	-	7.1	-	-	-	-
PERIOD 5	-	-	-	-	-	-	-	-
PERIOD 6	1.2	1.2	-	1.2	280.0	-	-	-
PERIOD 7	-	-	-	-	-	-	-	-
PERIOD 8	-	-	-	-	-	-	-	-
PERIOD 9	-	-	-	-	-	-	-	-
PERIOD 10	-	-	-	-	-	-	-	-
PERIOD 11	-	-	-	-	-	-	-	-
PERIOD 12	161.4	161.4	54.7	154.9	2964.9	-	-	-
PERIOD 13	266.4	266.4	378.6	255.7	5350.	5440.4	233.3	-

TABLE 6.1 AN EXPANSION PLAN FOR 13 PERIODS

6.3 Complex Example

A realistic steam and power plant given in Figure 6.5 was taken as the second example, although only one period was taken into consideration for demonstration, and different cases were compared as possible situations. The results of parametric studies are shown in part in Tables 6.2 and 6.3. Case 1 is where basic cost data was used. Case 2 is where high pressure steam was purchased from outside instead of generating steam by existing in-plant facilities when the fuel cost was relatively expensive. Case 3 is where existing condensing turbines (equipment nos. 14 and 25) were fully operated and it was recommended that one unit be added to meet the increased electricity demand when the demand for medium-pressure steam decreased. In addition, the price of purchased electricity was considerably more expensive when compared with case 1.

The LP problem for this example contains 143 equations and 454 coefficients just for one time period. The problem is too huge to solve when the time periods become long. However, the total problem need not be solved at one time, as described below.

The relation linking multi-time periods is not very strong, especially because it is merely the relation between existing capacity and new capacity to be installed and only the capacity to be expended is involved in the objective

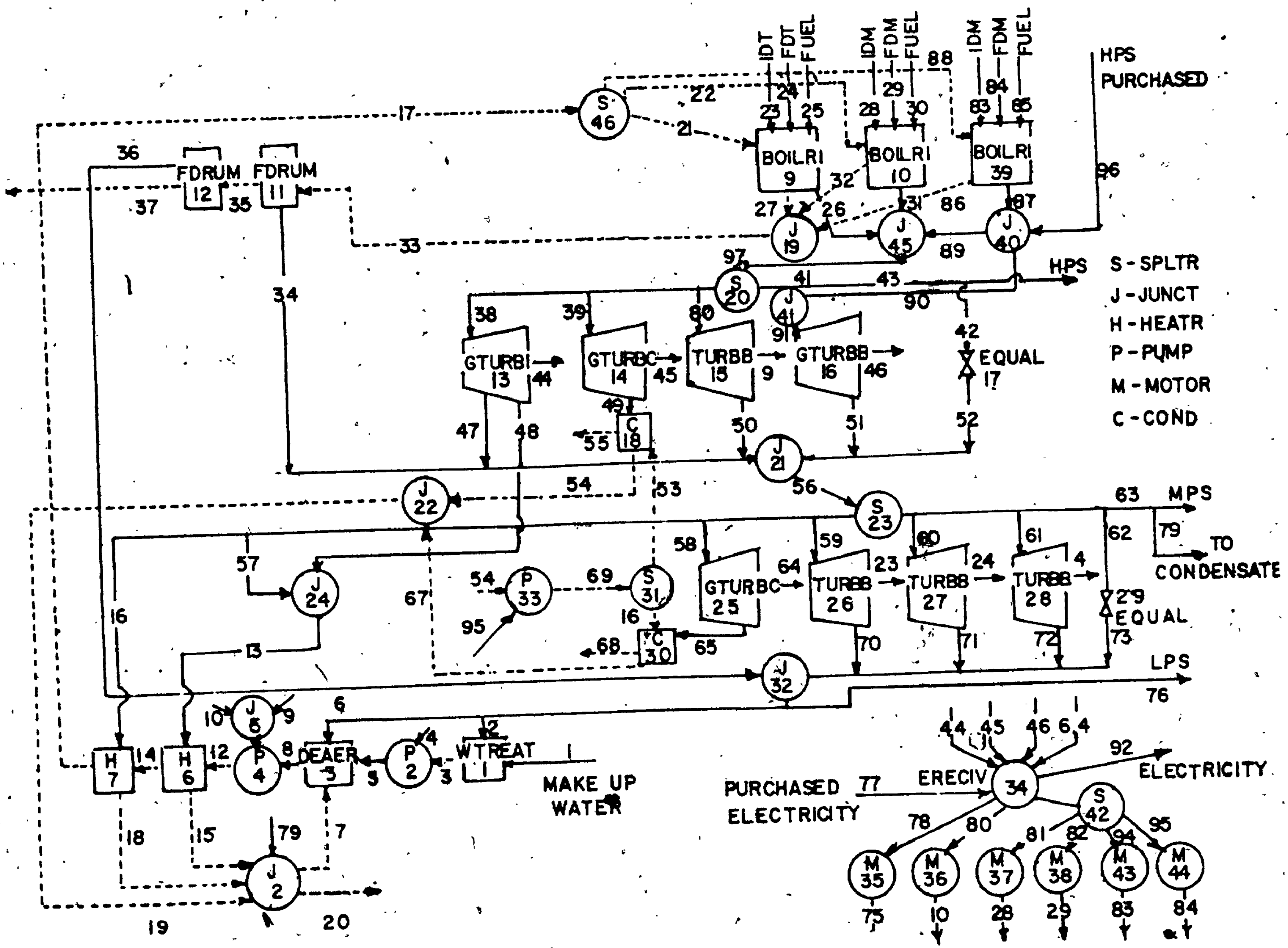


FIGURE 6.5 COMPLICATED STEAM AND POWER PLANT

SERVICES	STREAM NUMBER	CASE 1	CASE 2	CASE 3
		Mlb/hr		
Make-up water	1	1418.1	0.	874.8
L.P.S. to WTREAT	2	113.4	0.	70.0
M.P.S. to DEAER	6	114.9	0.	36.6
Condensate to DEAER	7	272.0	0.	683.1
L.P.S. to HEATR	13	90.8	0.	80.2
M.P.S. to HEATR	16	174.5	0.	157.6
Condensate to Process	20	1.0	1.0	1.0
B.F.W. to BOILR	21	947.4	0.	447.4
B.F.W. to BOILR	22	631.6	0.	631.6
Fuel to BOILR	25	36.	0.	50.4
Fuel to BOILR	30	24.	0.	33.6
Blow down to FDRUM	33	90.2	0.	79.7
H.P.S. to GTURB1	38	393.2	143.8	128.9
H.P.S. to GTURBC	39	0.	0.	92.8
H.P.S. to TURBB	40	52.2	0.	62.7
Bypass H.P.S.	42	410.0	410.0	410.0
H.P.S. Demand	43	128.4	128.4	128.4
Extractive Steam of GTURB1	47	302.4	143.5	48.7
Cooling Water to COND	53	0.	0.	1507.3
M.P.S. Header	56	1579.7	1178.0	1288.8
M.P.S. to GTURBC	58	0.	0.	353.4
M.P.S. to TURBB	59	45.7	0.	47.7
M.P.S. to TURBB	60	23.3	0.	24.3
M.P.S. to TURBB	61	1.5	0.	2.2
Bypass M.P.S.	62	245.6	91.7	120.5
M.P.S. Demand	63	1083.0	1083.0	583.0
Cooling water to COND	66	0.	0.	5873.0
L.P.S. Demand	76	92.0	92.0	92.0
		KWH		
Electricity purchased	77	5116.6	12818.1	0.0
Condensate return	79	0.	0.	0.
Fuel to BOILR	85	8.6	0.	0.8
B.F.W. to BOILR	88	226.0	0.	15.3
Electricity Demand	92	28700.0	28700.0	38700.0
H.P.S. purchased	96	0.	1302.1	0.
H.P.S. Header	97	1714.7	1307.9	1500.0

TABLE 6.2 FLOWRATE OF MAJOR STREAMS FOR THE SECOND EXAMPLE

SERVICES	FACILITY NUMBER	CASE 1	CASE 2	CASE 3
Water purification	1	o 1418.1 Mlb/hr	0.	o 874.6
Deaerator pump	2	o 1418.1 "	0.	o 874.6
Deaerator	3	n 205.9 "	0.	o 1594.2
B.F.W. Pump	4	n 205.9 "	0.	o 1594.2
Stage Heater 1	6	n 113.9 ft^2	0.	o 1425.4
Stage Heater 2	7	n 509.8 "	0.	n 268.0
Boiler 1	9	o 900.0 Mlb/hr	0.	o 900.0
Boiler 2	10	o 600.0 "	0.	o 600.0
Generator turbine 1	13	n 7609.1 KW	o 2608.1	o 4000.0
Generator turbine 2	14	0. "	0.	o 5700.0
Back turbine	15	o 1125.4 "	0.	o 994.0
Generator turbine 3	16	o 13280.0 "	o 13280.0	n 1446.8
Condenser 1	18	0 ft^2	0.	o 3327.0
Generator turbine 4	25	0 KW	0.	n 5487.5
Back turbine	26	o 1191. "	0.	o 1191.
Back turbine	27	o 441. "	0.	o 441.
Back turbine	28	o 28.2 "	0.	o 40.6
Condenser 2	30	o 0 ft^2	0.	o 12962.8
C.W. Pump	33	o 0 Mlb/hr	0.	o 7380.3
Power receiver	34	o 30005.7 KW	o 28706.2	n 6914.4
Motor for C.W. Pump	35	o 0 "	0.	o 247.6
Motor for B.F.W. Pump	36	o 0 "	0.	0.
Motor for I.D.F.	37	o 588.0 "	0.	o 588.0
Motor for F.D.F.	38	o 294.0 "	0.	o 294.0
Boiler 3	39	o 214.7 Mlb/hr	0.	o 14.5
Motor for I.D.F.	43	o 251.2 KW	0.	o 17.0
Motor for F.D.F.	44	o 105.2 "	0.	o 7.1

NOTE : o - Operating Capacity
n - New Expansion Capacity

Table 6.3 Expansion and Operating Capacities of Major Facilities for the Second Example function to be minimized. Furthermore, both the sensitivity and the absolute value of the installed cost are low enough to neglect the interaction among multi-time periods in the objective function. In fact, numerical studies confirmed that the operating cost dominates the fixed cost enough to use only operating cost as an objective function, as noted in Figure 6.2 of the first example.

Thus, a desirable strategy to determine an expansion policy over multi-time periods is outlined as follows:

1) The entire problem is broken down to sub-problems for each period, which are solved independently. This is a very efficient approach to computation.

2) Some adjustments are made for size and time of expansion overall. For instance, too small an expansion rate or consecutive expansion that are obtained as an original solution should be adjusted by early installation of larger capacities etc.

3) The problem is solved sequentially again considering the constraints imposed from an overall expansion policy. Namely, the confirmation of optimality is made under the

constraints based on an overall expansion policy.

6.4 Summary

Expansion problems of a steam and power plant were solved for two cases, i.e., with a simple system and with a complicated system. Numerical results found that the sensitivity and absolute value of the fixed cost are low enough to neglect the interaction among multi-time periods in the objective function. Then, a desirable strategy to determine an expansion policy over multi-time periods was outlined.

CHAPTER 7

APPLICATIONS TO THE OPTIMAL SYNTHESIS OF STEAM AND POWER PLANTS

7.1 Problem Description

The synthesis problem is to create a steam and power complex with an optimal configuration, designed to operate optimally to satisfy a set of internal and external demands.

Table 7.1 shows a set of steam and power demands for an information flow diagram given in Figure 7.1 where the basic structure of a steam and power plant is shown. This network is to be extended to a result that represents a selection, arrangement and set of optimum design conditions that will minimize venture cost including capital and operating cost.

As described in Chapter 4, the problem is solved by a two-level approach where an LP problem is solved at the upper level and a parameter optimization problem is solved at the lower level.

Table 7.1 Steam and Power Demands for the Synthesis Problem

Demand Items		Stream Number	Demands
Electricity		71	26140 kw
High Pressure Steam		25	75.6 Mlb/H
Medium Pressure Steam		74	1100 Mlb/H
Low Pressure Steam		94	139.3 Mlb/H
Condensate		81	26.3 Mlb/H
By-pass steam		26	> 410 Mlb/H
		80	> 10 Mlb/H
Internal			
power demand	1	4	(pump eq.2)
" "	2	4	(Pump eq.4)
" "	3	4	(Boiler eq.8)
" "	4	4	(Pump eq.26)
External			
power demand	5	4	184 KwH
" "	6	4	1288 KwH
" "	7	4	257.5 KwH
" "	8	4	225 KwH
" "	9	4	456 KwH
" "	10	4	368 KwH

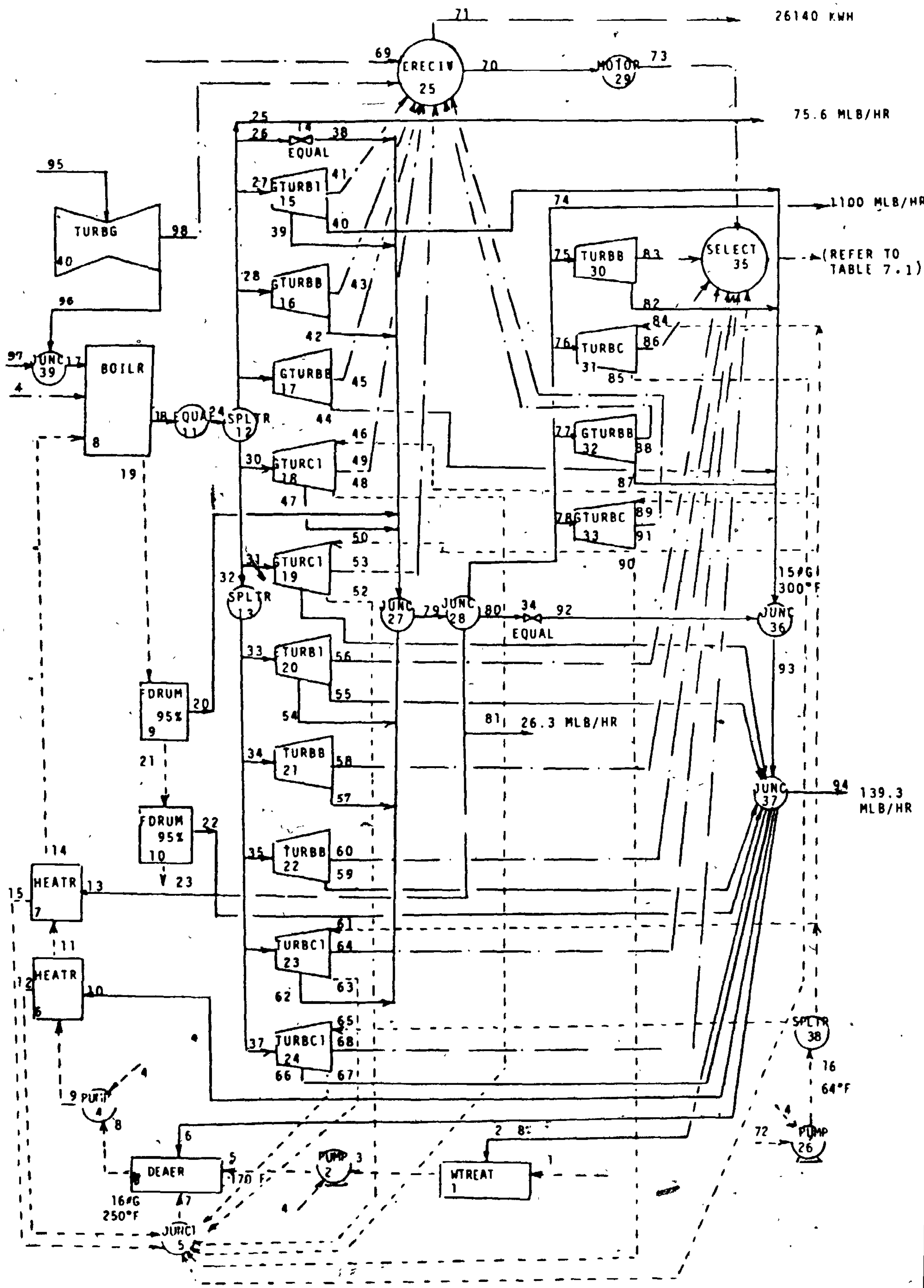


Figure 7.1 Information Flow Diagram of a Steam and Power Plant

7.2 Computational Results

Three steps were taken to attain the final optimal configuration for which parameters were also optimal. Table 7.2 presents the cost summary for every step of two-level coordination. Table 7.3 shows the parameter values assumed to obtain an optimal configuration for every step of coordination between two levels. Figures 7.2, 7.3 and 7.4 are optimal configurations which were obtained by using the parameter values assumed for every step of coordination between the two levels.

It should be noted that the parameter values assumed for every step are optimal for the configuration determined in every step except the initial values. In other words, basically a direct iteration method was employed for two-level coordination.

The computational time was about 0.3 minutes on a CDC CYBER 73 for LP formulation, 0.1 minutes by using MPSX on an IBM 370/165 for the determination of an optimal configuration and 1 minute on the CDC CYBER 73 for parameter optimization.

7.3 Discussions

As seen from the results shown in Tables 7.2 and 7.3, the the compound effect of temperature and pressure at the boiler

Table 7.2 Cost Summary for Every Step of the Two-level Coordination

OPTIMAL SOLUTION	STEP-1	STEP-2	STEP-3
OBJECTIVE FUNCTION *	55284.7	52993.4	52635.5
TOTAL ERECTED COST	36925.7	34708.9	37152.3
OPERATING COST(/YEAR)	35160.2	34077.1	32387.6
FUEL GAS	0.0	0.0	0.0
FUEL OIL	35019.3	32419.7	32105.9
COOLING WATER	0.0	0.0	0.0
ELECTRICITY	8.2	1524.8	149.0
INDUSTRIAL WATER	62.6	62.6	62.6
LABOR & SUPERVISION	70.0	70.0	70.0

(Thousand Dollars)

* Based on venture cost that is

$$V = C_p + dI + i_m(I + I_w)/(1-t)$$

where: $d = 20\%$, $i_m = 15\%$, $t = 50\%$

and $I_w = 15\%$ of I

Table 7.3 Parameters Assumed to Obtain an Optimal Configuration

DESCRIPTION	STEP-1	STEP-2	STEP-3	
Temperature of stream 11	290	290	290	^{o}F
Temperature of stream 14	350	350	350	^{o}F
Temperature of stream 18	750	605.9	601.3	^{o}F
Pressure of stream 18	574.7	452.7	538.9	psia

CONSTRAINTS			
Pressure of stream 18	=	Pressure of stream 9 - 25.3	psia
Pressure of stream 18	=	Pressure of stream 24 + 40.0	psia
Temperature of stream 18	=	Temperature of stream 24	^{o}F

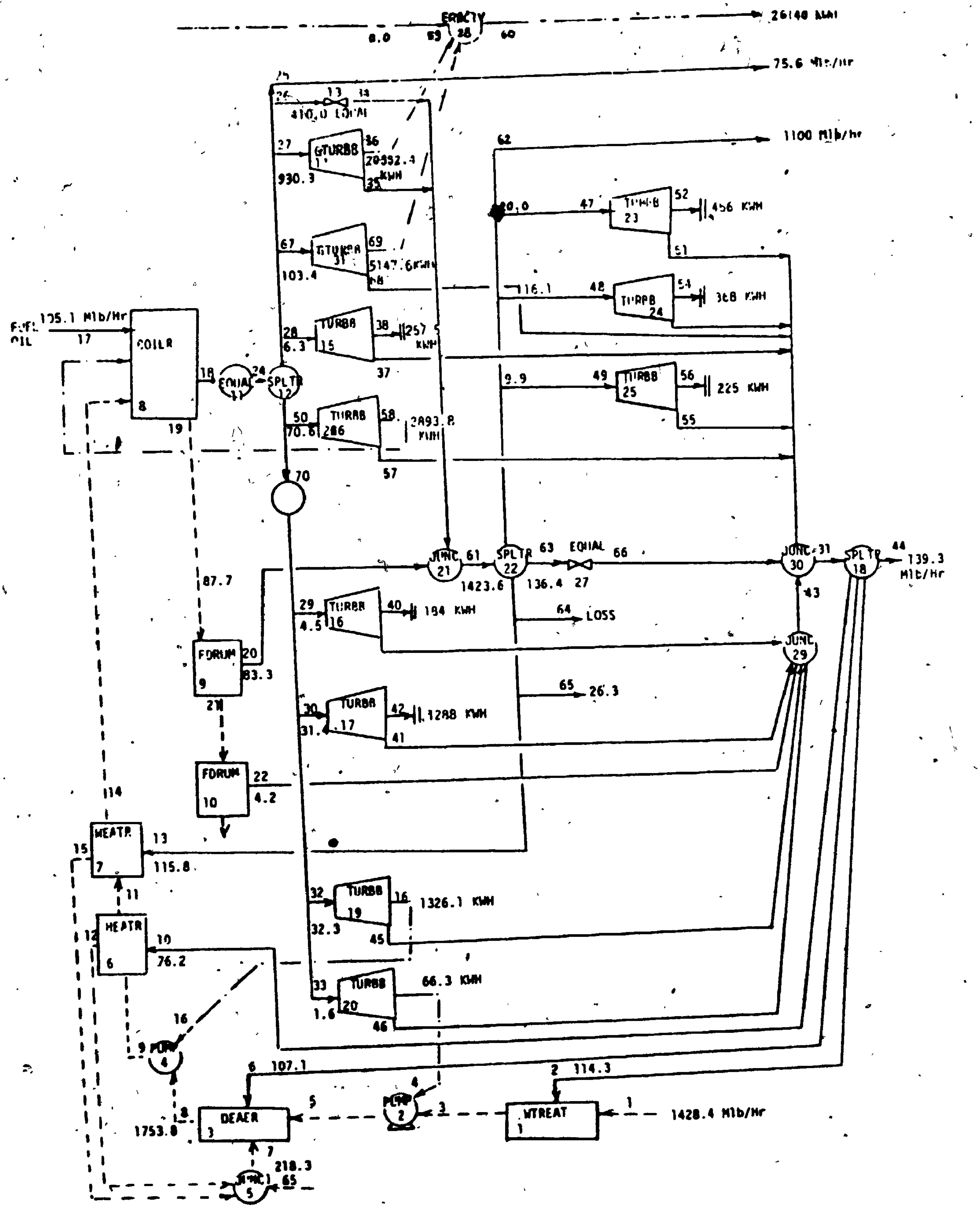


Figure 7.2 Information Flow Diagram for Optimal Configuration Step 1.

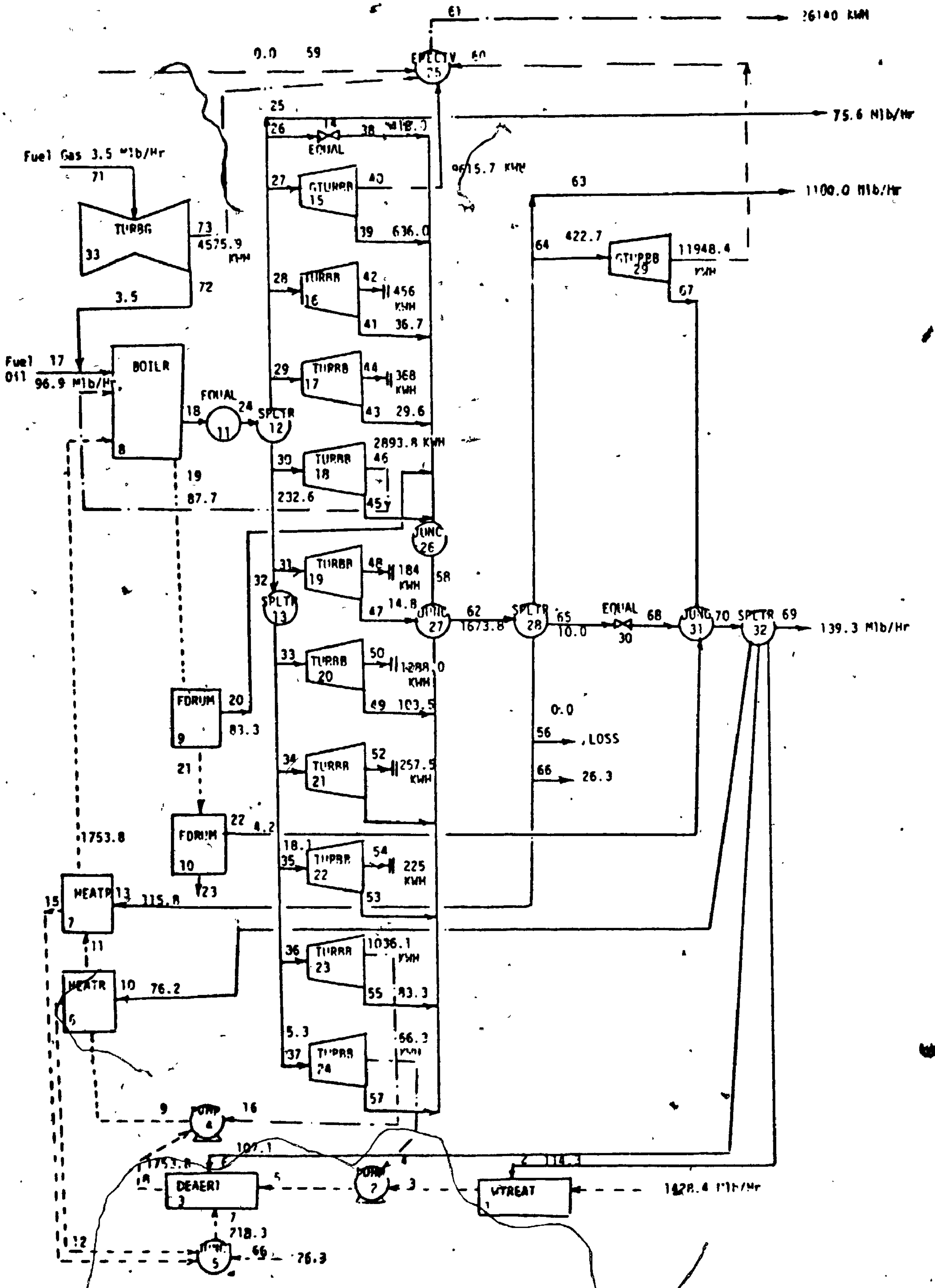


Figure 7.3 Information Flow Diagram for Optimal Configuration Step 2

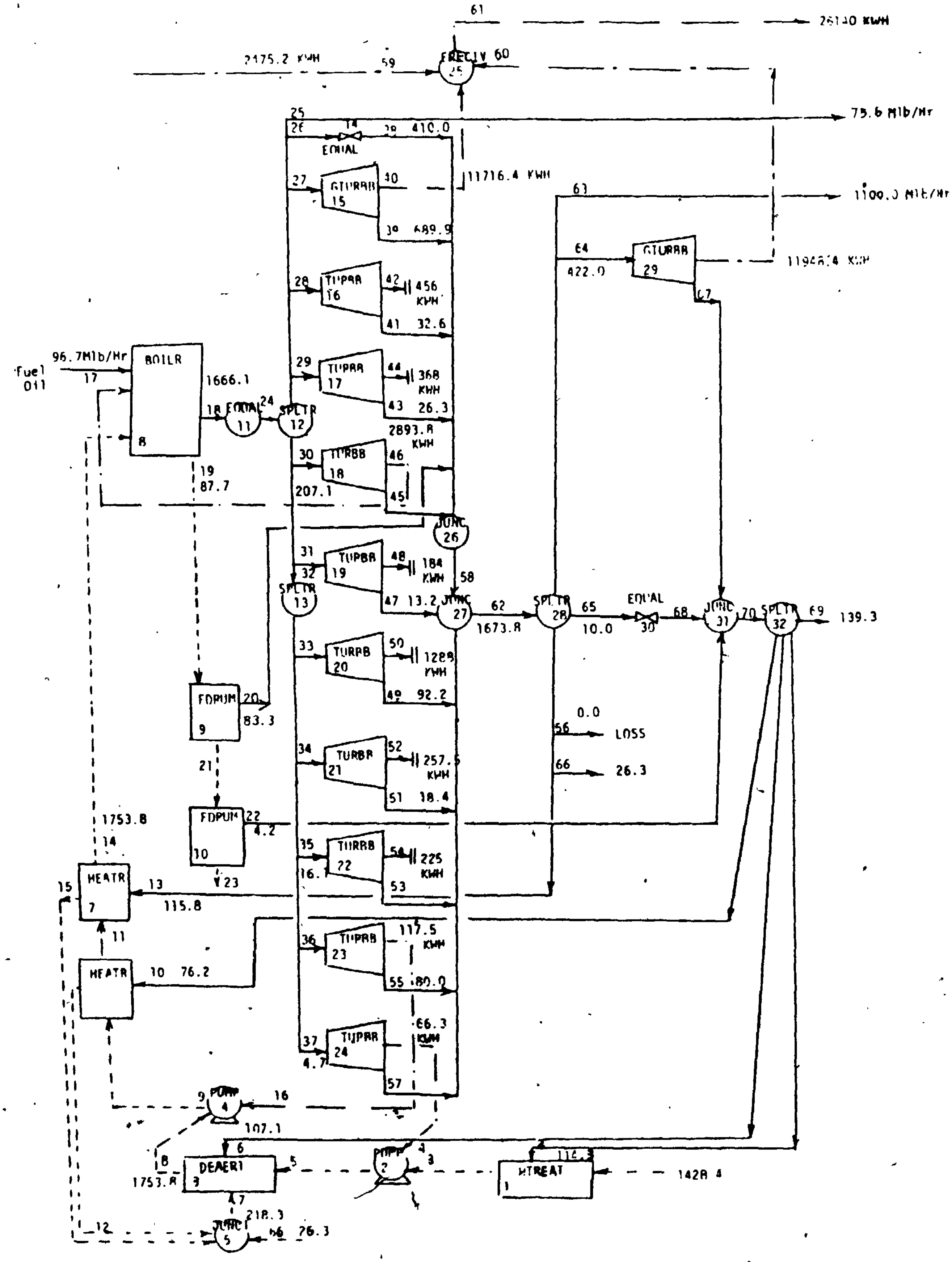


Figure 7.4 Information Flow Diagram for Optimal Configuration Step 3

outlet is the most important factor in determining an optimal configuration.

At high temperature and pressure in the initial step, the preferable configuration with which electricity is generated utilizing steam with high temperature and pressure is seen in Figure 7.2. Since the parameter optimization for Figure 7.2 finds as an optimal solution a boiler condition with low temperature and pressure (which conserves fuel and is a key factor for an optimal energy system), the configuration moves from Figure 7.2 to Figure 7.3 to satisfy the parameter changes. In the system of Figure 7.3 the relative amounts of higher and lower pressure steam generated are changed. Although a gas turbine appeared in the new configuration, the capacity is not large enough in practice. So the additional run was made by imposing constraints on the gas turbine and the LP program but this created Figure 7.4 as a secondary optimal configuration. In the system of Figure 7.4, the gas turbine has been removed from the plant. A slight modification such as this is practical and advantageous, compared to a complete automation of two-level coordination. Optimal parameters for Figure 7.4

are not very much different from those for Figure 7·2 and then the same configuration was confirmed to be optimal for the LP run using the optimal parameters for Figure 7·4.

Although the convergence of the direct substitution method for the two-level coordination would be quite good when optimal parameters are close for different configurations, it would not be guaranteed when optimal parameters depend heavily upon the configuration chosen. In other words, the direct iteration method for the two-level coordination would be good for the parameter-dominant case which is seen in a steam and power plant but it may not be satisfactory for the structure-dominant case which is often seen in a chemical process.

7.4 Summary

The practical synthesis problem for a steam and power plant was solved by a two-level approach where an LP problem was solved at the upper level and a parameter optimization problem was solved at the lower level. The coordination between two levels was satisfactorily made by a direct substitution method. The use of two computer executive systems "ODES" and "OSES" made an optimal synthesis of large system with heterogeneous structure possible within a short time.

CHAPTER 8

CONCLUSIONS AND RECOMMENDATIONS

8.1 Computer Executive System for Optimal Design of Arbitrary Energy Systems

A general computer program system for the optimal design for arbitrary energy systems was developed so that a prompt and efficient evaluation for arbitrary energy systems may be possible in accordance with any conditions set. The program system "ODES" has the following features:

- The program system structure is flexible and open-ended so that any new modules may be added for an evaluation of new energy systems.

- Unit models are set up by a modular approach and have a one-to-one relationship to physical equipment in most cases. An energy balance for arbitrary energy systems is formulated into a set of homogeneous linear equations; these are solved efficiently by the triangulation method. Only essential elements of calculation are stored for iterative calculations such as are encountered in optimization.

- The data structure is succinct. It makes possible a centralization of dimensioning for variables and an efficient storage for stream and equipment data with variable length.

- The format for input data is simple and has a common interface with that of other relevant computer program systems "OSES" and "OPES".

- An optimization routine is provided to choose optimal conditions. The Complex method is chosen as an optimization technique which is suitable for a problem with multi-variables and stiff inequality constraints.

Numerical examples demonstrated how useful the program package is for analysing and evaluating arbitrary energy systems. Numerical results also indicated the efficiency of the simulation method and how much energy saving can be achieved by seeking an optimum energy untilization.

Another example of a useful application of "ODES" is given in Appendix 2 that is a solving technique for heat exchanger networks.

8.2 Optimal Expansion of Steam and Power Plants

The expansion problem for a steam and power plant was formulated as a synthesis problem with constraints. A preliminary choice of an optimal expansion case was achieved by solving a multi-time period linear programming problem. The development of the "OPES" computer program system had made it easy to evaluate alternate expansion cases.

The program structure is flexible and open-ended so that new modules may be added for the evaluation of arbitrary energy systems.

8.3 Optimal Synthesis of Steam and Power Plants

A synthesis problem of steam and power plant which is a complex system with a heterogeneous structure was defined, formulated and solved by the combination of decomposition, heuristics and optimization.

The direct substitution method was used for two-level coordination in decomposition.

The solution method for optimal synthesis of steam and power plant was generalized by the support of two computer executive systems, "OSES" that automatically formulates an LP problem at the upper level, and "ODES" that automatically implements the parameter optimization at the lower level.

Numerical tests confirmed the effectiveness of the two-level approach and the usefulness of two computer executive systems.

8.4 Recommendations

Although the methodologies which were developed in this study and their applications were confined to steam and power plants for chemical complexes, the following extensions would be possible:

- The optimal design package "ODES" can be utilized for an optimal operation of energy systems in general by creating the appropriate "operation modules".

- Process systems can be included in an optimal design of energy systems in the case where process systems interact strongly with energy systems (Nishio, 1975). The triangulation method can be extended efficiently to network solving of process systems (Nishio, 1974).

- The optimal design package "ODES" can be applied to a heat recovery system (which is described briefly in Appendix 2), refrigeration systems with solar energy utilization, common utility center and so on.

- Integrated and interactive design systems for "ODES", "OPES" and IBM's MPSX, could enhance the effectiveness and productivity in studying process energy systems.

APPENDIX 1

CHOICE OF ECONOMIC CRITERION FOR A SYNTHESIS OF A STEAM AND POWER PLANT

When a synthesis of a steam and power plant for a chemical complex is planned, it is usually not independent from other systems. In other words, a synthesis of steam and power plant is considered because a synthesis of process systems is planned. Then, it can be said that the steam and power plant is only a sub-system of the total system. Accordingly, the economic criterion to be chosen for the sub-system should be consistent with that of the total system.

The use of venture-profit concept (Happel, 1958) seems to be common in an evaluation of the attractiveness of a venture in a chemical complex, so that basically the same criterion should be applied to the steam and power plant which is a sub-system in the complex, from a viewpoint of the decomposition principle.

$$V = P - i_m (I + I_w) \qquad (1.1)$$

where: V : venture profit

P : net profit

i_m : minimum rate of return considered attractive

I : investment in facilities

I_w : working capital

Net profit P can be expressed as follows:

$$P = R - eI - t\,(R - dI) \qquad (1.2)$$

where: R : gross profit

e : depreciation rate for accounting purpose

t : income tax rate (Federal and Provincial)

d : depreciation rate for tax purposes

It may be assumed that e and d are roughly equal for approximate purpose. Substituting equation (1.2) into equation (1.1) yields

$$V = \left[R - dI - \frac{i_m(I+I_w)}{(1-t)}\right](1-t) \qquad (1.3)$$

For a steam and power plant, the external energy demands are considered to be constant, therefore, only expenses may be taken into account in the term R of equation (1.3) as an objective function to optimize with respect to independent variables, namely:

$$\text{Minimize } V = Cop + dI + i_m\,(I + I_w)/(1-t) \qquad (1.4)$$

where Cop is operating expenses.

The same concept may be adapted to an expansion problem of steam and power plant except for the consideration over a period of years. Therefore, the venture-worth method seems to be reasonable as a criterion of attractiveness of venture.

The venture-worth method judges the profitability of a project by the present value of the venture profit which is obtained during each year of the project's expected life.

Similarly to equation (1.4) an objective function to optimize with respect to independent variables for expansion problem of steam and power plant can be written as follows:

$$\text{Minimize } W = \sum_{k=1}^{n} \frac{V_k}{(1+i)^k} \tag{1.5}$$

where:

i is the rate of earnings of the company on its invested capital, n is the period considered and

$$V_k = Cop,_k + d_k I_k + i_m (I_k + I_{w,k})/(1-t) \tag{1.6}$$

APPENDIX 2

A SOLVING TECHNIQUE FOR HEAT EXCHANGE NETWORKS

2.1 Introduction

This appendix introduces a solving technique of heat exchange networks as an example of a useful application of "ODES" to a system design problem.

In general, large heating and cooling duties are required in chemical processes in order to make final products. Some fluids that are to be cooled would be able to exchange their heat with some other fluids that are to be heated. Each fluid to be cooled or heated has a different temperature level. One of the challenges of synthesis is to create an optimum heat exchange network consisting of the number of fluids that are to be cooled or heated and, physically of coolers, heaters or heat exchangers which is an optimized system.

A conventional way of solving heat exchange networks is to solve each unit sequentially. However, this approach may not be wise because many of the system outlet temperatures and some of the intermediate temperatures are specified, and heat recycle loops may be often included.

On the other hand, Hiraizumi and Nishimura (1966) solved a heat exchanger network, which was represented by a"linear model" method, using a network processing technique. Also Ohnishi and Kikkawa (1968) solved an optimal design problem for heat exchange systems by using a simultaneous approach. However, both simultaneous approaches have drawbacks relating to the treatment of specified intermediate temperatures and heterogeneous units, i.e., coolers, heaters, air fin coolers and so on. Especially, the latter method is not efficient in computation time because there is no device to deal with the sparse matrix in simultaneous equations in case of optimal design.

In this appendix a new method which represents the system heat balance in terms of a set of simultaneous equations with a homogeneous form is presented.

2.2 Formulation of System Heat Balances

A heat exchange system shown in Figure II.1 is taken as an example to show how to formulate system heat balances.

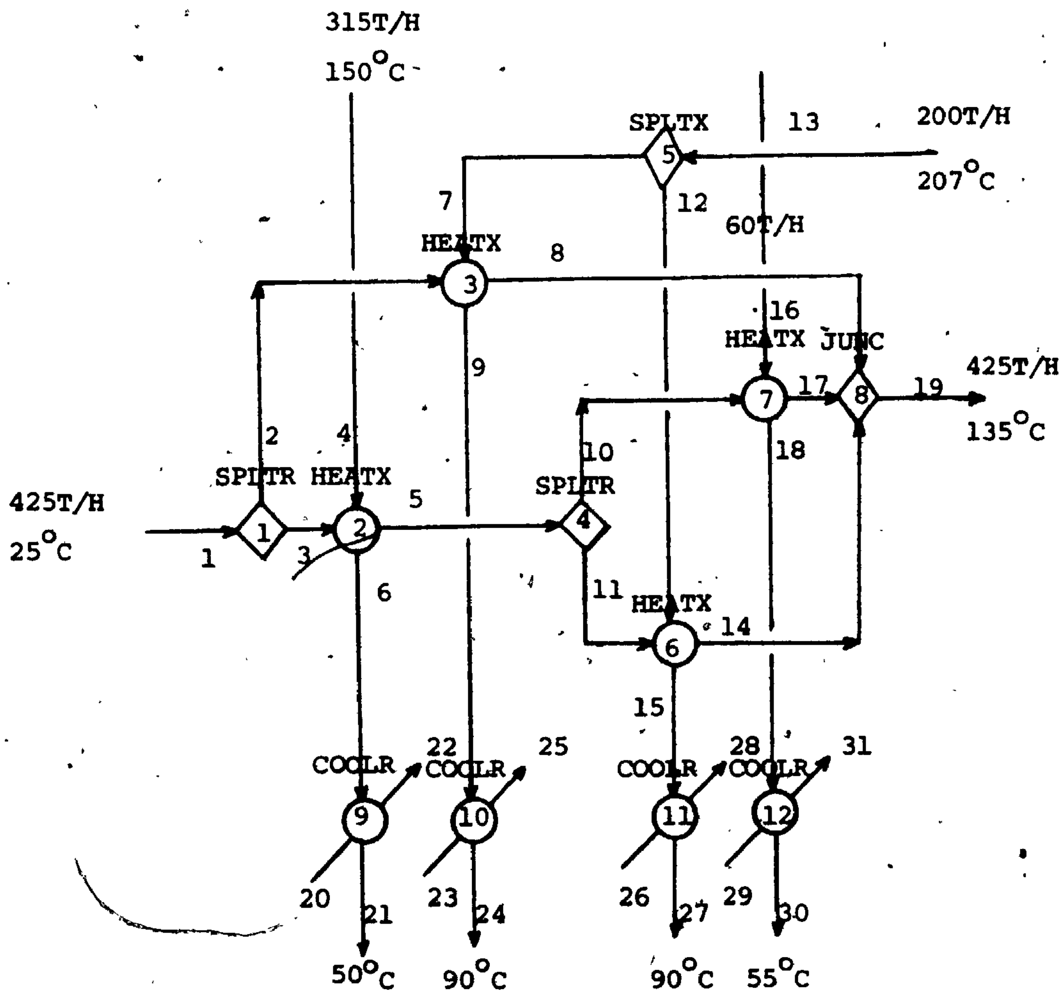


Figure II.1 Information Flow Diagram for a Heat Exchange System

In the similar way to the formulation of energy balances which was described in Chapter 2, the following heat balance equations are derived through the examination of each module:

$$Q_1 - Q_2 - Q_3 = 0 \quad \text{(SPLTR)} \tag{2.1}$$

$$Q_1 - \alpha_1 Q_2 = 0 \quad \text{(SPLTR)} \tag{2.2}$$

$$Q_6 - (1 - \beta_1) Q_4 = 0 \quad \text{(HEATEX)} \tag{2.3}$$

$$Q_5 - \beta_1 Q_4 - Q_3 = 0 \quad \text{(HEATEX)} \tag{2.4}$$

$$Q_7 - (1 - \beta_2) Q_9 = 0 \quad \text{(HEATEX)} \tag{2.5}$$

$$Q_8 - \beta_2 Q_7 - Q_2 = 0 \quad \text{(HEATEX)} \tag{2.6}$$

$$Q_5 - Q_{10} - Q_{11} = 0 \quad \text{(SPLTR)} \tag{2.7}$$

$$Q_5 - \alpha_2 Q_{10} = 0 \quad \text{(SPLTR)} \tag{2.8}$$

$$Q_{13} - Q_7 - Q_{12} = 0 \quad \text{(SPLTR)} \tag{2.9}$$

$$Q_{13} - \alpha_3 Q_7 = 0 \quad \text{(SPLTR)} \tag{2.10}$$

$$Q_{14} - (1-\beta_3) Q_{12} = 0 \quad \text{(HEATX)} \tag{2.11}$$

$$Q_{14} - \beta_3 Q_{12} - Q_{11} = 0 \quad \text{(HEATX)} \tag{2.12}$$

$$Q_{16} - Q_{18} - Q_{17} + Q_{10} = 0 \quad \text{(HEATX)} \tag{2.13}$$

$$Q_8 + Q_{17} + Q_{14} - Q_{19} = 0 \quad \text{(MIXER)} \quad (2.14)$$

$$Q_6 - Q_{21} - Q_{22} + Q_{20} = 0 \quad \text{(COOLR)} \quad (2.15)$$

$$Q_{22} - i_{20}/i_{22}\ Q_{20} = 0 \quad \text{(COOLR)} \quad (2.16)$$

$$Q_9 - Q_{24} - Q_{25} + Q_{23} = 0 \quad \text{(COOLR)} \quad (2.17)$$

$$Q_{25} - i_{23}/i_{25}\ Q_{23} = 0 \quad \text{(COOLR)} \quad (2.18)$$

$$Q_{15} - Q_{27} - Q_{28} + Q_{26} = 0 \quad \text{(COOLR)} \quad (2.19)$$

$$Q_{28} - i_{26}/i_{28}\ Q_{26} = 0 \quad \text{(COOLR)} \quad (2.20)$$

$$Q_{18} - Q_{30} - Q_{31} + Q_{29} = 0 \quad \text{(COOLR)} \quad (2.21)$$

$$Q_{31} - i_{29}/i_{31}\ Q_{29} = 0 \quad \text{(COOLR)} \quad (2.22)$$

where the unit model for a heat exchanger is as follows, based on a schematic diagram (Figure II. 2) and signal flow diagram (Figure II. 3).

$$\begin{pmatrix} Q_{1,out} \\ Q_{2,out} \end{pmatrix} = \begin{pmatrix} 1-\beta & 0 \\ \beta & 1 \end{pmatrix} \begin{pmatrix} Q_{1,in} \\ Q_{2,in} \end{pmatrix} \quad (2.23)$$

Equations (2.1) to (2.22) can be expressed in the compact forms

$$\underline{\underline{Q}}'\ \underline{\underline{A}} = \underline{0}' \quad (2.24)$$

where $\underline{\underline{A}}$ is defined in Table II.1.

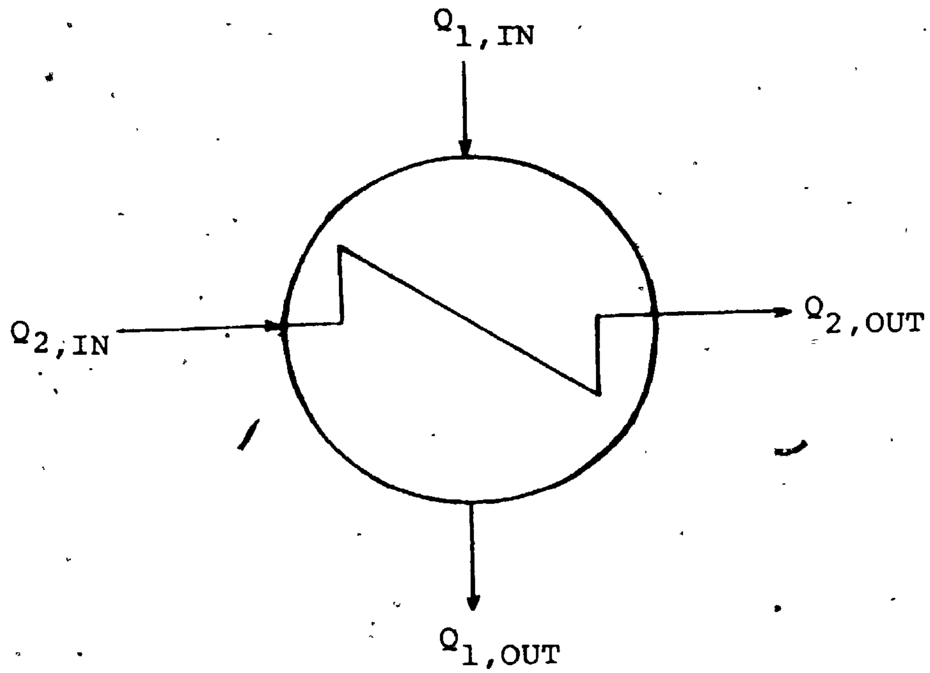


Figure II.2 Symbolic Figure of a Heat Exchanger Unit

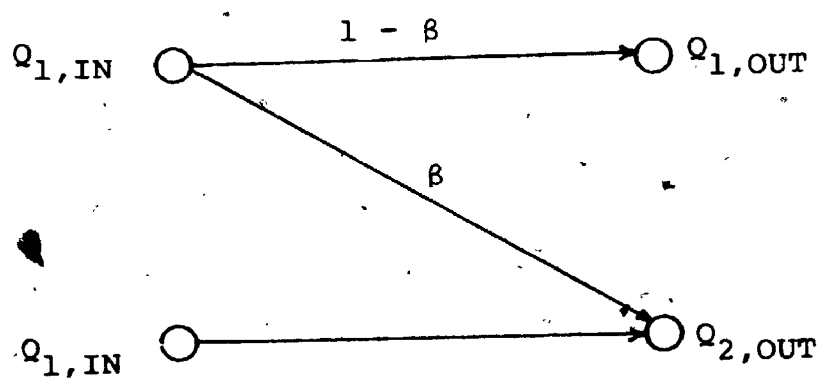


Figure II.3 Signal Flow Diagram of a Heat Exchanger Unit

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	-1	a_1	.	.	.	-1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	-1	.	.	-1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	.	.	a_2	a_3	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	1	.	.	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	1	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.
7	.	.	.	.	1	a_5	.	.	-1	a_7	.	.	.	.	.	.	.	.	.	.	.	.
8	.	.	.	.	.	1	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.
9	.	.	.	.	a_4	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.
10	.	.	.	.	.	.	-1	a_6	.	.	.	.	1	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	-1	.	.	.	.	-1	.	.	.	.	.	.	.	.	.	.
12	.	.	.	.	.	.	.	.	-1	.	a_8	a_9	.	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	1	1	.	.	.	.	.	.	.	.	.	.	.	.
14	.	.	.	.	.	.	.	.	.	.	.	1	.	1	.	.	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	1	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	-1	1	.	.	.	.	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	-1	.	.	.	.	.	.	.	1	.
19	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	.	.	.	.	.	.	.	.
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	a_{10}	.	.	.	.	.	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	1	.	.	.	.	.	.
22	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	.	.	.	.	.	.	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	a_{11}	.	.	.	.
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	.	.	.	.	.
25	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	1	.	.	.	.
26	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	a_{12}	.	.
27	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	.	.	.
28	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	-1	.	.
29	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	a_{13}
30	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	.
31	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-1	1

. implies zero element

Table II.1 Coefficient Matrix A.

2.3 Discussion

A set of simultaneous equations (2.24) includes 22 unknown variables and 9 known variables. As it is obvious in Table II.1, the equation set has a sparse coefficient matrix with density less than 10%. Accordingly the device to deal with a sparse matrix is quite important from the computational point of view in the case, where the equations are repeatedly solved, such as case study or optimization. Thus, the solution method that was developed in Chapter 2 can be applied to this problem. Above all, an entire problem, e.g. an optimal design of heat exchanger networks can be achieved within the framework of the "ODES" computer executive system (Nishio, 1975a).

APPENDIX 3

COMPARISON OF ENERGY BALANCE FORMULATION BY HOMOGENEOUS AND HETEROGENEOUS FORMS

To show the difference in formulation by heterogeneous and homogeneous forms, a simple steam and power system given in Figure III.1 is taken up as an energy system.

In an ordinary formulation of simultaneous equations by the heterogeneous form, the unknown variables are on the left-hand side and the known variables or constants are brought into the right hand side.

Table III.1 shows a formulation by heterogeneous form based on this ordinary way, where those variables which are known had to be designated a priori.

However, since this arrangement is not easy in the general case, it can be avoided by introducing extra equations as shown in Table III.2. Both formulations are expressed using a matrix form, respectively in Tables III.3 and III.4.

On the other hand, the formulation by a homogeneous form corresponds to an original equation set seen in Table III.2 before introducing extra equations. The equation set

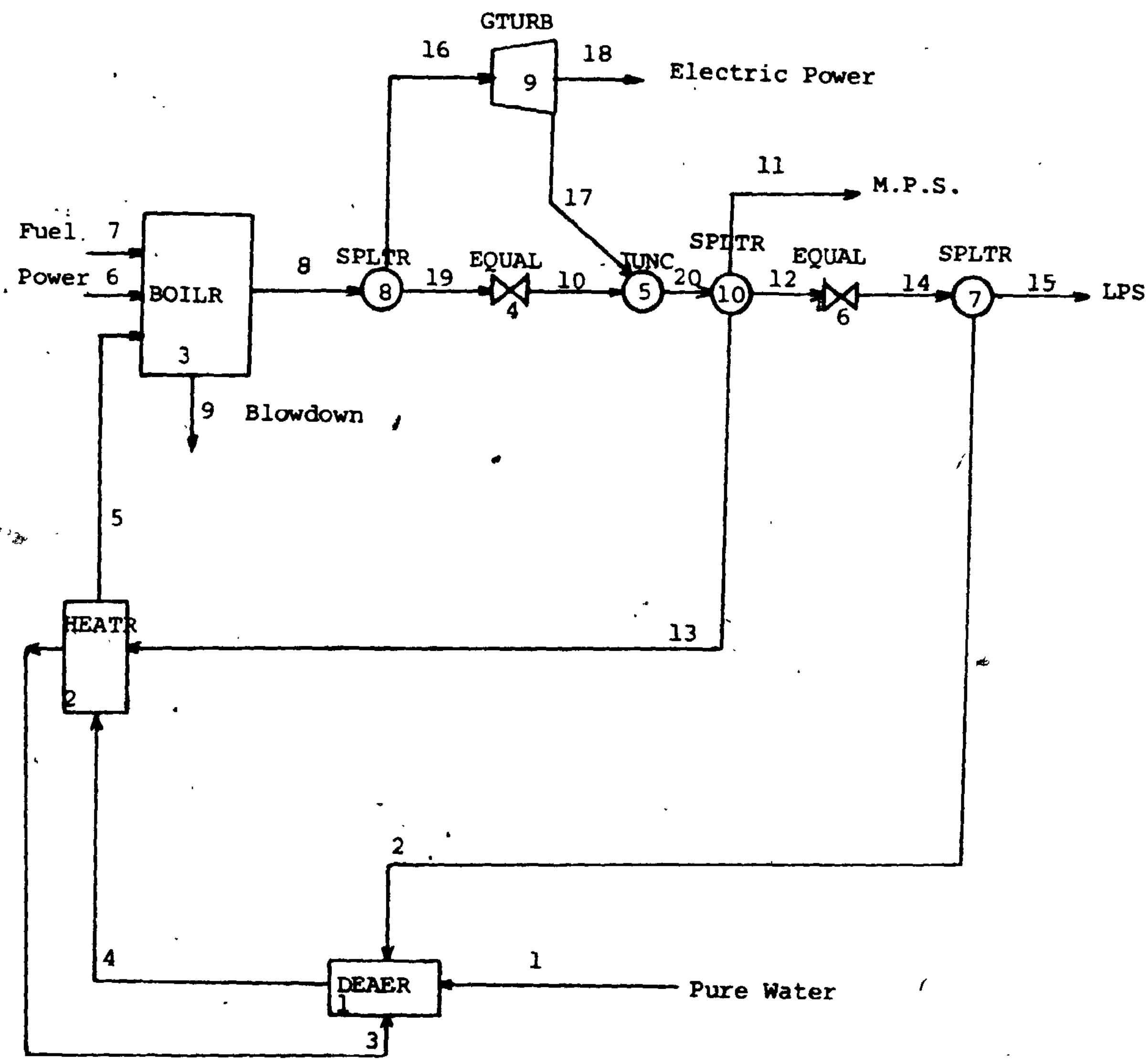


Figure III.1 Information Flow Diagram of a Simple Steam and Power System

Equation Number	Equations for Energy Balance	Module Name
1	$x_1 + x_2 + x_3 + x_4 = 0$	DEAER
2	$a_1 + a_2x_2 + a_3x_3 - a_4x_4 = 0$	DEAER
3	$x_4 - x_5 = 0$	HEATR
4	$x_{13} - x_3 = 0$	HEATR
5	$a_5x_3 - a_6x_4 + a_7x_5 - a_8x_{13} = 0$	HEATR
6	$x_8 - a_9x_5 = 0$	BOILER
7	$x_9 - a_{10}x_5 = 0$	BOILER
8	$a_{11}x_5 + a_{12}x_7 - a_{13}x_8 - a_{14}x_9 = 0$	BOILER
9	$x_6 - a_{15} x_5 = 0$	BOILER
10	$x_{19} - x_{10} = 0$	EQUAL
11	$x_{10} + x_{19} - x_{20} = 0$	JUNC
12	$x_{12} - x_{14} = 0$	EQUAL
13	$-x_{14} + x_2 = -x_{15}$	SPLTR
14	$-x_8 + x_{16} + x_{19} = 0$	SPLTR
15	$x_{16} - x_{17} = 0$	GTURBB
16	$a_{16} x_{16} = x_{18}$	GTURBB
17	$x_{12} + x_{13} - x_{20} = -x_{11}$	SPLTR

where each coefficient was simplified as a_i

Table III.1 Formulation of Energy Balance by Heterogeneous Form -1

Equation Number	Equations for Energy Balance	Module Name
1	$x_1 + x_2 + x_3 + x_4 = 0$	DEAER
2	$a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4 = 0$	DEAER
3	$x_4 - x_5 = 0$	HEATR
4	$x_{13} - x_3 = 0$	HEATR
5	$a_5x_3 - a_6 x_4 + a_7x_5 - a_8x13 = 0$	HEATR
6	$x_8 - a_9 x_5 = 0$	BOILR
7	$x_9 - a_{10}x_5 = 0$	BOILR
8	$a_{11}x_5 + a_{12}x_7 - a_{13}x_8 - a_{14}x_9 = 0$	BOILR
9	$x_6 - a_{15}x_5 = 0$	BOILR
10	$x_{19} - x_{10} = 0$	EQUAL
11	$x_{10} + x_{19} - x_{20} = 0$	JUNC
12	$x_{12} - x_{14} = 0$	EQUAL
13	$-x_{14} + x_2 + x_{15} = 0$	SPLTR
14	$-x_8 + x_{16} + x_{19} = 0$	SPLTR
15	$x_{16} - x_{17} = 0$	GTURBB
16	$a_{16}x_{16} - x_{18} = 0$	GTRUBB
17	$x_{11} + x_{12} + x_{13} - x_{20} = 0$	SPLTR
18	$x_{15} = a_{17}$	
19	$x_{18} = a_{18}$	
20	$x_{11} = a_{19}$	

Table III.2 Formulation of Energy Balance by Heterogeneous Form -2

Variable No.

Equation No.	1	2	3	4	5	6	7	8	9	10	12	13	14	16	17	19	20	x		b
1	1.0	1.0	1.0	1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	x_1		0
2	a_1	a_2	a_3	a_4	0	0	0	0	0	0	0	0	0	0	0	0	0	x_2		0
3	0	0	0	1.0	-1.0	0	0	0	0	0	0	0	0	0	0	0	0	x_3		0
4	0	0	-1	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	x_4		0
5	0	0	a_5	$-a_6$	a_7	0	0		0	0	0	0	$-a_8$	0	0	0	0	x_5		0
6	0	0	0	0	$-a_9$	0	0	1.0	0	0	0	0	0	0	0	0	0	x_6		0
7	0	0	0	0	$-a_{10}$	0	0	0	1.0	0	0	0	0	0	0	0	0	x_7		0
8	0	0	0	0	a_{11}	0	a_{12}	$-a_{13}$	$-a_{14}$	0	0	0	0	0	0	0	0	x_8	=	0
9	0	0	0	0	$-a_{15}$	1.0	0	0	0	0	0	0	0	0	0	0	0	x_9		0
10	0	0	0	0	0	0	0	0	0	-1.0	0	0	0	0	0	0	0	x_{10}		0
11	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	1.0	0	-1.0	x_{12}		0
12	0	0	0	0	0	0	0	0	0	0	1.0	0	-1.0	0	0	0	0	x_{13}		0
13	0	1.0	0	0	0	0	0	0	0	0	0	0	-1.0	0	0	0	0	x_{14}		$-x_{15}$
14	0	0	0	0	0	0	0	-1.0	0	0	0	0	0	1.0	0	1.0	0	x_{16}		0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1.0	-1.0	0	0	x_{17}		0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	a_{16}	0	0	0	x_{19}		x_{18}
17	0	0	0	0	0	0	0	0	0	0	1.0	1.0	0	0	0	0	-1.0	x_{20}		$-x_{11}$

Table III.3 Coefficient Matrix $\underline{A}$ and Vector $\underline{b}$ for Formulation -1

Variable no.

Equation no.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20			
1	1.0	1.0	1.0	-1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x_1		0
2	a_1	a_2	a_3	$-a_4$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x_2		0
3	0	0	0	1.0	-1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x_3		0
4	0	0	-1.0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	0	0	0	x_4		0
5	0	0	a_5	$-a_6$	a_7	0	0	0	0	0	0	0	$-a_8$	0	0	0	0	0	0	0	x_5		0
6	0	0	0	0	$-a_9$	0	0	1.0	0	0	0	0	0	0	0	0	0	0	0	0	x_6		0
7	0	0	0	0	$-a_{10}$	0	0	0	1.0	0	0	0	0	0	0	0	0	0	0	0	x_7		0
8	0	0	0	0	a_{11}	0	a_{12}	$-a_{13}$	$-a_{14}$	0	0	0	0	0	0	0	0	0	0	0	x_8		0
9	0	0	0	0	$-a_{15}$	1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x_9	=	0
10	0	0	0	0	0	0	0	0	0	-1.0	0	0	0	0	0	0	0	0	1.0	0	x_{10}		0
11	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	0	0	1.0	0	0	-1.0	x_{11}		0
12	0	0	0	0	0	0	0	0	0	0	0	1.0	0	-1.0	0	0	0	0	0	0	x_{12}		0
13	0	1.0	0	0	0	0	0	0	0	0	0	0	0	-1.0	1.0	0	0	0	0	0	x_{13}		0
14	0	0	0	0	0	0	0	-1.0	0	0	0	0	0	0	0	1.0	0	0	1.0	0	x_{14}		0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.0	-1.0	0	0	0	x_{15}		0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	a_{16}	0	0	-1.0	0	x_{16}		0
17	0	0	0	0	0	0	0	0	0	0	1.0	1.0	1.0	0	0	0	0	0	0	-1.0	x_{17}		0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	0	x_{18}		a_{17}
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.0	0	0	x_{19}		a_{18}
20	0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	0	0	0	0	0	x_{20}		a_{19}

Table III.4 Coefficient Matrix A and Vector b for Formulation -2 is transposed to Table III.5 so that the triangulation method can be applied to solve it as described in Chapter 2. It should be also noted that there is a need for pre-treatment in order to solve the equation set; this is described in Appendix 4.

$$\underline{X}'\underline{A} = \underline{O}'$$

where

$$\underline{X}' = (X_1,\ X_2, \ldots\ldots, X_{20})$$

$$\underline{O}' = (0,\ 0, \ldots\ldots\ldots, 0)$$

$\underline{A}$ =

Variable No. \ Equation No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1.0	a_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1.0	a_2	0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0
3	1.0	a_3	0	-1.0	a_5	0	0	0	0	0	0	0	0	0	0	0	0
4	-1.0	$-a_4$	1.0	0	$-a_6$	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	-1.0	0	a_7	$-a_9$	$-a_{10}$	a_{11}	$-a_{15}$	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	1.0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	a_{12}	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	1.0	0	$-a_{13}$	0	0	0	0	0	-1.0	0	0	0
9	0	0	0	0	0	0	1.0	$-a_{14}$	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	-1.0	1.0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.0
12	0	0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0	1.0
13	0	0	0	1.0	$-a_8$	0	0	0	0	0	0	0	0	0	0	0	1.0
14	0	0	0	0	0	0	0	0	0	0	0	-1.0	-1.0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	1.0	1.0	a_{16}	0
17	0	0	0	0	0	0	0	0	0	0	1.0	0	0	0	-1.0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1.0	0
19	0	0	0	0	0	0	0	0	0	1.0	0	0	0	1.0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	-1.0	0	0	0	0	0	-1.0

Table III.5: Coefficient Matrix for Homogeneous Form

APPENDIX 4

PRETREATMENT TO SOLVE A SET OF SIMULTANEOUS EQUATIONS

As pointed out in Appendix 3, when balance equations are formulated initially they are still not ready to be solved because unknown variables to be solved are not distinguishable from known variables and in addition diagonal elements are not always non-zero as is required for solution.

Thus, the first thing to do as a pretreatment of solving a set of simultaneous equations is a sorting of unknown and known variables. This is quite straight forward. Since known variables have to be placed on the lower part in the sequence of variables in the Triangulation method, the replacement between locations of known and unknown variables takes places systematically on the lower part in the sequence of variables. Figure IV.1 shows an example of the sorting.

The second thing to do is a rearrangement of location of variables and equations. The algorithm for this is to choose in turn the row or column which has minimum elements. As for the rows or columns with same number of elements, the corresponding column or row which has minimum elements is chosen as a pair. Figure IV.2 illustrates the sequence of rearrangement, for variables and equations for the example

→ points to changes at successive sorting steps

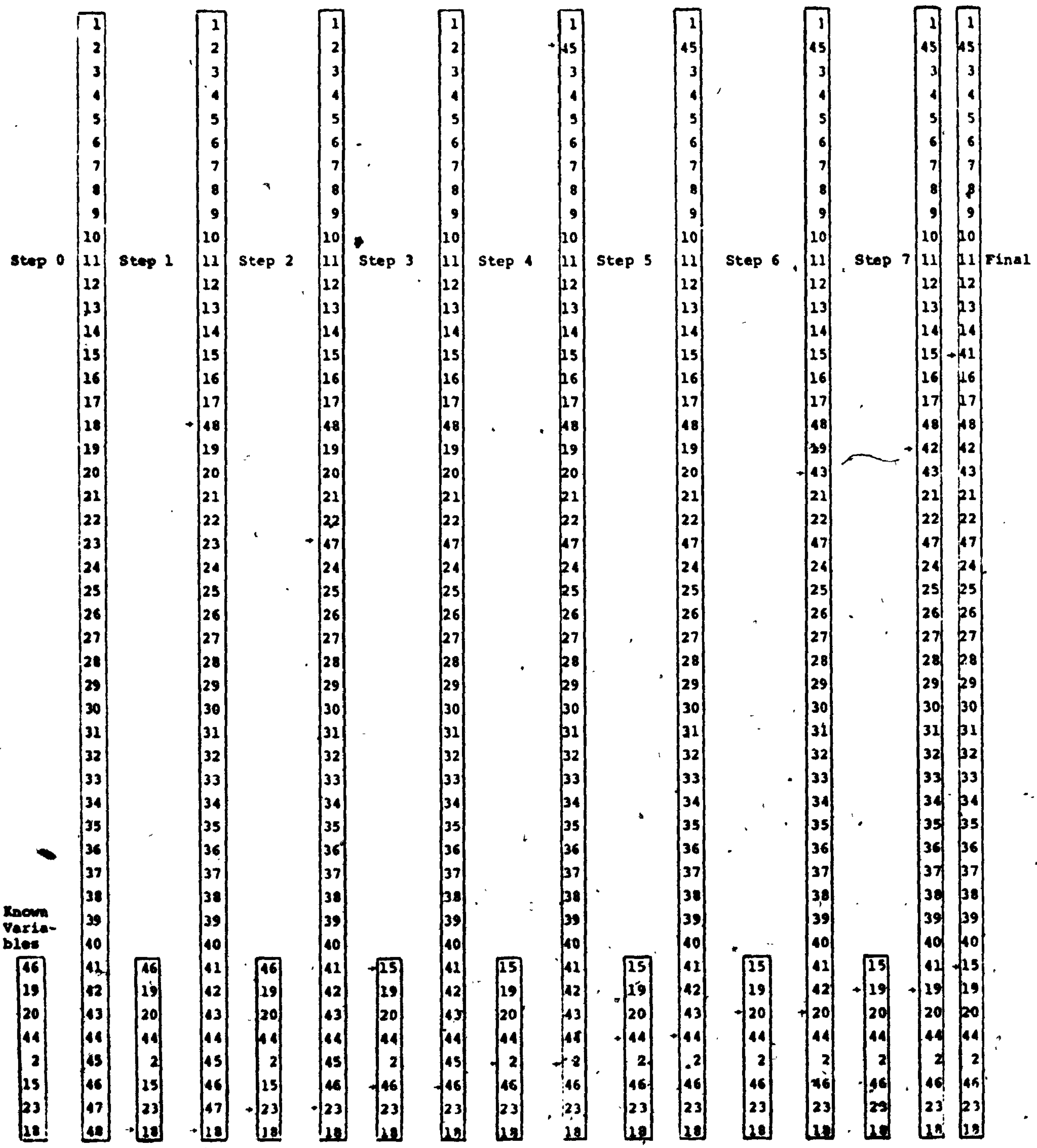


Figure IV.1 Sorting of Unknown and Known Variables

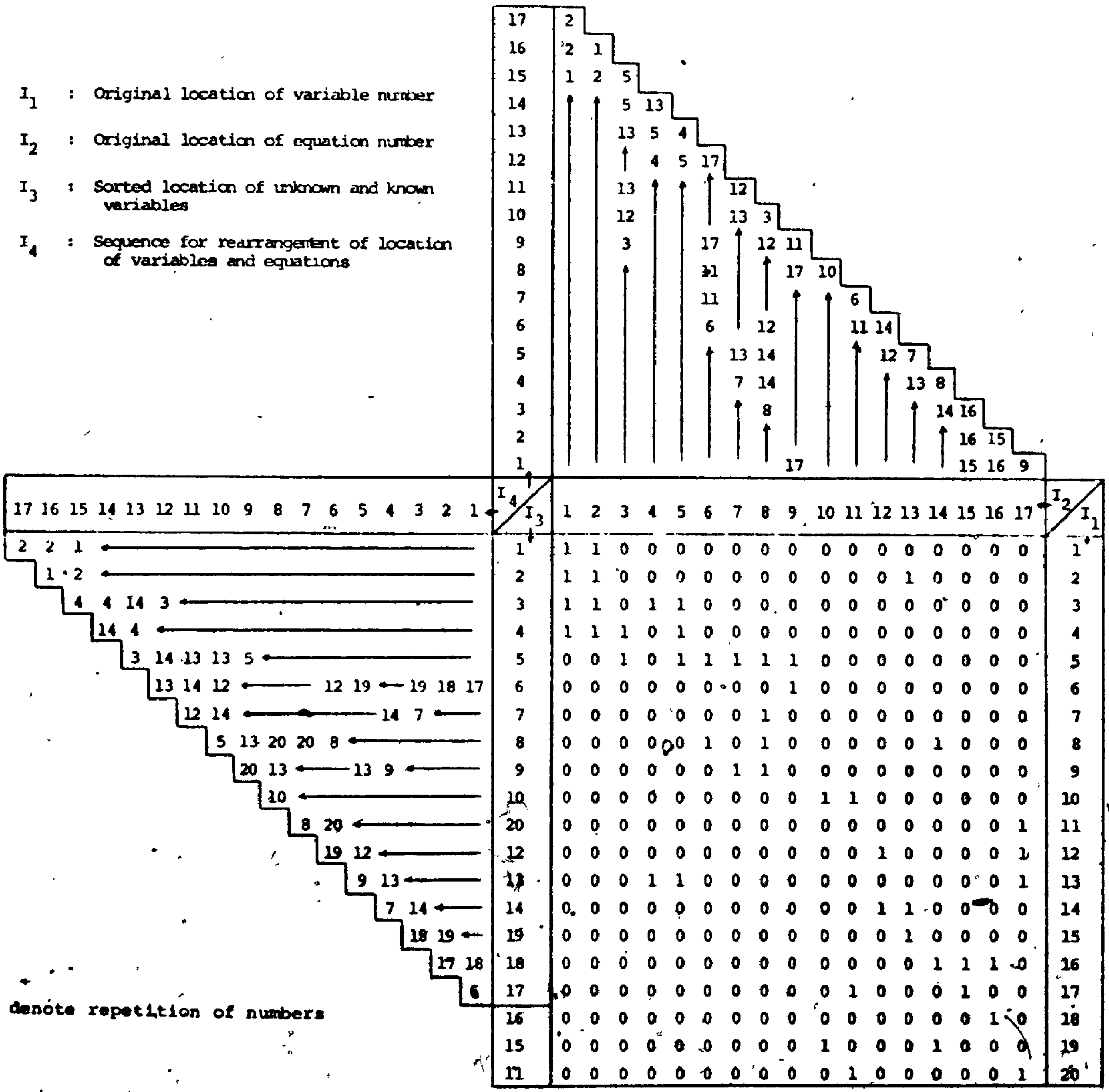


Figure IV.2 Sorting of Variables and Equations shown in Table III.5 in Appendix 3.

This pretreatment is done in subprogram PREPAR for which the program listing is given in Appendix 9.

APPENDIX 5

STORING THE COMPUTATION SEQUENCE
FOR ENERGY AND MATERIAL BALANCES

The calculation of energy and material balances occupies a significant part of the total computation time in an optimal design of an energy system. As mentioned in Chapter 2, the simultaneous equation set representing energy and material balances has a sparse matrix, so the extraction and repeated calculations of essential elements for solving the equation set serve to save computation time. This will be described here using a simple example.

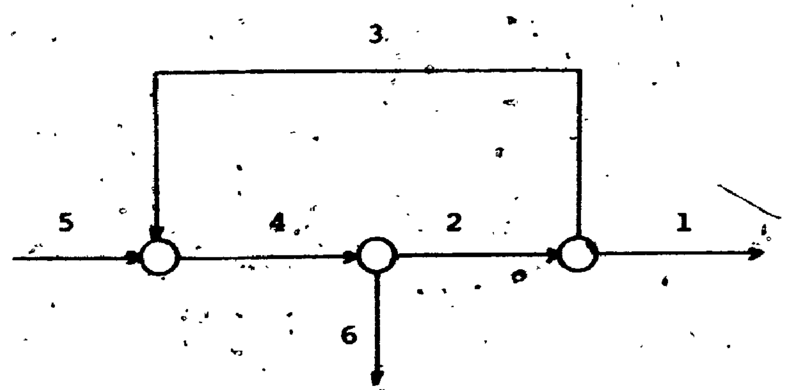


Figure V.1 Hypothetical System

For simplicity, a hypothetical system such as Figure V.1 is used and it is assumed that balance equations are obtained from this system as follows:

$$x_2 a_1 - x_1 = 0 \tag{5.1}$$

$$x_2 + x_6 - x_4 = 0 \tag{5.2}$$

$$x_1 + x_3 - x_2 = 0 \tag{5.3}$$

$$x_4 a_2 - x_6 = 0 \tag{5.4}$$

$$x_3 + x_5 - x_4 = 0 \tag{5.5}$$

where

x_6 is assumed to be given.

These equations are also expressed as follows:

$$\underline{X}' \underline{\underline{A}} = \underline{0}' \tag{5.6}$$

$$A = \begin{bmatrix} -1.0 & 0 & 1.0 & 0 & 0 \\ a_1 & 1.0 & -1.0 & 0 & 0 \\ 0 & 0 & 1.0 & 0 & 1.0 \\ 0 & -1.0 & 0 & a_2 & -1.0 \\ 0 & 0 & 0 & 0 & 1.0 \\ 0 & 1.0 & 0 & -1.0 & 0 \end{bmatrix} \tag{5.7}$$

These simultaneous equations can be solved as follows:

$$\left.\begin{aligned} b_{i,k} &= a_{i,k} - \sum_{j=1}^{k-1} b_{ij}c_{j,k}, \quad (i=k,\ldots 6) && (5.8) \\ c_{k,i} &= (a_{k,i} - \sum_{j=1}^{k-1} b_{k,j}c_{j,i})/b_{k,k}, \quad (i=k+1,..5) && (5.9) \end{aligned}\right\}(k=1,..5)$$

$$x_{n-k+1} = \sum_{j=n-k+2}^{6} x_j b_{j,n-k+1}/b_{n-k+1,n-k+1}, \quad (k=1,..5) \qquad (5.10)$$

Only essential components of the above computations can be extracted and stored by decomposing them into the computation sequence of the four rules of arithmetic such that:

1. E(I) = E(J) + E(K)
2. E(I) = E(J) - E(K)
3. E(I) = E(J) * E(K)
4. E(I) = E(J) / E(K)
5. E(I) = -E(J) / E(K)
6. E(I) = -E(J)
7. E(I) = E(J)

where I,J and K respectively point to each of the direct addressers where all actual values from state variables to the coefficient matrix are lined up in one dimension which is sketched in Figure V.2 and also matrix $\underline{\underline{A}}$ is overridden afterwards by matrices $\underline{\underline{B}}$ and $\underline{\underline{C}}$ for the purpose of saving data space. Final values of $\underline{\underline{B}}$ and $\underline{\underline{C}}$ are as follows:

$$\begin{bmatrix} -1.0 & 0 & -1.0 & 0.0 & 0 \\ a_1 & 1.0 & c_{2,3} & 0 & 0 \\ 0 & 0 & 1.0 & 0 & 1.0 \\ 0 & -1.0 & b_{4,3} & a_2 & c_{4,5} \\ 0 & 0 & 0 & 0 & 1.0 \\ 0 & 1.0 & b_{6,3} & -1.0 & b_{6,5} \end{bmatrix}$$

($\underline{\underline{C}}$: upper part above the staircase line; $\underline{\underline{B}}$: lower part)

where

$$c_{2,3} = -1.0 - a_1$$

$$b_{4,3} = c_{2,3} = -b_{6,3}$$

$$c_{4,5} = (-1.0 - b_{4,3})/a_2$$

$$b_{6,5} = -(b_{6,3} - c_{4,5})$$

A set of computation sequence for the example is given in Table V.1. The numerical operations described above are done in subprogram TRIAN listed in Appendix 9.

1	2	......	m	m+1	m+2	m+3	m+4		mxn+m+2
x_1	x_2	......	x_m	w_1	w_2	$A_{1,1}$ ($B_{1,1}$)	$A_{2,1}$ ($B_{2,1}$)	......	$A_{m,n}$ ($B_{m,n}$)

where w_1 and w_2 are working areas

Figure V.2 A Physical Structure of Variables

	L	I	J	K	Calculation
1	4	21	21	9	$c_{1,3} = a_{1,3}/b_{1,1}$
2	3	7	10	21	$w_1 = b_{2,1} * c_{1,3}$
3	1	22	22	7	$c_{2,3} = a_{2,3} + w_1$
4	3	7	18	22	$w_1 = b_{4,2} * c_{2,3}$
5	6	24	7	—	$b_{4,3} = -w_1$
6	7	7	22	—	$w_1 = c_{2,3}$
7	6	26	7	—	$b_{6,3} = -w_1$
8	7	7	24	—	$w_1 = b_{4,3}$
9	2	7	36	7	$w_1 = a_{45} - w_1$
10	4	36	7	30	$c_{4,5} = w_1/b_{4,4}$
11	7	7	26	—	$w_1 = b_{6,3}$
12	7	8	32	36	$w_2 = b_{6,4} * c_{4,5}$
13	1	7	7	8	$w_1 = w_1 + w_2$
14	6	38	7	—	$b_{6,5} = -w_1$
15	3	7	6	38	$w_1 = x_6 * b_{6,5}$
16	6	5	7	—	$x_5 = -w_1$
17	3	7	6	32	$w_1 = x_6 * b_{6,4}$
18	5	4	7	30	$x_4 = -w_1/b_{4,4}$
19	3	7	4	24	$w_1 = x_{4,} * b_{4,3}$
20	3	8	6	26	$w_2 = x_6 * b_{6,3}$
21	1	7	7	8	$w_1 = w_1 + w_2$
22	6	3	7	—	$x_3 = -w_1$
23	3	7	4	18	$w_1 = x_4 * b_{4,2}$
24	7	8	6	—	$w_2 = x_6$
25	1	7	7	8	$w_1 = w_1 + w_2$
26	6	2	7	—	$x_2 = -w_1$
27	3	7	2	10	$w_1 = x_2 * b_{2,1}$
28	6	1	7	9	$x_1 = -w_1/b_{1,1}$

Table V.1 A Computation Sequence Set

APPENDIX 6

IMPLEMENTATION OF THE OPTIMIZATION TECHNIQUE

As described in Chapter 2, the Complex method was chosen as a search technique to seek an optimal design condition set in arbitrary energy systems. This appendix gives a brief description for the Complex method and describes how it was implemented in "ODES".

6.1 Complex Method

Box (1965) has given an alternative version of the Simplex Method which searches for the minimum of an n-variable function using simplex and takes into account inequality constraints of the form

$$g_i \leq x_i \leq s_i \qquad (i = 1, ..n) \qquad (6.1)$$

and $$G_j \leq \phi_j (\underline{X}) \leq S_j, \qquad (j = 1, ..m) \qquad (6.2)$$

where g_i, s_i, G_j, and S_j, are either constants or functions of $\underline{X}$.

This variant is called the Complex method.

The main features of this method are as follows:

1. It uses a non-regular complex with $k>n+1$ points instead of $k=n+1$ as in the simplex method. The complex with $k=n+1$ points (vertices) shows a tendency to collapse into subspace, particularly when it approaches constraints. This variation especially has important meaning when numbers of variables are small.

2. A basic operation used in the Complex method is an over-reflection defined by:

$$\underline{X} = (1 + \alpha)\, \underline{X}_O - \alpha \underline{X}_W \qquad (6.3)$$

where $\alpha \geq 1$,

$\underline{X}_O$ is the centroid

and $\underline{X}_W$ is the worst point in the complex.

The method works in the following fashion:

(i) The initial complex are generated by the formula:

$$X_i = g_i + \gamma_i (s_i - g_i) \qquad (i = 1, ..n) \qquad (6.4)$$

where the γ_i are pseudo-random numbers evenly distributed in the internal [0,1]

(ii) The over-reflection move is applied. If the trial point again gives the highest function value then it is moved half-way toward the centroid of the remaining points to give a new trial point. This procedure is repeated until a constraint is violated.

(iii) If over-reflection produces a point which violates a constraint (1) or (2), then the point is moved half-way toward the centroid of the remaining points and ultimately a feasible point is located.

6.2 Implementation of the Complex Method

The subprogram OPTIM was written for seeking an optimal point using the Complex method. Also, the subprogram SETVAL was written for setting up design variables from independent variables and checking explicit constraints (6.1). The following shows a basic implementation of the optimization routine:

```
10   CALL OPTIM(NV,X,U,RMIN,RMAX,RLI,RUI,G..,ICHECK,..)
     CALL SETVAL (KOUNT,ICHECK, ....)
     IF (ICHECK.LT.0) GO TO 10
     CALL OBJECT (U,..)
     IF (II.EQ.0) GO TO 10
```

where OPTIM controls optimization step. SETVAL sets up design variables from (Xi i=1,n) generated by optimization step and checks explicit constraints (6.1). If ICHECK is defined between calling of SET[illegible] and OBJECT as the negative which means violation of implicit constraints (6.2), the control has to be sent back to the statement number 10. OBJECT computes the objective function U and the control is sent back to OPTIM unless II is 1 which indicates convergence of optimization. Relative change of objective functions in the complex is employed as a stopping criterion.

The actual implementation of this routine can be found in the subprogram "MAIN" listed in Appendix 9.

APPENDIX 7

DATA SOPHISTICATION
in LP FORMULATION

In formulating an LP problem generally, there are two items which should be taken into consideration, i.e. input data format for MPSX of IBM and variable identification in LP modeling and LP solution. Data sophistication in LP formulation plays an important role in taking the above two items into account.

Input data to MPSX consists of the following four parts that can be seen in the input example in Appendix 8:

- Row specification
- Column specification
- Right hand side (RHS) specification
- Upper and lower bound specification.

The format and sequence for these data must be followed exactly. These limitations require sophisticated variable codes which were defined in Table 3.1 in Chapter 3 so that a systematic sorting of data can be performed making use of variable codes.

To illustrate data sophistication in LP formulation the first equation that was derived for a back-pressure turbine unit in Chapter 3 is taken up as an example. Namely, the first equation is:

$$X_1 - X_2 = 0$$

This equation is actually stored into a data set as follows:

IROW(N) = M + L x 100 + K x 10000 + JCODE x 10000000

ICOL(N) = I_1 + L x 100 + ICODE x 10000000

RDATA(N) = 1.0

NEQ(MEQ) = IROW(N)

IROW(N+1) = IROW(N)

ICOL(N+1) = I_2 + L x 100 + ICODE x 10000000

RDATA(N+1) = -1.0

where

JCODE is 1 and ICODE is 1

The multiplications by powers of ten are for the purpose of packing 4 pieces of information in one machine location.

After creating whole set of data, input data to MPSX are generated according to the following procedure:

1) Data set {NEQ(MEQ)} for Row specification.

2) Code ICODE for column is checked for the number less than 6 and sorted in order for column specification.

3) Code ICODE for column is checked for data having code 6 and sorted in order for RHS specification.

4) Code ICODE for column is checked for the number greater than 6 and Code JCODE for row is checked for the number having 5 and 6 for lower and upper bounds specification.

Data sets which were created as shown above are printed out according to the sequence by which the equations were defined, therefore, model checking is quite easy because variables and equations are clearly identifiable according to the agreement defined in Table 3.1 in chapter 3 and the information flow diagram. Similarly, data checking for MPSX and analysis of results obtained can be achieved easily.

Sorting of data set and data generation for MPSX is done in subprogram LPSORT and OUTPUT respectively, listed in Appendix 10.

APPENDIX 8

PROGRAM INPUT FORMATS FOR "ODES", "OPES" AND "OSES"

8.1 Input Format for "ODES"

The input data to "ODES" consist mainly of the following three groups of data:

1) Stream Codes

2) Data which represent a configuration of energy systems and comprise a set of keyword identifier and information streams.

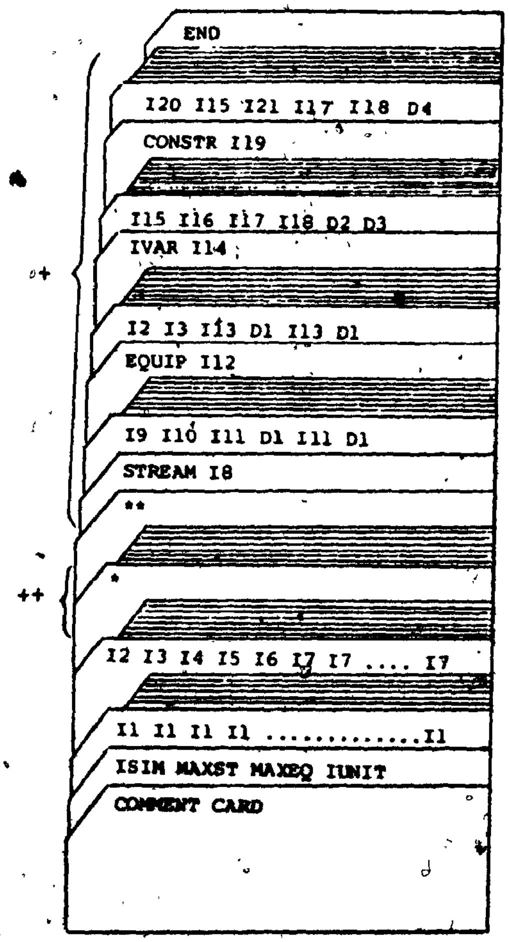
3) Data for constant parameters for streams and equipments.

When optimization is involved, the following data is added:

4) Data for designation of independent variables and constrained variables

Figure VIII.1 shows the format of a set of input data.

General constant data, that are another kind of input data, usually do not have to be entered; however, if it is necessary to change default values for general constant



+ Sequence of these block data does not matter

++ If process system is included

Variable	Format	Description
ISIM	(I5)	1 : Linear balance
		2 : Nonlinear balance
		3 : Cost estimation
		4 : Optimization
MAXST	(I5)	Max. stream number
MAXEQ	(I5)	Max. equipment number
IUNIT	(I5)	1 : c.g.s. unit
		2 : Btu-lb unit
*I1	(I5)	Stream code for each stream (refer to Table 2.1)
I2	(A10)	Keyword for module
I3	(I5)	Equipment no.
I4	(I5)	No. of equipment data
I5	(I5)	No. of input stream
I6	(I5)	No. of output stream
I7	(I5)	Stream no. associated
I8	(I5)	No. of stream data designated
I9	(A10)	Identification name for stream
I10	(I5)	Stream number
I11	(I5)	Attribute no. of stream
D1	(F10.2)	Actual value designated
I12	(I5)	No. of equipment data designated
I13	(I5)	Attribute no. of equipment
I14	(I5)	No. of independent variables
I15	(10X,I5)	Independent variable no.
I16	(I5)	1 : I.V. is stream parameter
		2 : I.V. is equipment parameter
I17	(I5)	Equipment no. or stream no.
I18	(I5)	Attribute no. of stream or equipment
D2	(F10.2)	[illegible] limit of I.V. for optimization
D3	(F10.2)	Upper limit of I.V. for optimization
I19	(I5)	Number of constraints
I20	(10X,I5)	0 : Equality constraint
		1 : Less inequality
		2 : Greater inequality
D4	(F10.2)	Constant term in constraint
I21	(I5)	1 : Constraint is stream parameter
		2 : Constraint is equip. parameter

Figure VIII.1 An Example of a Set of Input Data to "OCES" data, for instance, cost data, maximum number of streams and units which can be dealt with by the program system and so on, it is possible through the main program "ODES" as seen in Listing VIII.1 in which variable descriptions are given.

A simple energy system shown in Figure VIII.2 is taken as an example for the use of "ODES" package from an optimal design problem. Table VIII.1 presents steam and power demands for Figure VIII.2. By preliminary analysis of this system, the variables listed in Table VIII.2 are chosen as independent variables for the optimization. This information is transformed into input data to "ODES" shown in Listing VIII.2.

Computational results are given in Listing VIII.3. Listing VIII.4 shows iterative values for objective function and independent variables in optimization.

8.2 Input Format for "OPES" and "OSES"

The input format to "OPES" are exactly the same as that to "OSES". Also, the input data to "OPES" and "OSES" are not very much different from "ODES" and consist mainly of the following four different groups of data:

1) Stream codes

LISTING VIII.1 MAIN PROGRAM OF "ODES"

```
      PROGRAM ODES(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C     PROGRAM FOR OPTIMAL DESIGN OF ENERGY SYSTEMS
C     /SYS1/   SNEN(K)   ... REFER TO TABLE 2.2
C              IS(I,K)   ... REFER TO TABLE 2.1
C              IE(J,K)   ... REFER TO TABLE 2.1
C     /SYS2/   LENGS(I)  ... LENGTH OF REAL STREAM DATA FOR EACH STREAM CODE I
C     /SYS3/   NM        ... NO. OF REGISTERED KEYWORDS
C              KYWRD(I)  ... REGISTERED KEYWORDS FOR UNIT MODULES
C     /OBJ1/   JIN       ... NO. OF SYSTEM INPUT STREAMS
C              IIN(I)    ... SYSTEM INPUT STREAM NUMBER
C     /OBJ2/   JOUT      ... NO. OF SYSTEM OUTPUT STREAMS
C              IOUT(I)   ... SYSTEM OUTPUT STREAM NUMBER
C     /OBJ3/   CUST(I)   ... UNIT PRICE FOR EACH REGISTERED STREAM
C                            4 - FUEL OIL
C                            5 - FUEL GAS
C                            6 - INDUSTRIAL WATER
C                            7 - COOLING WATER
C                            8 - POWER
C                            9 - ELECTRICITY
C                            10- PURE WATER
C                            13- LOW PRESSURE STEAM
C                            14- MEDIUM PRESSUE STEAM
C                            15- HIGH PRESSURE STEAM
C     /OBJ4/   ECON(I)   ... ECONOMIC DATA FOR OPTIMIZATION
C                            5 - OPERATING DAYS
C                            6 - LANG FACTOR
C                            7 - DEPRECIATION + RATE OF RETURN*(1+IW)/(1-TAX)
C                                IW - RATIO OF WORKING CAPITAL TO INVESTMENT
C                            8 - INCOME TAX
C                            9 - LABOR AND SUPERVISION
C                            10- RATE OF EARNINGS
C     /OBJ5/   CRIT      ... OPTION CODE FOR OBJECTIVE FUNCTION,1 - VENTURE COST
C                            2 - VENTURE PROFIT 3 - OPERATING COST
C     /SIM1/   F(I)      ... MASS OR ENERGY FLOW RATE
C              WORK(2)   ... WORKING AREAS
C              A(I,J)    ... COEFFICIENT MATRIX FOR SIMULTANEOUS LINEAR EQUATIOS
C     /SIM2/   IL(I)     ... ESSENTIAL INFORMATION NECESSARY FOR SOLVING
```

LISTING VIII.1 - CONTINUED

```
C     /SIM3/   II(I)       ... SIMULTANEOUS LINEAR EQUATIONS IS  STORED
C     /SIM4/   IJ(I)       ...
C     /SIM5/   IK(I)       ...
C     /SIM6/   NI(I)       ... INFORMATION SHOWING STRUCTURE OF SIMULTANEOUS
C     /SIM7/   NJ(I)       ... LINEAR EQUATIONS
C     /SIM8/   NK(I)       ...
C     /SIM9/   NS          ... NO. OF SPLITTERS
C              ISP(I)      ... EQUIPMENT NO. OF SPLITTER
C     /OPT1/   NV          ... NO. OF INDEPENDENT VARIABLES
C              X(I)        ... INDEPENDENT VARIABLES
C     /OPT2/   RMIN(I)     ... LOWER LIMIT FOR INDEPENDENT VARIABLES
C     /OPT3/   RMAX(I)     ... UPPER LIMIT FOR INDEPENDENT VARIABLES
C     /OPT4/   RLI(I)      ... LOWER LIMIT FOR AN INITIALIZATION OF I.D.
C     /OPT5/   RUI(I)      ... UPPER LIMIT FOR AN INITIALIZATION OF I.D.
C     /OPT6/   G           ... TOLERANCE FOR STOPPING CRITERION OF  OPTIMIZATION
C              FUNC(I)     ... FUNCTION VALUES FOR A COMPLEX
C              XA(I,J)     ... INDEPENDENT VARIABLE SET FOR A COMPLEX
C     /OPT7/   XO(I)       ... OLD VALUES OF INDEPENDENT VARIABLES
C     /OPT8/   XR(I)       ... CENTROID OF EACH INDEPENDENT VARIABLE IN A COMPLEX
C     /OPT9/   W(I)        ... RANDOM VARIABLES FOR AN INTIALIZATION OF COMPLEX
C     /OPT10/  IX(I)       ... INFORMATION FOR RELATION BETWEEN INDEPENDENT
C                              VARIABLE AND DESIGN VARIABLE
C     /OPT11/  NCD         ... NO. OF CONSTRAINTS
C              CNST(I)     ... CONSTANT TERM FOR CONSTRAINTS
C              ICON(I,3)   ... INFORMATION FOR RELATION BETWEEN CONSTRAINT
C                              DESIGN VARIABLES
C     /GENRL/  NCP         ... NO. OF COMPONENTS IF PROCESS SYSTEM IS INCLUDED
C              M           ... NO. OF VARIABLES IN SIMULTANEOUS EQUATIONS
C              N           ... NO. OF EQUATIONS IN SIMULTANEOUS EQUATIONS
C              IEQ         ... EQUIPMENT NO. OF MODULES CALCULATED
C              MAXST       ... MAXIMUM STREAM NO. DEALT WITH IN THE SYSTEM
C              IUNIT       ... CODE FOR UNIT CONVERSION , 1 - C.G.S 2 - BTU-LB
C              INV         ... NO. OF KNOWN VARIABLES SIMULTANEOUS EQUATIONS
C              LOOP        ... ITERATION NO. IN NON-LINEAR BALANCE CALCULATIONS
C              ISIM        ... SIMULATION CODE
C                              1 - LINEAR BALANCE 2 - NONLINEAR BALANCE
C                              3 - COST EVALUATION 4 - OPTIMIZATION
C              I1          ... NO. OF PROCESS MODULES TO BE CALCULATED
C              I2          ... TOTAL NO. OF MODULES TO BE CALCULATED
```

```
      LISTING VIII.1 - CONTINUED
C             IPOINT     ... MAXIMUM NO. OF SNEN DATA
C     NST                ... MAXIMUM LIMIT OF STREAM NO. TO BE STORED
C     NEN                ... MAXIMUM LIMIT OF EQUIPMENT NO. TO BE STORED
C     NEQ                ... MAXIMUM LIMIT OF EQUATIONS TO BE STORED
C     NIS                ... MAXIMUM LIMIT OF ATTRIBUTE OF IS DATA
C     NIE                ... MAXIMUM LIMIT OF ATTRIBUTE OF IE DATA
C
      COMMON /SYS1/SNEN(400),IS(80,2),IE(40,16)/SYS2/LENGS(16)
      COMMON /SYS3/NM,KYWRD(30)/OBJ1/JIN,IIN(10)/OBJ2/JOUT,IOUT(15)
      COMMON /OBJ3/COST(16)/OBJ4/ECON(20)/OBJ5/CRIT
      COMMON /SIM1/F(80),WORK(2),A(80,60)/SIM2/IL(4120)/SIM3/II(4120)
      COMMON /SIM4/IJ(4120)/SIM5/IK(4120)/SIM6/NI(80)/SIM7/NJ(80)
      COMMON /SIM8/NK(80)/SIM9/NS,ISP(15)
      COMMON /OPT1/MAXM,X(13)/OPT2/RMIN(13)/OPT3/RMAX(13)/OPT4/RLI(13)
      COMMON /OPT5/RUI(13)/OPT6/G,FUNC(14),XA(13,14)/OPT7/XO(13)
      COMMON /OPT8/XR(13)/OPT9/R(13)/OPT10/IX(13)/OPT11/NCD,CNST(20)
     1        ,ICON(20,3)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM,I1,I2,IPOINT
C     REGISTERED KEYWORD
      DATA  KYWRD/  4HJUNC,6HERECIV, 5HDEAER,  4HPUMP,  4HFURN, 5HSPLTR
     1          , 5HJUNC1,   3HWHB, 5HFDRUM, 5HTURBG, 5HTURBB, 5HVALVE
     1          , 5HTURBC, 5HCONST,6HHEATEX, 5HSCOND, 5HBOILR,6HETURB1
     1          ,6HETURB2, 5HDESUP,6HGTURBB,6HGTURB1,6HGTURB2,6HGTURBC
     1          ,6HGTURC1,6HDEAER1, 5HEQUAL,6HWTREAT, 5HMOTOR, 5HHEATR/
C     GENERAL DATA
      DATA NST,NEN,NEQ,NIS,NIE,NM,NC,NN,G /80,40,60,2,16,30,10, 5,0.001/
      DATA CRIT/1/
      DATA LENGS/3*14,4*4,1,1,7*4/
      DATA COST/3*0.,38.,28.35,0.005,0.010,0.013,0.013,0.1,3*0.,3*2.5/
      DATA ECON(5),ECON(6),ECON(7),ECON(8),ECON(9)/8.76,2.,0.545,.5,70./
C
   10 CALL        INPUT(IS,IE,NST,NEN,NIS,NIE,NV,MAXM,NC,ICON)
      IF(NV.EQ.0) GO TO 30
      DO 20 I=1,NV
      RMAX(I)=RUI(I)
   20 RMIN(I)=RLI(I)
   30 CALL        MAIN  (A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICON,NC,XA,NV,NN)
      STOP
      END
```

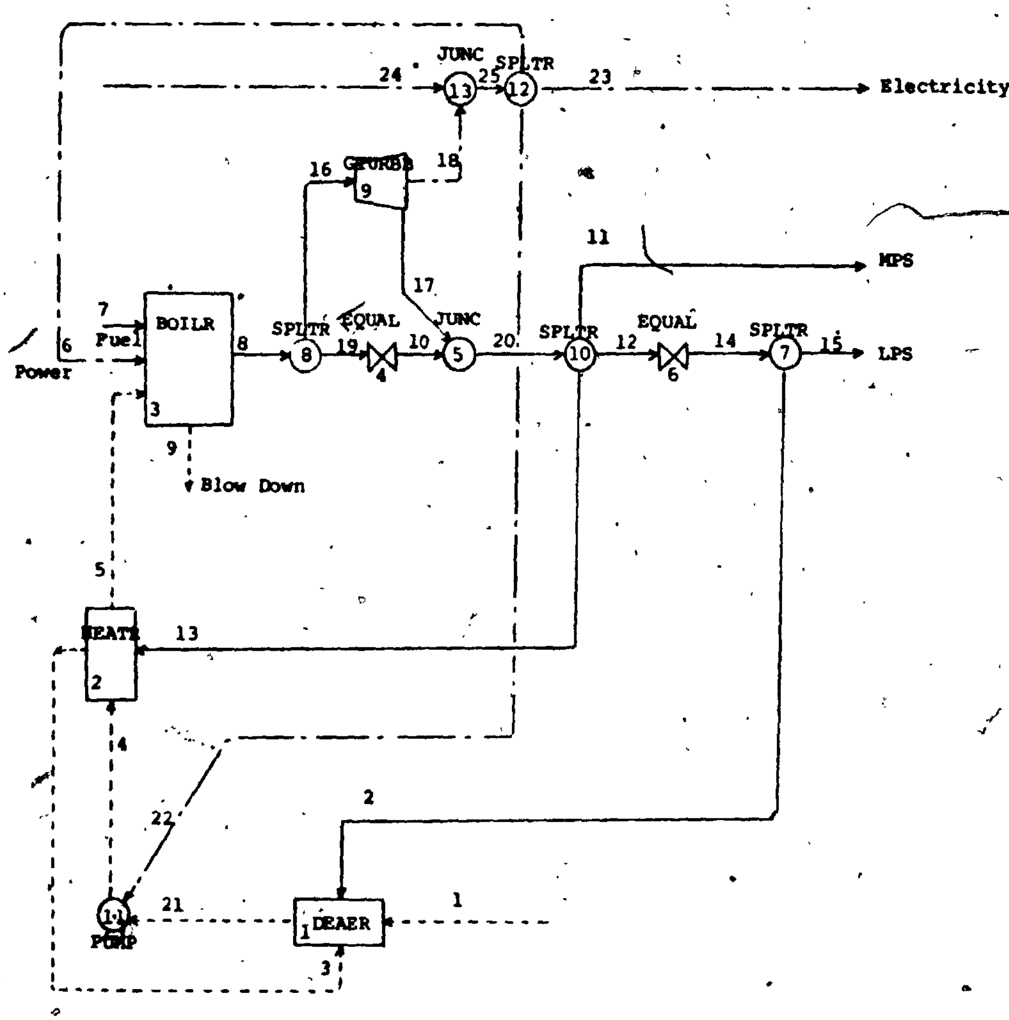


Figure VIII.2 Information Flow Diagram of a Simple Steam and Power System

Demand Item	Stream No.	Demands	Temp.	Press.
Electricity	23	27000 KWH	-	-
Medium Pressure Steam	11	1100 Mlb/hr	500 °F	179.7 psia
Low Pressure Steam	15	139.3 Mlb/hr	300 °F	29.7 psia

Table VIII.1 Steam and Power Demands for a Simple Steam and Power System

I.V. No.	Variable Description	Lower Bound	Upper Bound
1	Temperature of Stream 5	400 °F	490
2	Temperature of Stream 8	600 °F	1000
3	Pressure of Stream 8	500 psia	1000
4	Flowrate of Stream 16	400 Mlb/hr	800
	Constraint		psia
	Pressure of Stream 8 =	Pressure of Stream 4 - 50	

Table VIII.2 Independent Variables and Constraints

```
LISTING VIII.2 INPUT DATA TO "ODES"

** OPTIMAL DESIGN OF STEAM AND POWER PLANT **

     ISIM     MAXST     MAXEQ     IUNIT
        4        25        13         2

STREAM CODE
   10   12   10   10   10    8    4   14   10   13   13   13   13   12   12   14
   13    9   14   13   10    8    9    9    9

                        NO. OF I/O STREAMS   STREAM NUMERS RELATING TO 'EQUIPMENT'
SYSTEM NETWORK
DEAER1         1    3    3    1    1    2    3   21    0    0    0    0    0    0
HEATR          2    4    2    2   13    4    3    5    0    0    0    0    0    0
BOILR          3    6    3    2    5    6    7    8    9    0    0    0    0    0
EQUAL          4    1    1    1   19   10    0    0    0    0    0    0    0    0
JUNC           5    1    2    1   10   17   20    0    0    0    0    0    0    0
EQUAL          6    1    1    1   12   14    0    0    0    0    0    0    0    0
SPLTR          7    1    1    2   14   15    2    0    0    0    0    0    0    0
SPLTR          8    1    1    2    8   16   19    0    0    0    0    0    0    0
GTURBB         9    4    2    1   16   17   18    0    0    0    0    0    0    0
SPLTR         10    1    1    3   20   11   12   13    0    0    0    0    0    0
PUMP          11    3    2    1   21   22    4    0    0    0    0    0    0    0
SPLTR         12    1    1    3   25    6   22   23    0    0    0    0    0    0
JUNC          13    1    2    1   18   24   25    0    0    0    0    0    0    0
**             0    0    0    0    0    0    0    0    0    0    0    0    0    0

STREAM DATA         9     TEMP.          PRESS.
STREAM         1    2    170.00    3     49.70    0    0.00    0    0.00
STREAM         3    2    286.00    0      0.00    0    0.00    0    0.00
STREAM         4    2    250.00    3     29.70    0    0.00    0    0.00
STREAM        10    2    500.00    3    179.70    0    0.00    0    0.00
STREAM        14    2    300.00    3     29.70    0    0.00    0    0.00
STREAM        11    1   1100.00    0      0.00    0    0.00    0    0.00
STREAM        15    1    139.30    0      0.00    0    0.00    0    0.00
STREAM        21    2    250.00    3     29.70    0    0.00    0    0.00
STREAM        23    1  27000.00    0      0.00    0    0.00    0    0.00
```

LISTING VIII.2 - CONTINUED

```
EQUIPMENT DATA       2
HEATR          2     4     500.00     0      0.00     0      0.00     0      0.00
BOILR          3     4       .05      5  18900.00     6      1.65     0      0.00

INDEPENDENT VARIABLES        100
     1     1     5     2     400.00    490.00
     2     1     8     2     600.00   1000.00
     3     1     8     3     500.00   1000.00
     4     1    16     1     400.00    800.00

CONSTRAINTS
     0     2     1     4     3    -50.00
```

LISTING VIII.3 COMPUTATIONAL RESULTS FOR THE EXAMPLE

OPTIMAL SOLUTION	
OBJECTIVE FUNCTION	-44489.7
TOTAL ERECTED COST	24407.1
OPERATING COST	31187.8
FUEL GAS	0.0
FUEL OIL	28701.9
COOLING WATER	0.0
ELECTRICITY	1251.9
INDUSTRIAL WATER	0.0
BOILER FEED WATER	1164.0
LABOR $ SUPERVISION	70.0

EQUIPMENT DATA	13	COST					
DEAERI	1	12.00	1788.26	0.00			
HEATR	2	4.13	81.12	0.00	500.00		
BOILR	3	10549.40	1698.85	0.00	.05	18900.00	1.65
EQUAL	4	0.00					
JUNC	5	0.00					
EQUAL	6	0.00					
SPLTR	7	0.00					
SPLTR	8	0.00					
GTURBB	9	1631.60	19496.11	0.00	0.00		
SPLTR	10	0.00					
PUMP	11	6.44	1788.26	0.00			
SPLTR	12	0.00					
JUNC	13	0.00					

LISTING VIII.3 - CONTINUED

STREAM DATA FLOW RATE

1	1358.71	170.00	49.70	0.00
2	96.59	300.00	29.70	0.00
3	362.95	286.00	0.00	0.00
4	1788.26	250.00	654.52	0.00
5	1788.26	446.69	0.00	0.00
6	2950.63			
7	86.22	0.00	0.00	0.00
8	1698.85	604.52	681.44	0.00
9	89.41	0.00	0.00	0.00
10	909.99	500.00	179.70	0.00
11	1100.00	500.00	179.70	0.00
12	235.89	500.00	179.70	0.00
13	362.95	500.00	179.70	0.00
14	235.89	300.00	29.70	0.00
15	139.30	300.00	29.70	0.00
16	788.86	604.52	681.44	0.00
17	788.86	500.00	179.70	0.00
18	19496.11			
19	909.99	604.52	681.44	0.00
20	1698.85	500.00	179.70	0.00
21	1788.26	250.00	29.70	0.00
22	1481.39			
23	27000.00			
24	10992.96			
25	30489.06			

LISTING VIII.4 ITERATIVE VALUES FOR OBJECTIVE FUNCTION AND INDEPENDENT VARIABLES

ITERATION NUMBER	Objective Function	Variable-1	Variable-2	Variable-3	Variable-4	Operating Cost	Erected Cost
1	-50316.87	400.00	966.90	728.76	774.78	36835.07	24737.25
2	-50072.64	472.59	850.26	963.39	401.61	36707.03	24524.05
3	-49932.02	400.36	898.36	551.40	572.80	37223.72	23317.99
4	-45270.65	440.45	639.47	685.69	677.23	32122.42	24125.20
5	-51588.89	435.37	973.79	993.75	432.72	38406.30	24188.25
6	-47795.45	424.84	771.22	601.59	693.55	34602.29	24097.54
7	-47004.45	451.84	701.29	686.40	492.05	34018.43	23827.55
8	-47100.98	418.57	728.17	548.24	660.73	34236.85	23603.91
9	-44993.47	450.71	615.88	670.02	659.94	31639.62	24135.51
10	-45165.24	448.17	621.19	670.59	586.96	32159.06	23864.54
11	-44927.19	462.40	602.60	743.14	575.70	32201.83	23349.29
12	-44798.39	450.26	609.59	693.11	641.57	31603.13	24064.69
13	-44829.98	455.99	605.53	696.34	600.75	31738.41	24021.24
14	-44723.20	458.18	602.00	715.69	635.76	31628.56	24026.86
15	-44766.52	458.21	602.19	722.58	601.82	31766.72	23852.84
16	-44768.29	448.91	607.06	670.72	664.24	31645.08	24079.28

LISTING VIII.4 - CONTINUED

ITERATION NUMBER

17	-44655.40	451.78	604.90	704.70	670.95	31503.23	24132.58
18	-44680.16	456.28	601.26	708.56	644.00	31573.67	24048.61
19	-44808.00	459.71	600.36	733.97	625.08	31999.45	23501.92
20	-44727.47	457.91	601.47	723.43	631.60	31712.25	23881.15
21	-44621.69	453.87	602.62	703.62	689.34	31434.66	24196.34
22	-44646.98	452.14	603.92	692.66	688.42	31455.73	24204.13
23	-44593.99	448.86	604.35	689.19	710.60	31398.39	24212.12
24	-44592.31	447.05	606.64	686.61	735.65	31364.66	24270.91
25	-44571.43	449.17	603.87	681.44	741.05	31322.90	24309.23
26	-44537.25	447.33	604.82	687.57	749.90	31290.79	24305.44
27	-44543.16	442.34	607.21	668.79	779.26	31301.63	24296.38
28	-44528.03	444.09	606.92	673.01	792.33	31244.13	24374.13
29	-44496.85	444.42	604.77	668.79	795.62	31212.85	24374.31
30	-44503.14	442.23	606.96	671.09	798.39	31230.85	24352.83
31	-44489.65	446.69	604.52	681.44	788.86	31187.76	24407.13
32	-44500.22	443.98	605.91	671.83	799.29	31209.[illegible]	243[illegible]6.02
32	-44489.65	446.69	604.52	681.44	788.86	31187.76	24407.13

2) Data which represent a basic structure of energy systems and comprise a set of keyword identifier and information streams.

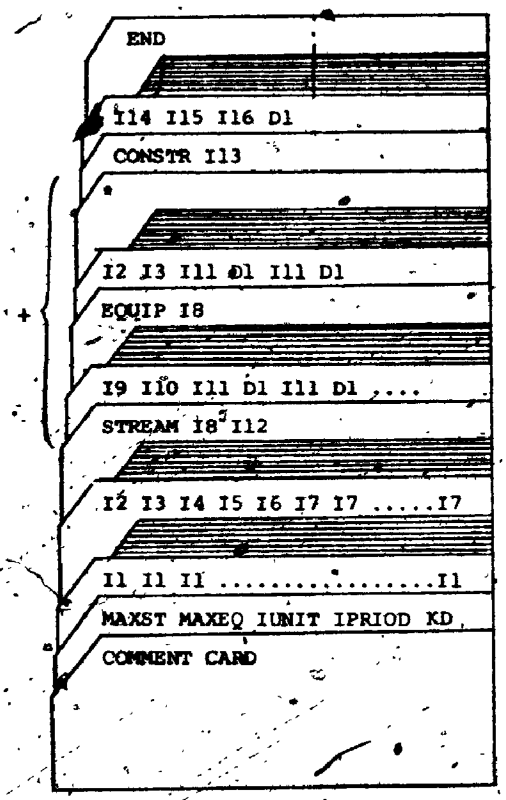
3) Data for constant parameters for stream and equipment including energy demands and existing capacity in case of expansion problem.

4) Data for the designation of constraints on flow rate or capacity to be expanded.

Figure VIII.3 shows the format of a set of input data.

Features concerning general constant data are the same as those for "ODES"; if it is required to change default values for general constant data, it is possible through the main program as seen in Listing VIII.5 in which variable descriptions are given.

An input example for an expansion problem is taken up for the first example in applications given in Chapter 6 and shown in Listing VIII.6. The program system "OPES" generates LP models and the sample output are given in Listing VIII.7. These models are automatically sorted out in accordance with the input format of MPSX on IBM. A typical set of control cards for MPSX run is seen in Listing VIII.8.



+ These sets of data are repeated for the period considered

Variable	Format	Description
MAXST	(I5)	Max. stream number
MAXEQ	(I5)	Max. equipment number
IUNIT	(I5)	1: c.g.s. unit
		2: BTU-lb unit
IPRIOD	(I5)	Expansion period considered
KD	(I5)	0: Expansion problem
		1: Synthesis problem
I1	(I5)	Stream code for each stream (refer to Table 2.1)
I2	(A10)	Keyword for module
I3	(I5)	Equipment number
I4	(I5)	No. of equipment data
I5	(I5)	No. of input stream
I6	(I5)	No. of output stream
I7	(I5)	Input & output stream associated
I8	(I5)	No. of stream data specified
I9	(A10)	Identification name for stream
I10	(I5)	Stream number
I11	(I5)	Attribute no. of stream or equipment
D1	(F10.2)	Real value specified
I12	(I5)	Period considered
I13	(I5)	No. of constraints
I14	(I5)	0: Equality constraint
		1: Less inequality
		2: Greater inequality
I15	(I5)	Stream or Equipment no.
I16	(I5)	1: Constraint on flow rate
		2: Constraint on capacity

Figure VIII.3 An Example of a Set of Input Data to "OSES" and "OPES"

3

LISTING VIII.5 MAIN PROGRAM OF "OSES" AND "OPES"

```
      PROGRAM OSES(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7)
C     PROGRAM FOR OPTIMAL SYNTHESIS OF ENERGY SYSTEMS
C     /SYS1/  SNEN(K)   ... REFER TO TABLE 2.2
C             IS(I,K)   ... REFER TO TABLE 2.1
C             IE(J,K)   ... REFER TO TABLE 2.1
C     /SYS2/  LENGS(I)  ... LENGTH OF REAL STREAM DATA FOR EACH STREAM CODE I
C     /SYS3/  NM        ... NO. OF REGISTERED KEYWORDS
C             KYWRD(I)  ... REGISTERED KEYWORDS FOR UNIT MODULES
C     /SYS4/  NJ(I)     ... WORKING AREA TO FIND OUT SYSTEM INPUT STREAMS
C     /SYS5/  NS        ... NO. OF SPLITTERS
C             ISP(I)    ... EQUIPMENT NO. OF SPLITTER
C     /OBJ1/  JIN       ... NO. OF SYSTEM INPUT STREAMS
C             IIN(I)    ... SYSTEM INPUT STREAM NUMBER
C     /OBJ2/  JOUT      ... NO. OF SYSTEM OUTPUT STREAMS
C             IOUT(I)   ... SYSTEM OUTPUT STREAM NUMBER
C     /OBJ3/  COST(I)   ... UNIT PRICE FOR EACH REGISTERED STREAM
C                           4 - FUEL OIL
C                           5 - FUEL GAS
C                           6 - INDUSTRIAL WATER
C                           7 - COOLING WATER
                            8 - POWER
C                           9 - ELECTRICITY
C                           10- PURE WATER
C                           13- LOW PRESSURE STEAM
C                           14- MEDIUM PRESSUE STEAM
C                           15- HIGH PRESSURE STEAM
C     /OBJ4/  ECON(I)   ... ECONOMIC DATA FOR OPTIMIZATION
C                           5 - OPERATING DAYS
C                           6 - LANG FACTOR
C                           7 - DEPRECIATION + RATE OF RETURN*(1+IW)/(1-TAX)
C                               IW - RATIO OF WORKING CAPITAL TO INVESTMENT
C                           8 - INCOME TAX
C                           9 - LABOR AND SUPERVISION
C                           10- RATE OF EARNINGS
```

LISTING VIII.5 - CONTINUED

```
C      /OBJ5/  JEQ         ... NO. OF EQUIPMENTS POSSIBLE TO BE EXPANDED
C              IEX(I)      ... EQUIPMENT NUMBER POSSIBLE TO BE EXPANDED
C      /OBJ6/  IW          ... NO. OF UNITS WHOSE CAPACITY IS NOT REPRESENTED
C                              IN TERMS OF FLOW RATE
C              ICAP(I)     ... UNIT NUMBER
C      /LP1/   IROW(I)     ... ROW NUMBER FOR COEFFICIENT OF LP MODEL
C      /LP2/   ICOL(I)     ... COLUMN NUMBER FOR COEFFICIENT OF LP MODEL
C      /LP3/   RDATA(I)    ... COEFFICIENT OF LP MODEL
C      /LP4/   NEQ(I)      ... EQUATION NUMBER OF LP MODEL
C      /LP5/   CNST(I)     ... CONSTRAINTS ON FLOW RATE OR CAPACITY TO BE EXPANDED
C              ICON(I,J)   ... INTEGER INFORMATION
C      /GENRL/ NCP         ... NO. OF COMPONENTS IF PROCESS SYSTEM IS INCLUDED
C              M           ... NO. OF EQUATIONS IN ENERGY BALANCES
C              N           ... TOTAL NO. OF COEFFICIENT IN MODELS
C              IEQ         ... EQUIPMENT NO. OF MODULES CALCULATED
C              MAXST       ... MAXIMUM STREAM NO. DEALT WITH IN THE SYSTEM
C              IUNIT       ... CODE FOR UNIT CONVERSION , 1 - C.G.S 2 - BTU-LB
C              M1          ... TOTAL NO. OF EQUATIONS LINKING BETWEEN DEMAND
C                              AND SUPPLY
C              M2          ... TOTAL NO. OF EQUATIONS RELATING TO OBJECTIVE FUNCTION
C              MEQ         ... SEQUENCE NO. OF EQUATIONS
C              ISEQ        ... MAXIMUM NO. OF EQUIPMENT DEALT WITH IN ENERGY SYSTEMS
C              K           ... WHETHER OR NOT PUNCHED CARDS OF LP MODELS TO MPSX
C                              SHOULD BE GENERATED
C              KD          ... NUMBER OF POWER DEMANDS FOR DRIVER SELECTION
C              IPRIOD      ... PLANT OPERATION PERIODS CONSIDERED
C
```

LISTING VIII.5 - CONTINUED

```
      COMMON /SYS1/SNEN(999),IS(99,2),IE(50,15)/SYS2/LENGS(16)
      COMMON /SYS3/NM,KYWRD(30)/SYS4/NJ(100)/SYS5/NS,ISP(10)
      COMMON /OBJ1/JIN,IIN(10)/OBJ2/JOUT,IOUT(10)/OBJ3/COST(16)
      COMMON /OBJ4/ECON(20)/OBJ5/JEQ,IEX(150)/OBJ6/IW,ICAP(10)
      COMMON /LP1/IROW(1500)/LP2/ICOL(1500)/LP3/RDATA(1500)/LP4/NEQ(600)
      COMMON /LP5/CNST(20),ICON(20,3)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ,K   ,KD,IPRIOD
C     REGISTERED KEYWORD
      DATA  KYWRD/  4HJUNC,6HERECIV, 5HDEAER,  4HPUMP,  4HFURN, 5HSPLTR
     1          , 5HJUNC1,   3HWHR, 5HFDRUM, 5HTURBG, 5HTURBB, 5HVALVE
     1          , 5HTURBC,6HTURBC1,6HHEATEX, 5HSCOND, 5HBOILR,6HETURB1
     1          ,6HETURB2, 5HDESUP,6HGTURBB,6HGTURB1,6HGTURB2,6HGTURBC
     1          ,6HGTURC1,6HSELECT, 5HEQUAL,6HWTREAT, 5HMOTOR, 5HHEATR/
C     GENERAL DATA
      DATA NST,NEN,NIS,NIE,NM/99,50,2,15,30/
      DATA LENGS/3*14,4*4,1,1,7*4/
      DATA COST/3*0.,38.,28.35,0.005 ,0.010 ,0.013,0.013,0.1,3*0.,3*2.5/
      DATA ECON(5),ECON(6),ECON(7),ECON(8),ECON(9)/8.76,2.,0.545,.5,70./
      DATA ECON(10)/0.10 /
      DATA K/7/
C
      CALL      MAIN(    IS,IE,NST,NEN,    NIS,NIE,ICON,NC)
      STOP
      END
```

```
 MAXST    MAXEQ     IUNIT    IPRIOD
    24       10         2         3
STREAM CODE
 10  12  10  10   8  10  13  10  10   4   8  14  13  12   9   9
  9   9  13  13  12  12  12  -0
SYSTEM NETWORK
DEAER     1   3   3   1   1   2   3   4  -0  -0  -0  -0  -0  -0
PUMP      2   3   2   1   4   5   6  -0  -0  -0  -0  -0  -0  -0
HEATR     3   4   2   2   7   6   3   8  -0  -0  -0  -0  -0  -0
BOILR     4   6   3   2   8  11  10  12   9  -0  -0  -0  -0  -0
GTURBI    5   4   1   3  12  13  14  15  -0  -0  -0  -0  -0  -0
ERECIV    6   3   2   2  15  16  17  18  -0  -0  -0  -0  -0  -0
MOTOR     7   3   1   1  18  11  -0  -0  -0  -0  -0  -0  -0  -0
SPLTR     8   1   1   4  13   7  20  19  24  -0  -0  -0  -0  -0
TURBB     9   3   1   2  20   5  21  -0  -0  -0  -0  -0  -0  -0
JUNC     10   1   2   3  14  21   2  23  22  -0  -0  -0  -0  -0
STREAM DATA   11 PERIOD-  1
STREAM    1  2    122.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM    3  2    365.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM    4  3     29.700   2   250.000  -0   -0.000  -0      -0.000
STREAM    6  3    500.000   2   250.000  -0   -0.000  -0      -0.000
STREAM    8  3    474.700   2   350.000  -0   -0.000  -0      -0.000
STREAM   12  3    474.700   2   650.000  -0   -0.000  -0      -0.000
STREAM   13  3    179.700   2   520.000  -0   -0.000  -0      -0.000
STREAM   14  3     29.700   2   300.000  -0   -0.000  -0      -0.000
STREAM   17  1  35100.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM   19  1   1284.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM   22  1     91.000  -0    -0.000  -0   -0.000  -0      -0.000
EQUIPMENT DATA   9
DEAER     1  2   1600.000   3     4.000   1    0.030  -0      -0.000
PUMP      2  2   1600.000   3     6.000   1    0.006  -0      -0.000
HEATR     3  2   3000.000   3    -0.000   4  500.000   1       0.050
BOILR     4  2   1900.000   3    12.000   4    0.040   5   19000.000
BOILR     4  6      1.471   1     6.330  -0   -0.000  -0      -0.000
GRURBI    5  2  33000.000   3    15.000   1    0.100  -0      -0.000
ERECIV    6  2  40000.000   3    -0.000   1    0.200  -0      -0.000
MOTOR     7  2   3320.000   3    11.000   1    0.041  -0      -0.000
TURBB     9  2   1640.000   3     5.000   1    0.037  -0      -0.000
STREAM DATA    3 PERIOD-  2
STREAM   17  1  35100.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM   19  1   1430.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM   22  1    119.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM DATA    3 PERIOD-  3
STREAM   17  1  41700.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM   19  1   1563.000  -0    -0.000  -0   -0.000  -0      -0.000
STREAM   22  1    178.000  -0    -0.000  -0   -0.000  -0      -0.000
```

LISTING VIII.6 AN EXAMPLE OF INPUT DATA TO "OPES"

LISTING VIII.7 LP MODELS FOR AN EXPANSION PROBLEM

NUMBER OF EQUATION = 144

NUMBER OF COEFF. = 456

			E3020014	Y3010202	1.000000
			E3020015	Z4000102	-1.000000
			E3020015	Y3010302	1.000000
			E3020016	Z4000202	-1.000000
E5000100	X7000117	35100.000000	E3020016	Y3020202	1.000000
E5000100	X7000119	1284.000000	E3020017	Z4000202	-1.000000
E5000100	X7000122	91.000000	E3020017	Y3020302	1.000000
E2010001	Y3000101	1.000000	E3020018	Z4000302	-1.000000
E2010001	Y3010101	1.000000	E3020018	Y3030302	1.000000
E2010001	X1000104	-1.000000	E2030019	Y3000103	1.000000
E2010002	Y3000201	1.000000	E2030019	Y3010103	1.000000
E2010002	Y3010201	1.000000	E2030019	W2000103	-1.000000
E2010002	Y3020201	1.000000	E2030020	Y3000203	1.000000
E2010002	X1000204	-1.000000	E2030020	Y3010203	1.000000
E2010003	Y3000301	1.000000	E2030020	Y3020203	1.000000
E2010003	Y3010301	1.000000	E2030020	W2000203	-1.000000
E2010003	Y3020301	1.000000	E2030021	Y3000303	1.000000
E2010003	Y3030301	1.000000	E2030021	Y3010303	1.000000
E2010003	X1000304	-1.000000	E2030021	Y3020303	1.000000
E6000000	Y9000101	1600.000000	E2030021	Y3030303	1.000000
E6000000	Y9000201	1600.000000	E2030021	W2000303	-1.000000
E6000000	Y9000301	1600.000000	E6000000	Y9000103	3000.000000
E3010004	Z4000101	-1.000000	E6000000	Y9000203	3000.000000
E3010004	Y3010101	1.000000	E6000000	Y9000303	3000.000000
E3010005	Z4000101	-1.000000	E3030022	Z4000103	-1.000000
E3010005	Y3010201	1.000000	E3030022	Y3010103	1.000000
E3010006	Z4000101	-1.000000	E3030023	Z4000103	-1.000000
E3010006	Y3010301	1.000000	E3030023	Y3010203	1.000000
E3010007	Z4000201	-1.000000	E3030024	Z4000103	-1.000000
E3010007	Y3020201	1.000000	E3030024	Y3010303	1.000000
E3010008	Z4000201	-1.000000	E3030025	Z4000203	-1.000000
E3010008	Y3020301	1.000000	E3030025	Y3020203	1.000000
E3010009	Z4000301	-1.000000	E3030026	Z4000203	-1.000000
E3010009	Y3030301	1.000000	E3030026	Y3020303	1.000000
E2020010	Y3000102	1.000000	E3030027	Z4000303	-1.000000
E2020010	Y3010102	1.000000	E3030027	Y3030303	1.000000
E2020010	X1000106	-1.000000	E2040028	Y3000104	1.000000
E2020011	Y3000202	1.000000	E2040028	Y3010104	1.000000
E2020011	Y3010202	1.000000	E2040028	X1000112	-1.000000
E2020011	Y3020202	1.000000	E2040029	Y3000204	1.000000
E2020011	X1000206	-1.000000	E2040029	Y3010204	1.000000
E2020012	Y3000302	1.000000	E2040029	Y3020204	1.000000
E2020012	Y3010302	1.000000	E2040029	X1000212	-1.000000
E2020012	Y3020302	1.000000	E2040030	Y3000304	1.000000
E2020012	Y3030302	1.000000	E2040030	Y3010304	1.000000
E2020012	X1000306	-1.000000	E2040030	Y3020304	1.000000
E6000000	Y9000102	1600.000000	E2040030	Y3030304	1.000000
E6000000	Y9000202	1600.000000	E2040030	X1000312	-1.000000
E6000000	Y9000302	1600.000000	E6000000	Y9000104	1900.000000
E3020013	Z4000102	-1.000000	E6000000	Y9000204	1900.000000
E3020013	Y3010102	1.000000	E6000000	Y9000304	1900.000000
E3020014	Z4000102	-1.000000	E3040031	Z4000104	-1.000000

LISTING VIII.7 - CONTINUED

E3040031	Y3010104	1.000000	E3060049	Y3010106	1.000000
E3040032	Z4000104	-1.000000	E3060050	Z4000106	-1.000000
E3040032	Y3010204	1.000000	E3060050	Y3010206	1.000000
E3040033	Z4000104	-1.000000	E3060051	Z4000106	-1.000000
E3040033	Y3010304	1.000000	E3060051	Y3010306	1.000000
E3040034	Z4000204	-1.000000	E3060052	Z4000206	-1.000000
E3040034	Y3020204	1.000000	E3060052	Y3020206	1.000000
E3040035	Z4000204	-1.000000	E3060053	Z4000206	-1.000000
E3040035	Y3020304	1.000000	E3060053	Y3020306	1.000000
E3040036	Z4000304	-1.000000	E3060054	Z4000306	-1.000000
E3040036	Y3030304	1.000000	E3060054	Y3030306	1.000000
E2050037	Y3000105	1.000000	E2070055	Y3000107	1.000000
E2050037	Y3010105	1.000000	E2070055	Y3010107	1.000000
E2050037	X1000115	-1.000000	E2070055	X1000111	-1.000000
E2050038	Y3000205	1.000000	E2070056	Y3000207	1.000000
E2050038	Y3010205	1.000000	E2070056	Y3010207	1.000000
E2050038	Y3020205	1.000000	E2070056	Y3020207	1.000000
E2050038	X1000215	-1.000000	E2070056	X1000211	-1.000000
E2050039	Y3000305	1.000000	E2070057	Y3000307	1.000000
E2050039	Y3010305	1.000000	E2070057	Y3010307	1.000000
E2050039	Y3020305	1.000000	E2070057	Y3020307	1.000000
E2050039	Y3030305	1.000000	E2070057	Y3030307	1.000000
E2050039	X1000315	-1.000000	E2070057	X1000311	-1.000000
E6000000	Y9000105	33000.000000	E6000000	Y9000107	3320.000000
E6000000	Y9000205	33000.000000	E6000000	Y9000207	3320.000000
E6000000	Y9000305	33000.000000	E6000000	Y9000307	3320.000000
E3050040	Z4000105	-1.000000	E3070058	Z4000107	-1.000000
E3050040	Y3010105	1.000000	E3070058	Y3010107	1.000000
E3050041	Z4000105	-1.000000	E3070059	Z4000107	-1.000000
E3050041	Y3010205	1.000000	E3070059	Y3010207	1.000000
E3050042	Z4000105	-1.000000	E3070060	Z4000107	-1.000000
E3050042	Y3010305	1.000000	E3070060	Y3010307	1.000000
E3050043	Z4000205	-1.000000	E3070061	Z4000207	-1.000000
E3050043	Y3020205	1.000000	E3070061	Y3020207	1.000000
E3050044	Z4000205	-1.000000	E3070062	Z4000207	-1.000000
E3050044	Y3020305	1.000000	E3070062	Y3020307	1.000000
E3050045	Z4000305	-1.000000	E3070063	Z4000307	-1.000000
E3050045	Y3030305	1.000000	E3070063	Y3030307	1.000000
E2060046	Y3000106	1.000000	E2090064	Y3000109	1.000000
E2060046	Y3010106	1.000000	E2090064	Y3010109	1.000000
E2060046	W2000106	-1.000000	E2090064	X1000105	-1.000000
E2060047	Y3000206	1.000000	E2090065	Y3000209	1.000000
E2060047	Y3010206	1.000000	E2090065	Y3010209	1.000000
E2060047	Y3020206	1.000000	E2090065	Y3020209	1.000000
E2060047	W2000206	-1.000000	E2090065	X1000205	-1.000000
E2060048	Y3000306	1.000000	E2090066	Y3000309	1.000000
E2060048	Y3010306	1.000000	E2090066	Y3010309	1.000000
E2060048	Y3020306	1.000000	E2090066	Y3020309	1.000000
E2060048	Y3030306	1.000000	E2090066	Y3030309	1.000000
E2060048	W2000306	-1.000000	E2090066	X1000305	-1.000000
E6000000	Y9000106	40000.000000	E6000000	Y9000109	1640.000000
E6000000	Y9000206	40000.000000	E6000000	Y9000209	1640.000000
E6000000	Y9000306	40000.000000	E6000000	Y9000309	1640.000000
E3060049	Z4000106	-1.000000	E3090067	Z4000109	-1.000000

LISTING VIII.7 - CONTINUED

E3090067	Y3010109	1.000000
E3090068	Z4000109	-1.000000
E3090068	Y3010209	1.000000
E3090069	Z4000109	-1.000000
E3090069	Y3010309	1.000000
E3090070	Z4000209	-1.000000
E3090070	Y3020209	1.000000
E3090071	Z4000209	-1.000000
E3090071	Y3020309	1.000000
E3090072	Z4000309	-1.000000
E3090072	Y3030309	1.000000
E1010101	X1000101	1.000000
E1010101	X1000102	1.000000
E1010101	X1000103	1.000000
E1010101	X1000104	-1.000000
E1010102	X1000101	90.139871
E1010102	X1000102	1190.766027
E1010102	X1000103	337.251137
E1010102	X1000104	-218.999102
E1020103	X1000104	1.000000
E1020103	X1000106	-1.000000
E1020104	X1000105	1.000000
E1020104	X1000104	-.623531
E1030105	X1000106	1.000000
E1030105	X1000108	-1.000000
E1030106	X1000108	1.000000
E1030106	W2000103	-.703565
E1030107	X1000107	1.000000
E1030107	X1000103	-1.000000
E1030108	X1000108	321.522206
E1030108	X1000106	-218.999102
E1030108	X1000107	-1285.706599
E1030108	X1000103	337.251137
E1040109	X1000112	1.000000
E1040109	X1000108	-.960000
E1040110	X1000109	1.000000
E1040110	X1000108	-.040000
E1040111	X1000112	-1333.969169
E1040111	X1000109	-442.490886
E1040111	X1000108	321.522206
E1040111	X1000110	17100.000000
E1040112	X1000111	1.000000
E1040112	X1000108	-1.471000
E1050113	X1000113	1.000000
E1050113	X1000114	1.000000
E1050113	X1000112	-1.000000
E1050114	X1000112	21.209075
E1050114	X1000114	32.443611
E1050114	X1000115	-1.002989
E1060115	X1000115	1.000000
E1060115	X1000116	1.000000
E1060115	W2000106	-1.000000
E1060116	X1000117	1.000000
E1060116	X1000118	1.000000
E1060116	W2000106	-1.000000
E1070117	X1000111	1.000000
E1070117	X1000118	-.950000
E1080118	X1000107	1.000000
E1080118	X1000120	1.000000
E1080118	X1000119	1.000000
E1080118	X1000124	1.000000
E1080118	X1000113	-1.000000
E1090119	X1000120	1.000000
E1090119	X1000121	-1.000000
E1090120	X1000105	1.000000
E1090120	X1000120	-25.582776
E1100121	X1000114	1.000000
E1100121	X1000121	1.000000
E1100121	X1000102	-1.000000
E1100121	X1000123	-1.000000
E1100121	X1000122	-1.000000
E5000200	X7000217	35100.000000
E5000200	X7000219	1430.000000
E5000200	X7000222	119.000000
E1010201	X1000201	1.000000
E1010201	X1000202	1.000000
E1010201	X1000203	1.000000
E1010201	X1000204	-1.000000
E1010202	X1000201	90.139871
E1010202	X1000202	1190.766027
E1010202	X1000203	337.251137
E1010202	X1000204	-218.999102
E1020203	X1000204	1.000000
E1020203	X1000206	-1.000000
E1020204	X1000205	1.000000
E1020204	X1000204	-.623531
E1030205	X1000206	1.000000
E1030205	X1000208	-1.000000
E1030206	X1000208	1.000000
E1030206	W2000203	-.703565
E1030207	X1000207	1.000000
E1030207	X1000203	-1.000000
E1030208	X1000208	321.522206
E1030208	X1000206	-218.999102
E1030208	X1000207	-1285.706599
E1030208	X1000203	337.251137
E1040209	X1000212	1.000000
E1040209	X1000208	-.960000
E1040210	X1000209	1.000000
E1040210	X1000208	-.040000
E1040211	X1000212	-1333.969169
E1040211	X1000209	-442.490886
E1040211	X1000208	321.522206
E1040211	X1000210	17100.000000
E1040212	X1000211	1.000000
E1040212	X1000208	-1.471000
E1050213	X1000213	1.000000
E1050213	X1000214	1.000000

LISTING VIII.7 - CONTINUED

E1050213	X1000212	-1.000000	E1040310	X1000309	1.000000
E1050214	X1000212	21.209075	E1040310	X1000308	-.040000
E1050214	X1000214	32.443611	E1040311	X1000312	-1333.969169
E1050214	X1000215	-1.002989	E1040311	X1000309	-442.490886
E1060215	X1000215	1.000000	E1040311	X1000308	321.522206
E1060215	X1000216	1.000000	E1040311	X1000310	17100.000000
E1060215	W2000206	-1.000000	E1040312	X1000311	1.000000
E1060216	X1000217	1.000000	E1040312	X1000308	-1.471000
E1060216	X1000218	1.000000	E1050313	X1000313	1.000000
E1060216	W2000206	-1.000000	E1050313	X1000314	1.000000
E1070217	X1000211	1.000000	E1050313	X1000312	-1.000000
E1070217	X1000218	-.950000	E1050314	X1000312	21.209075
E1080218	X1000207	1.000000	E1050314	X1000314	32.443611
E1080218	X1000220	1.000000	E1050314	X1000315	-1.002989
E1080218	X1000219	1.000000	E1060315	X1000315	1.000000
E1080218	X1000224	1.000000	E1060315	X1000316	1.000000
E1080218	X1000213	-1.000000	E1060315	W2000306	-1.000000
E1090219	X1000220	1.000000	E1060316	X1000317	1.000000
E1090219	X1000221	-1.000000	E1060316	X1000318	1.000000
E1090220	X1000205	1.000000	E1060316	W2000306	-1.000000
E1090220	X1000220	-25.582776	E1070317	X1000311	1.000000
E1100221	X1000214	1.000000	E1070317	X1000318	-.950000
E1100221	X1000221	1.000000	E1080318	X1000307	1.000000
E1100221	X1000202	-1.000000	E1080318	X1000320	1.000000
E1100221	X1000223	-1.000000	E1080318	X1000319	1.000000
E1100221	X1000222	-1.000000	E1080318	X1000324	1.000000
E5000300	X7000317	41700.000000	E1080318	X1000313	-1.000000
E5000300	X7000319	1563.000000	E1090319	X1000320	1.000000
E5000300	X7000322	178.000000	E1090319	X1000321	-1.000000
E1010301	X1000301	1.000000	E1090320	X1000305	1.000000
E1010301	X1000302	1.000000	E1090320	X1000320	-25.582776
E1010301	X1000303	1.000000	E1100321	X1000314	1.000000
E1010301	X1000304	-1.000000	E1100321	X1000321	1.000000
E1010302	X1000301	90.139871	E1100321	X1000302	-1.000000
E1010302	X1000302	1190.766027	E1100321	X1000323	-1.000000
E1010302	X1000303	337.251137	E1100321	X1000322	-1.000000
E1010302	X1000304	-218.999102	E7000000	C5000101	1.000000
E1020303	X1000304	1.000000	E7000000	C5000102	1.000000
E1020303	X1000306	-1.000000	E7000000	C5000201	.975610
E1020304	X1000305	1.000000	E7000000	C5000202	.975610
E1020304	X1000304	-.623531	E7000000	C5000301	.951814
E1030305	X1000306	1.000000	E7000000	C5000302	.951814
E1030305	X1000308	-1.000000	E4000101	Z4000101	.030000
E1030306	X1000308	1.000000	E4000101	Z4000102	.006000
E1030306	W2000303	-.703565	E4000101	Z4000103	.050000
E1030307	X1000307	1.000000	E4000101	Z4000104	6.330000
E1030307	X1000303	-1.000000	E4000101	Z4000105	.100000
E1030308	X1000308	321.522206	E4000101	Z4000106	.200000
E1030308	X1000306	-218.999102	E4000101	Z4000107	.040500
E1030308	X1000307	-1285.706599	E4000101	Z4000109	.037300
E1030308	X1000303	337.251137	E4000101	C5000103	-1.000000
E1040309	X1000312	1.000000	E4000102	C5000103	.200000
E1040309	X1000308	-.960000	E4000102	C5000101	-1.000000

LISTING VIII.7 - CONTINUED

E4000103	X1000101	.219000
E4000103	X1000110	39.420000
E4000103	X1000116	.028470
E4000103	C5000102	-1.000000
E4000204	Z4000201	.030000
E4000204	Z4000202	.006000
E4000204	Z4000203	.050000
E4000204	Z4000204	6.330000
E4000204	Z4000205	.100000
E4000204	Z4000206	.200000
E4000204	Z4000207	.040500
E4000204	Z4000209	.037300
E4000204	C5000203	-1.000000
E4000205	C5000203	.200000
E4000205	C5000201	-1.000000
E4000206	X1000201	.219000
E4000206	X1000210	39.420000
E4000206	X1000216	.028470
E4000206	C5000202	-1.000000
E4000307	Z4000301	.030000
E4000307	Z4000302	.006000
E4000307	Z4000303	.050000
E4000307	Z4000304	6.330000
E4000307	Z4000305	.100000
E4000307	Z4000306	.200000
E4000307	Z4000307	.040500
E4000307	Z4000309	.037300
E4000307	C5000303	-1.000000
E4000308	C5000303	.200000
E4000308	C5000301	-1.000000
E4000309	X1000301	.219000
E4000309	X1000310	39.420000
E4000309	X1000316	.028470
E4000309	C5000302	-1.000000

Listing VIII.8 A typical Set of Control Cards for MPSX Run

```
//UWO3574 JOB (TY,5273,xxx,500,80),'NISHIO',
                         CLASS=C,REGION=(200K)
/*PRIORITY 8
//EXEC MPSX
//MPS.SYSIN DD
PROGRAM
INITIALZ
MOVE(XPBNAME. 'PBFILE')
CONVERT ('SUMMARY')
SETUP ('BOUND','DEMAND')
MOVE(XOBJ,'E7000000')
MOVE(XRHS,'R6')
PRIMAL
SOLUTION
EXIT
PEND
/*
//GO. SYSIN DD *
DATA GENERATED FROM "OSES" OR "OPES"
//
/$
/*EOF
```

An input example for a synthesis problem is for the application problem that was taken up in Chapter 7 and shown in Listing VIII.9. The program system "OSES" generates LP input data to MPSX which are given in Listing VIII.10, where only data for the first step in two-level coordination necessary for the example taken up are listed. The control cards for MPSX run are exactly the same as those for an expansion problem.

LISTING VIII.9 INPUT DATA TO "OSES"

```
** OPTIMAL SYNTHESIS PROBLEM FOR STEAM AND POWER PLANT **

     MAXST      MAXEQ      IUNIT      IPRIOD     SIMCODE
        99         41          2          1          1

STREAM CODE
     6  12  10   9  10  12  10  10  10  13  10  10  13  10  10   7
     4  15  10  13  10  12  10  14  14  14  14  14  14  14  14  14
    14  14  14  14  14  13  13  12   9  14   9  12   9   6  13  10
     9   7  12  10   9  13  12   9  13   9  12   9   7  13  10   9
     7  12  10   9   8   8   8   7   9  13  13  13  13  13  13  13
    13  12   9   7  10   9  12   8   7  10   8  12  12  12   5   5
     4   8   8

SYSTEM NETWORK
WTREAI       1   4   2   1   1   2   3   0   0   0   0   0   0   0
PUMP         2   3   2   1   3   4   5   0   0   0   0   0   0   0
DEAER        3   3   3   1   5   6   7   8   0   0   0   0   0   0
PUMP         4   3   2   1   8   4   9   0   0   0   0   0   0   0
JUNC1        5   1   9   1  12  15  81  67  63  52  48  90  85   7
HEATR        6   4   2   2  10   9  12  11   0   0   0   0   0   0
HEATR        7   4   2   2  13  11  15  14   0   0   0   0   0   0
BOILR        8   6   3   2  14   4  17  18  19   0   0   0   0   0
FDRUM        9   4   1   2  19  20  21   0   0   0   0   0   0   0
FDRUM       10   4   1   2  21  22  23   0   0   0   0   0   0   0
EQUAL       11   1   1   1  18  24   0   0   0   0   0   0   0   0
SPLTR       12   1   1   8  24  25  26  27  28  29  30  31  32   0
SPLTR       13   1   1   5  32  33  34  35  36  37   0   0   0   0
EQUAL       14   1   1   1  26  38   0   0   0   0   0   0   0   0
GTURB1      15   4   1   3  27  39  40  41   0   0   0   0   0   0
GTURBB      16   4   1   2  28  42  43   0   0   0   0   0   0   0
```

LISTING VIII.9 - CONTINUED

GTURBB	17	4	1	2	29	44	45	0	0	0	0	0	0	0
GTURC1	18	7	2	3	30	46	47	48	49	0	0	0	0	0
GTURC1	19	7	2	3	31	50	51	52	53	0	0	0	0	0
ETURB1	20	4	1	3	33	54	55	56	0	0	0	0	0	0
TURBB	21	3	1	2	34	57	58	0	0	0	0	0	0	0
TURBB	22	3	1	2	35	59	60	0	0	0	0	0	0	0
TURBC1	23	7	2	3	36	61	62	63	64	0	0	0	0	0
TURBC1	24	7	2	3	37	65	66	67	68	0	0	0	0	0
ERECIV	25	3	9	1	69	41	43	45	49	53	88	91	98	70
PUMP	26	3	2	1	72	4	16	0	0	0	0	0	0	0
JUNC	27	1	8	1	38	39	42	20	47	54	57	62	79	0
SPLTR	28	1	1	8	79	74	75	76	77	78	80	81	13	0
MOTOR	29	3	1	1	99	73	0	0	0	0	0	0	0	0
TURBB	30	3	1	2	75	82	83	0	0	0	0	0	0	0
TURBC	31	4	2	2	76	84	85	86	0	0	0	0	0	0
GTURBB	32	4	1	2	77	87	88	0	0	0	0	0	0	0
GTURBC	33	7	2	2	78	89	90	91	0	0	0	0	0	0
EQUAL	34	1	1	1	80	92	0	0	0	0	0	0	0	0
JUNC	36	1	5	1	40	82	44	87	92	93	0	0	0	0
JUNC	37	1	6	4	93	51	55	59	22	66	94	2	6	10
SPLTR	38	1	1	6	16	65	61	50	89	46	84	0	0	0
JUNC	39	1	2	1	97	96	17	0	0	0	0	0	0	0
TURBG	40	4	1	2	95	96	98	0	0	0	0	0	0	0
SPLTR	41	1	1	2	70	71	99	0	0	0	0	0	0	0
SELEC1	35	30	8	1	73	83	86	56	58	60	64	68	4	0

LISTING VIII.9 - CONTINUED

STREAM DATA		22	PERIOD- 1						
STREAM	9	2	250.000	3	600.000	0	0.000	0	0.000
STREAM	18	2	750.000	3	574.700	0	0.000	0	0.000
STREAM	24	2	750.000	3	534.700	0	0.000	0	0.000
STREAM	3	3	14.700	0	0.000	0	0.000	0	0.000
STREAM	5	2	170.000	3	49.700	0	0.000	0	0.000
STREAM	7	2	300.000	0	0.000	0	0.000	0	0.000
STREAM	8	2	250.000	3	29.700	0	0.000	0	0.000
STREAM	11	2	290.000	0	0.000	0	0.000	0	0.000
STREAM	12	2	286.000	0	0.000	0	0.000	0	0.000
STREAM	14	2	350.000	0	0.000	0	0.000	0	0.000
STREAM	15	2	365.000	0	0.000	0	0.000	0	0.000
STREAM	16	2	64.000	3	40.000	0	0.000	0	0.000
STREAM	25	1	75.600	0	0.000	0	0.000	0	0.000
STREAM	26	1	410.000	0	0.000	0	0.000	0	0.000
STREAM	38	2	500.000	3	179.700	0	0.000	0	0.000
STREAM	40	2	300.000	3	29.700	0	0.000	0	0.000
STREAM	71	1	26140.000	0	0.000	0	0.000	0	0.000
STREAM	72	3	14.700	0	0.000	0	0.000	0	0.000
STREAM	74	1	1100.000	0	0.000	0	0.000	0	0.000
STREAM	80	1	10.000	0	0.000	0	0.000	0	0.000
STREAM	81	1	26.300	0	0.000	0	0.000	0	0.000
STREAM	94	1	139.300	0	0.000	0	0.000	0	0.000
EQUIPMENT DATA		37							
WTREAT	1	1	.050	2	0.000	3	1.000	4	.080
PUMP	2	1	.006	2	0.000	3	5.000	0	0.000
DEAER	3	1	.030	2	0.000	3	8.000	0	0.000
PUMPR	4	1	.006	2	0.000	3	9.000	0	0.000
HEATR	6	1	.050	2	0.000	3	0.000	4	500.000
HEATR	7	1	.050	2	0.000	3	0.000	4	500.000

LISTING VIII.9 - CONTINUED

BOILR	8	1	6.330	2	0.000	3	18.000	4	.050
BOILR	8	5	18900.400	6	1.650	0	0.000	0	0.000
FDRUM	9	1	.050	2	0.000	3	19.000	4	.950
FDRUM	10	1	.050	2	0.000	3	21.000	4	.950
GTURB1	15	1	.100	2	0.000	3	41.000	4	.020
GTURBB	16	1	.090	2	0.000	3	43.000	4	.020
GTURBB	17	1	.090	2	0.000	3	45.000	4	.020
GTURC1	18	1	.150	2	0.000	3	49.000	4	.020
GTURC1	18	5	28.500	6	122.000	7	122.000	0	0.000
GTURC1	19	1	.150	2	0.000	3	53.000	4	.020
GTURC1	19	5	28.500	6	122.000	7	122.000	0	0.000
ETURB1	20	1	.040	2	0.000	3	56.000	0	0.000
TURBB1	21	1	.035	2	0.000	3	58.000	0	0.000
TURBB1	22	1	.035	2	0.000	3	60.000	0	0.000
TURBC1	23	1	.050	2	0.000	3	64.000	0	0.000
TURBC1	23	5	28.500	6	122.000	7	122.000	0	0.000
TURBC1	24	1	.050	2	0.000	3	68.000	0	0.000
TURBC1	24	5	28.500	6	122.000	7	122.000	0	0.000
ERECIV	25	1	.200	2	0.000	3	0.000	0	0.000
MOTORV	29	1	.041	2	0.000	3	73.000	0	0.000
TURBB	30	1	.035	2	0.000	3	83.000	0	0.000
TURBC	31	1	.050	2	0.000	3	86.000	0	0.000
TURBC	31	5	28.500	6	122.000	7	122.000	0	0.000
GTURBB	32	1	.090	2	0.000	3	88.000	4	.020
GTURBC	33	1	.100	2	0.000	3	91.000	4	.020
GTURBC	33	5	28.500	6	122.000	7	122.000	0	0.000
PUMP	26	1	.006	2	0.000	3	16.000	0	0.000
TURBG	40	1	.150	2	0.000	3	98.000	0	0.000
SELECT	35	1	2.000	2	4.000	3	8.000	4	26.000
SELECT	35	5	184.000	6	1288.000	7	257.500	8	225.000
SELECT	35	9	456.000	10	368.000	0	0.000	0	0.000

LISTING VIII.10 INPUT DATA TO MPSX FOR A SYNTHESIS PROBLEM

```
NAME          ENERGY

ROWS
 N  E7000000
 E  E1010101
 E  E1010102
 E  E1020103
 E  E1020104
 E  E1030105
 E  E1030106
 E  E1040107
 E  E1040108
 E  E1050109
 E  E1060110
 E  E1060111
 E  E1060112
 E  E1060113
 E  E1070114
 E  E1070115
 E  E1070116
 E  E1070117
 E  E1080118
 E  E1080119
 E  E1080120
 E  E1080121
 E  E1090122
 E  E1090123
 E  E1100124
 E  E1100125
 E  E1110126
 E  E1120127
 E  E1130128
 E  E1140129
 E  E1150130
 E  E1150131
 E  E1160132
 E  E1160133
 E  E1170134
 E  E1170135
 E  E1180136
 E  E1180137
 E  E1180138
 E  E1190139
 E  E1190140
 E  E1190141
 E  E1200142
 E  E1200143
 E  E1210144
 E  E1210145
 E  E1220146
 E  E1220147
 E  E1230148
 E  E1230149
 E  E1230150
 E  E1240151
 E  E1240152
 E  E1240153
 E  E1250154
 E  E1250155
 E  E1260156
 E  E1260157
 E  E1270158
 E  E1280159
 E  E1290160
 E  E1300161
 E  E1300162
 E  E1310163
 E  E1310164
 E  E1310165
 E  E1320166
 E  E1320167
 E  E1330168
 E  E1330169
 E  E1330170
 E  E1340171
 E  E1360172
 E  E1370173
 E  E1380174
 E  E1390175
 E  E1400176
 E  E1400177
 E  E1410178
 E  E1350179
 E  E1350180
 E  E1290181
 E  E1300182
 E  E1300183
 E  E1310184
 E  E1310185
 E  E1310186
 E  E1200187
 E  E1200188
 E  E1210189
 E  E1210190
 E  E1220191
 E  E1220192
 E  E1230193
 E  E1230194
 E  E1230195
 E  E1240196
 E  E1240197
 E  E1240198
 E  E1350199
 E  E1260200
 E  E1300201
 E  E1300202
 E  E1310203
 E  E1310204
 E  E1310205
 E  E1200206
 E  E1200207
 E  E1210208
 E  E1210209
 E  E1220210
 E  E1220211
 E  E1230212
 E  E1230213
 E  E1230214
 E  E1240215
 E  E1240216
 E  E1240217
 E  E1350218
 E  E1290219
 E  E1300220
 E  E1300221
 E  E1310222
 E  E1310223
 E  E1310224
 E  E1200225
 E  E1200226
 E  E1210227
 E  E1210228
 E  E1220229
 E  E1220230
 E  E1230231
```

LISTING VIII.10 - CONTINUED

E E1230232
E E1230233
E E1240234
E E1240235
E E1240236
E E1350237
E E1290238
E E1300239
E E1300240
E E1310241
E E1310242
E E1310243
E E1200244
E E1200245
E E1210246
E E1210247
E E1220248
E E1220249
E E1230250
E E1230251
E E1230252
E E1240253
E E1240254
E E1240255
E E1350256
E E1290257
E E1300258
E E1300259
E E1310260
E E1310261
E E1310262
E E1200263
E E1200264
E E1210265
E E1210266
E E1220267
E E1220268
E E1230269
E E1230270
E E1230271
E E1240272
E E1240273
E E1240274
E E1350275
E E1290276
E E1300277
E E1300278
E E1310279
E E1310280
E E1310281
E E1200282
E E1200283
E E1210284
E E1210285
E E1220286
E E1220287
E E1230288
E E1230289
E E1230290
E E1240291
E E1240292
E E1240293
E E1350294
E E1200295
E E1300296
E E1300297
E E1310298
E E1310299
E E1310300
E E1200301
E E1200302
E E1210303
E E1210304
E E1220305
E E1220306
E E1230307
E E1230308
E E1230309
E E1240310
E E1240311
E E1240312
E E1350313
E E1290314
E E1300315
E E1300316
E E1310317
E E1310318
E E1310319
E E1200320
E E1200321
E E1210322
E E1210323
E E1220324
E E1220325
E E1230326
E E1230327
E E1230328
E E1240329
E E1240330
E E1240331
E E1350332
E E1290333
E E1300334
E E1300335
E E1310336
E E1310337
E E1310338
E E1200339
E E1200340
E E1210341
E E1210342
E E1220343
E E1220344
E E1230345
E E1230346
E E1230347
E E1240348
E E1240349
E E1240350
E E4000101
E E4000102
E E4000103

LISTING VIII.10 - CONTINUED

COLUMNS

X1000101	E1010101	1.000000	X1000116	E1380174	-1.000000
X1000101	E1010102	-.080000	X1000116	E4000101	.006000
X1000101	E4000101	.050000	X1000117	E1080120	17010.000000
X1000101	E4000103	.043800	X1000117	E1390175	-1.000000
X1000102	E1010102	1.000000	X1000118	E1080120	-1387.173465
X1000102	E1370173	-1.000000	X1000118	E1110126	-1.000000
X1000103	E1010101	-1.000000	X1000118	E1080118	1.000000
X1000103	E1020103	1.000000	X1000118	E4000101	6.330000
X1000103	E1020104	-.046404	X1000119	E1080120	-465.551116
X1000104	E4000101	0.000000	X1000119	E1090122	-.950000
X1000105	E1020103	-1.000000	X1000119	E1090123	-.050000
X1000105	E1030105	1.000000	X1000119	E1080119	1.000000
X1000105	E1030106	138.367204	X1000119	E4000101	.050000
X1000105	E4000101	.006000	X1000120	E1090122	1.000000
X1000106	E1030105	1.000000	X1000120	E1270158	1.000000
X1000106	E1030106	1190.766027	X1000121	E1090123	1.000000
X1000106	E1370173	-1.000000	X1000121	E1100124	-.950000
X1000107	E1030105	1.000000	X1000121	E1100125	-.050000
X1000107	E1030106	269.862153	X1000121	E4000101	.050000
X1000107	E1050109	-1.000000	X1000122	E1100124	1.000000
X1000108	E1030105	-1.000000	X1000122	E1370173	1.000000
X1000108	E1030106	-218.999102	X1000123	E1100125	1.000000
X1000108	E1040107	1.000000	X1000124	E1120127	-1.000000
X1000108	E1040108	-.756113	X1000124	E1110126	1.000000
X1000108	E4000101	.030000	X1000125	E1120127	1.000000
X1000109	E1060110	1.000000	X1000126	E1140129	-1.000000
X1000109	E1060113	-218.999102	X1000126	E1120127	1.000000
X1000109	E1040107	-1.000000	X1000127	E1120127	1.000000
X1000109	E4000101	.006000	X1000127	E1150130	-1.000000
X1000110	E1060112	1.000000	X1000127	E1150131	23.026803
X1000110	E1060113	-1190.766027	X1000128	E1160132	-1.000000
X1000110	E1370173	-1.000000	X1000128	E1160133	23.026803
X1000111	E1060110	-1.000000	X1000128	E1120127	1.000000
X1000111	E1060111	1.000000	X1000129	E1120127	1.000000
X1000111	E1060113	259.640208	X1000129	E1170134	-1.000000
X1000111	E1070114	1.000000	X1000129	E1170135	50.795122
X1000111	E1070117	-259.640208	X1000130	E1120127	1.000000
X1000112	E1060112	-1.000000	X1000130	E1180136	-1.000000
X1000112	E1060113	255.559425	X1000130	E1180137	23.026803
X1000112	E1050109	1.000000	X1000131	E1120127	1.000000
X1000113	E1070116	1.000000	X1000131	E1190139	-1.000000
X1000113	E1070117	-1274.387960	X1000131	E1190140	50.795122
X1000113	E1280159	1.000000	X1000132	E1120127	1.000000
X1000114	E1070114	-1.000000	X1000132	E1130128	-1.000000
X1000114	E1070115	1.000000	X1000138	E1140129	1.000000
X1000114	E1070117	321.522206	X1000138	E1270158	1.000000
X1000114	E1080118	-.950000	X1000139	E1150130	1.000000
X1000114	E1080119	-.050000	X1000139	E1270158	1.000000
X1000114	E1080120	321.522206	X1000140	E1150131	28.265869
X1000114	E1080121	-1.650000	X1000140	E1150130	1.000000
X1000115	E1070116	-1.000000	X1000140	E1360172	1.000000
X1000115	E1070117	337.251137	X1000141	E1250154	1.000000
X1000115	E1050109	1.000000	X1000141	E1150131	-1.020408
X1000116	E1260156	-1.000000	X1000141	E4000101	.100000
			X1000142	E1160132	1.000000

LISTING VIII.10 - CONTINUED

X1000142	E1270158	1.000000
X1000143	E1160133	-1.020408
X1000143	E1250154	1.000000
X1000143	E4000101	.090000
X1000144	E1170134	1.000000
X1000144	E1360172	1.000000
X1000145	E1250154	1.000000
X1000145	E1170135	-1.020408
X1000145	E4000101	.090000
X1000146	E1180138	-1.000000
X1000146	E1380174	1.000000
X1000147	E1180136	1.000000
X1000147	E1270158	1.000000
X1000148	E1180136	1.000000
X1000148	E1180138	16.768744
X1000148	E1050109	1.000000
X1000148	E1180137	32.662614
X1000149	E1250154	1.000000
X1000149	E1180137	-1.020408
X1000149	E4000101	.150000
X1000150	E1190141	-1.000000
X1000150	E1380174	1.000000
X1000151	E1370173	1.000000
X1000151	E1190139	1.000000
X1000152	E1190140	5.674947
X1000152	E1190141	17.888390
X1000152	E1050109	1.000000
X1000152	E1190139	1.000000
X1000153	E1250154	1.000000
X1000153	E1190140	-1.020408
X1000153	E4000101	.150000
X1000169	E1250154	1.000000
X1000169	E4000103	.113880
X1000170	E1250155	1.000000
X1000170	E1410178	-1.000000
X1000171	E1410178	1.000000
X1000172	E1260156	1.000000
X1000172	E1260157	-.033543
X1000172	E4000103	.087600
X1000174	E1280159	1.000000
X1000177	E1280159	1.000000
X1000177	E1320166	-1.000000
X1000177	E1320167	28.265869
X1000178	E1280159	1.000000
X1000178	E1330168	-1.000000
X1000178	E1330169	65.192830
X1000179	E1270158	-1.000000
X1000179	E1280159	-1.000000
X1000180	E1280159	1.000000
X1000180	E1340171	-1.000000
X1000181	E1280159	1.000000
X1000181	E1050109	1.000000
X1000187	E1320166	1.000000
X1000187	E1360172	1.000000
X1000188	E1320167	-1.020408
X1000188	E4000101	.090000
X1000188	E1250154	1.000000
X1000189	E1330170	-1.000000
X1000189	E1360174	1.000000
X1000190	E1050109	1.000000
X1000190	E1330168	1.000000
X1000190	E1330170	13.681296
X1000191	E1250154	1.000000
X1000191	E1330169	-1.020408
X1000191	E4000101	.100000
X1000192	E1340171	1.000000
X1000192	E1360172	1.000000
X1000193	E1360172	-1.000000
X1000193	E1370173	1.000000
X1000194	E1370173	-1.000000
X1000195	E1400176	-1.000000
X1000195	E1400177	-1296.540000
X1000195	E4000103	248.346000
X1000196	E1390175	1.000000
X1000196	E1400176	1.000000
X1000197	E1390175	1.000000
X1000197	E4000103	332.880000
X1000198	E1250154	1.000000
X1000198	E1400177	1.000000
X1010104	E1350179	-1.000000
X1010104	E1020104	1.000000
X1010133	E1200142	-1.000000
X1010133	E1200143	18.570003
X1010133	E1130128	1.000000
X1010134	E1210144	1.000000
X1010134	E1130128	1.000000
X1010134	E1210145	-18.570003
X1010135	E1220146	1.000000
X1010135	E1130128	1.000000
X1010135	E1220147	-40.963808
X1010136	E1230148	-1.000000
X1010136	E1130128	1.000000
X1010136	E1230149	18.570003
X1010137	E1130128	1.000000
X1010137	E1240152	40.963808
X1010137	E1240151	-1.000000
X1010154	E1270158	1.000000
X1010154	E1200142	1.000000
X1010155	E1370173	1.000000
X1010155	E1200142	1.000000
X1010155	E1200143	22.795055
X1010156	E1200143	-1.000000
X1010156	E1350179	1.000000
X1010156	E4000101	.040000
X1010157	E1270158	1.000000
X1010157	E1210144	-1.000000
X1010158	E1350179	1.000000
X1010158	E1210145	1.000000
X1010158	E4000101	.035000
X1010159	E1370173	1.000000

LISTING VIII.10 - CONTINUED

X1010159	E1220146	-1.000000	X1020134	E1210189	1.000000
X1010160	E1350179	1.000000	X1020134	E1210190	-18.570003
X1010160	E1220147	1.000000	X1020134	E1130128	1.000000
X1010160	E4000101	.035000	X1020135	E1220191	1.000000
X1010161	E1380174	1.000000	X1020135	E1220192	-40.963808
X1010161	E1230150	-1.000000	X1020135	E1130128	1.000000
X1010162	E1230148	1.000000	X1020136	E1230193	-1.000000
X1010162	E1270158	1.000000	X1020136	E1230194	18.570003
X1010163	E1050109	1.000000	X1020136	E1130128	1.000000
X1010163	E1230150	16.768744	X1020137	E1240196	-1.000000
X1010163	E1230148	1.000000	X1020137	E1240197	40.963808
X1010163	E1230149	26.340818	X1020137	E1130128	1.000000
X1010164	E1230149	-1.000000	X1020154	E1200187	1.000000
X1010164	E1350179	1.000000	X1020154	E1270158	1.000000
X1010164	E4000101	.050000	X1020155	E1200187	1.000000
X1010165	E1380174	1.000000	X1020155	E1200188	22.795055
X1010165	E1240153	-1.000000	X1020155	E1370173	1.000000
X1010166	E1240151	1.000000	X1020156	E1350180	1.000000
X1010166	E1370173	1.000000	X1020156	E1200188	-1.000000
X1010167	E1240153	17.888390	X1020156	E4000101	.040000
X1010167	E1050109	1.000000	X1020157	E1210189	-1.000000
X1010167	E1240152	4.576571	X1020157	E1270158	1.000000
X1010167	E1240151	1.000000	X1020158	E1350180	1.000000
X1010168	E1350179	1.000000	X1020158	E1210190	1.000000
X1010168	E1240152	-1.000000	X1020158	E4000101	.035000
X1010168	E4000101	.050000	X1020159	E1370173	1.000000
X1010173	E1290160	1.000000	X1020159	E1220191	-1.000000
X1010173	E1350179	1.000000	X1020160	E1220192	1.000000
X1010173	E4000101	.041000	X1020160	E1350180	1.000000
X1010175	E1300161	1.000000	X1020160	E4000101	.035000
X1010175	E1300162	-22.795055	X1020161	E1230195	-1.000000
X1010175	E1280159	1.000000	X1020161	E1380174	1.000000
X1010176	E1310163	1.000000	X1020162	E1230193	1.000000
X1010176	E1310164	-72.666306	X1020162	E1270158	1.000000
X1010176	E1280159	1.000000	X1020163	E1230194	26.340818
X1010182	E1300161	-1.000000	X1020163	E1230195	16.768744
X1010182	E1360172	1.000000	X1020163	E1050109	1.000000
X1010183	E1300162	1.000000	X1020163	E1230193	1.000000
X1010183	E1350179	1.000000	X1020164	E1230194	-1.000000
X1010183	E4000101	.035000	X1020164	E1350180	1.000000
X1010184	E1310165	-1.000000	X1020164	E4000101	.050000
X1010184	E1380174	1.000000	X1020165	E1240198	-1.000000
X1010185	E1310163	-1.000000	X1020165	E1380174	1.000000
X1010185	E1050109	1.000000	X1020166	E1370173	1.000000
X1010185	E1310165	32.432176	X1020166	E1240196	1.000000
X1010186	E1310164	1.000000	X1020167	E1240198	17.888390
X1010186	E4000101	.050000	X1020167	E1240196	1.000000
X1010186	E1350179	1.000000	X1020167	E1050109	1.000000
X1010199	E1290160	-.950000	X1020167	E1240197	4.576571
X1010199	E1410178	1.000000	X1020168	E1240197	-1.000000
X1020104	E1350180	-1.000000	X1020168	E1350180	1.000000
X1020104	E1040108	1.000000	X1020168	E4000101	.050000
X1020133	E1200187	-1.000000	X1020173	E1350180	1.000000
X1020133	E1200188	18.570003	X1020173	E1290181	1.000000
X1020133	E1130128	1.000000	X1020173	E4000101	.041000

LISTING VIII.10 - CONTINUED

X1020175	E1280159	1.000000
X1020175	E1300182	1.000000
X1020175	E1300183	-22.795055
X1020176	E1310184	1.000000
X1020176	E1280159	1.000000
X1020176	E1310185	-72.666306
X1020182	E1300182	-1.000000
X1020182	E1360172	1.000000
X1020183	E1350180	1.000000
X1020183	E1300183	1.000000
X1020183	E4000101	.035000
X1020184	E1380174	1.000000
X1020184	E1310186	-1.000000
X1020185	E1310184	-1.000000
X1020185	E1310186	32.432176
X1020185	E1050109	1.000000
X1020186	E1310185	1.000000
X1020186	E1350180	1.000000
X1020186	E4000101	.050000
X1020199	E1290181	-.950000
X1020199	E1410178	1.000000
X1030104	E1350199	-1.000000
X1030104	E1080121	1.000000
X1030133	E1200207	18.570003
X1030133	E1130128	1.000000
X1030133	E1200206	-1.000000
X1030134	E1210208	1.000000
X1030134	E1210209	-18.570003
X1030134	E1130128	1.000000
X1030135	E1220210	1.000000
X1030135	E1220211	-40.963808
X1030135	E1130128	1.000000
X1030136	E1230212	-1.000000
X1030136	E1230213	18.570003
X1030136	E1130128	1.000000
X1030137	E1240215	-1.000000
X1030137	E1240216	40.963808
X1030137	E1130128	1.000000
X1030154	E1270158	1.000000
X1030154	E1200206	1.000000
X1030155	E1200207	22.795055
X1030155	E1370173	1.000000
X1030155	E1200206	1.000000
X1030156	E1350199	1.000000
X1030156	E4000101	.040000
X1030156	E1200207	-1.000000
X1030157	E1210208	-1.000000
X1030157	E1270158	1.000000
X1030158	E1210209	1.000000
X1030158	E1350199	1.000000
X1030158	E4000101	.035000
X1030159	E1220210	-1.000000
X1030159	E1370173	1.000000
X1030160	E1350199	1.000000
X1030160	E1220211	1.000000
X1030160	E4000101	.035000
X1030161	E1230214	-1.000000
X1030161	E1380174	1.000000
X1030162	E1270158	1.000000
X1030162	E1230212	1.000000
X1030163	E1230213	26.340818
X1030163	E1230214	16.768744
X1030163	E1050109	1.000000
X1030163	E1230212	1.000000
X1030164	E1230213	-1.000000
X1030164	E1350199	1.000000
X1030164	E4000101	.050000
X1030165	E1240217	-1.000000
X1030165	E1380174	1.000000
X1030166	E1240215	1.000000
X1030166	E1370173	1.000000
X1030167	E1240215	1.000000
X1030167	E1240216	4.576571
X1030167	E1050109	1.000000
X1030167	E1240217	17.888390
X1030168	E1350199	1.000000
X1030168	E1240216	-1.000000
X1030168	E4000101	.050000
X1030173	E1350199	1.000000
X1030173	E1290200	1.000000
X1030173	E4000101	.041000
X1030175	E1300201	1.000000
X1030175	E1300202	-22.795055
X1030175	E1280159	1.000000
X1030176	E1310204	-72.666306
X1030176	E1280159	1.000000
X1030176	E1310203	1.000000
X1030182	E1360172	1.000000
X1030182	E1300201	-1.000000
X1030183	E4000101	.035000
X1030183	E1300202	1.000000
X1030183	E1350199	1.000000
X1030184	E1380174	1.000000
X1030184	E1310205	-1.000000
X1030185	E1050109	1.000000
X1030185	E1310203	-1.000000
X1030185	E1310205	32.432176
X1030186	E1350199	1.000000
X1030186	E1310204	1.000000
X1030186	E4000101	.050000
X1030199	E1290200	-.950000
X1030199	E1410178	1.000000
X1040104	E1350218	-1.000000
X1040104	E1260157	1.000000
X1040133	E1200225	-1.000000
X1040133	E1200226	18.570003
X1040133	E1130128	1.000000
X1040134	E1210227	1.000000
X1040134	E1210228	-18.570003
X1040134	E1130128	1.000000

LISTING VIII.10 - CONTINUED

X1040135	E1220230	-40.963808	X1040176	E1310222	1.000000
X1040135	E1130128	1.000000	X1040176	E1310223	-72.666306
X1040135	E1220229	1.000000	X1040176	E1280159	1.000000
X1040136	E1230231	-1.000000	X1040182	E1360172	1.000000
X1040136	E1230232	18.570003	X1040182	E1300220	-1.000000
X1040136	E1130128	1.000000	X1040183	E1300221	1.000000
X1040137	E1240234	-1.000000	X1040183	E4000101	.035000
X1040137	E1240235	40.963808	X1040183	E1350218	1.000000
X1040137	E1130128	1.000000	X1040184	E1310224	-1.000000
X1040154	E1200225	1.000000	X1040184	E1380174	1.000000
X1040154	E1270158	1.000000	X1040185	E1310222	-1.000000
X1040155	E1370173	1.000000	X1040185	E1310224	32.432176
X1040155	E1200226	22.795055	X1040185	E1050109	1.000000
X1040155	E1200225	1.000000	X1040186	E1350218	1.000000
X1040156	E1200226	-1.000000	X1040186	E1310223	1.000000
X1040156	E4000101	.040000	X1040186	E4000101	.050000
X1040156	E1350218	1.000000	X1040199	E1290219	-.950000
X1040157	E1270158	1.000000	X1040199	E1410178	1.000000
X1040157	E1210227	-1.000000	X1050104	E1350237	-1.000000
X1040158	E1350218	1.000000	X1050133	E1200244	-1.000000
X1040158	E1210228	1.000000	X1050133	E1200245	18.570003
X1040158	E4000101	.035000	X1050133	E1130128	1.000000
X1040159	E1370173	1.000000	X1050134	E1210247	-18.570003
X1040159	E1220229	-1.000000	X1050134	E1130128	1.000000
X1040160	E1220230	1.000000	X1050134	E1210246	1.000000
X1040160	E1350218	1.000000	X1050135	E1220249	-40.963808
X1040160	E4000101	.035000	X1050135	E1220248	1.000000
X1040161	E1230233	-1.000000	X1050135	E1130128	1.000000
X1040161	E1380174	1.000000	X1050136	E1130128	1.000000
X1040162	E1230231	1.000000	X1050136	E1230250	-1.000000
X1040162	E1270158	1.000000	X1050136	E1230251	18.570003
X1040163	E1050109	1.000000	X1050137	E1240253	-1.000000
X1040163	E1230231	1.000000	X1050137	E1240254	40.963808
X1040163	E1230233	16.768744	X1050137	E1130128	1.000000
X1040163	E1230232	26.340818	X1050154	E1200244	1.000000
X1040164	E1230232	-1.000000	X1050154	E1270158	1.000000
X1040164	E1350218	1.000000	X1050155	E1370173	1.000000
X1040164	E4000101	.050000	X1050155	E1200244	1.000000
X1040165	E1380174	1.000000	X1050155	E1200245	22.795055
X1040165	E1240236	-1.000000	X1050156	E1350237	1.000000
X1040166	E1240234	1.000000	X1050156	E1200245	-1.000000
X1040166	E1370173	1.000000	X1050156	E4000101	.040000
X1040167	E1240234	1.000000	X1050157	E1210246	-1.000000
X1040167	E1240236	17.888390	X1050157	E1270158	1.000000
X1040167	E1240235	4.576571	X1050158	E1350237	1.000000
X1040167	E1050109	1.000000	X1050158	E4000101	.035000
X1040168	E1350218	1.000000	X1050158	E1210247	1.000000
X1040168	E1240235	-1.000000	X1050159	E1370173	1.000000
X1040168	E4000101	.050000	X1050159	E1220248	-1.000000
X1040173	E1350218	1.000000	X1050160	E1220249	1.000000
X1040173	E1290219	1.000000	X1050160	E1350237	1.000000
X1040173	E4000101	.041000	X1050160	E4000101	.035000
X1040175	E1300221	-22.795055	X1050161	E1230252	-1.000000
X1040175	E1280159	1.000000	X1050161	E1380174	1.000000
X1040175	E1300220	1.000000	X1050162	E1270158	1.000000

LISTING VIII.10 - CONTINUED

X1050162	E1230250	1.000000
X1050163	E1230250	1.000000
X1050163	E1230251	26.340818
X1050163	E1230252	16.768744
X1050163	E1050109	1.000000
X1050164	E1350237	1.000000
X1050164	E1230251	-1.000000
X1050164	E4000101	.050000
X1050165	E1240255	-1.000000
X1050165	E1380174	1.000000
X1050166	E1370173	1.000000
X1050166	E1240253	1.000000
X1050167	E1050109	1.000000
X1050167	E1240253	1.000000
X1050167	E1240255	17.888390
X1050167	E1240254	4.576571
X1050168	E1350237	1.000000
X1050168	E1240254	-1.000000
X1050168	E4000101	.050000
X1050173	E1350237	1.000000
X1050173	E1290238	1.000000
X1050173	E4000101	.041000
X1050175	E1280159	1.000000
X1050175	E1300239	1.000000
X1050175	E1300240	-22.795055
X1050176	E1280159	1.000000
X1050176	E1310242	-72.666306
X1050176	E1310241	1.000000
X1050182	E1360172	1.000000
X1050182	E1300239	-1.000000
X1050183	E1300240	1.000000
X1050183	E1350237	1.000000
X1050183	E4000101	.035000
X1050184	E1380174	1.000000
X1050184	E1310243	-1.000000
X1050185	E1310243	32.432176
X1050185	E1310241	-1.000000
X1050185	E1050109	1.000000
X1050186	E1350237	1.000000
X1050186	E1310242	1.000000
X1050186	E4000101	.050000
X1050199	E1290238	-.950000
X1050199	E1410178	1.000000
X1060104	E1350256	-1.000000
X1060133	E1200263	-1.000000
X1060133	E1130128	1.000000
X1060133	E1200264	18.570003
X1060134	E1210265	1.000000
X1060134	E1210266	-18.570003
X1060134	E1130128	1.000000
X1060135	E1220267	1.000000
X1060135	E1220268	-40.963808
X1060135	E1130128	1.000000
X1060136	E1130128	1.000000
X1060136	E1230269	-1.000000
X1060136	E1230270	18.570003
X1060137	E1240272	-1.000000
X1060137	E1240273	40.963808
X1060137	E1130128	1.000000
X1060154	E1270158	1.000000
X1060154	E1200263	1.000000
X1060155	E1200264	22.795055
X1060155	E1200263	1.000000
X1060155	E1370173	1.000000
X1060156	E1350256	1.000000
X1060156	E1200264	-1.000000
X1060156	E4000101	.040000
X1060157	E1270158	1.000000
X1060157	E1210265	-1.000000
X1060158	E1210266	1.000000
X1060158	E4000101	.035000
X1060158	E1350256	1.000000
X1060159	E1220267	-1.000000
X1060159	E1370173	1.000000
X1060160	E1350256	1.000000
X1060160	E1220268	1.000000
X1060160	E4000101	.0[illegible]5000
X1060161	E1380174	1.000000
X1060161	E1230271	-1.000000
X1060162	E1230269	1.000000
X1060162	E1270158	1.000000
X1060163	E1050109	1.000000
X1060163	E1230269	1.000000
X1060163	E1230271	16.768744
X1060163	E1230270	26.340818
X1060164	E1350256	1.000000
X1060164	E1230270	-1.000000
X1060164	E4000101	.050000
X1060165	E1380174	1.000000
X1060165	E1240274	-1.000000
X1060166	E1370173	1.000000
X1060166	E1240272	1.000000
X1060167	E1240274	17.888390
X1060167	E1240273	4.576571
X1060167	E1240272	1.000000
X1060167	E1050109	1.000000
X1060168	E1350256	1.000000
X1060168	E1240273	-1.000000
X1060168	E4000101	.050000
X1060173	E1290257	1.000000
X1060173	E1350256	1.000000
X1060173	E4000101	.041000
X1060175	E1280159	1.000000
X1060175	E1300258	1.000000
X1060175	E1300259	-22.795055
X1060176	E1310261	-72.666306
X1060176	E1310260	1.000000
X1060176	E1280159	1.000000
X1060182	E1360172	1.000000
X1060182	E1300258	-1.000000

LISTING VIII.10 - CONTINUED

X1060183	E1350256	1.000000	X1070164	E1350275	1.000000
X1060183	E1300259	1.000000	X1070164	E1230289	-1.000000
X1060183	E4000101	.035000	X1070164	E4000101	.050000
X1060184	E1310262	-1.000000	X1070165	E1380174	1.000000
X1060184	E1380174	1.000000	X1070165	E1240293	-1.000000
X1060185	E1050109	1.000000	X1070166	E1370173	1.000000
X1060185	E1310262	32.432176	X1070166	E1240291	1.000000
X1060185	E1310260	-1.000000	X1070167	E1240293	17.888390
X1060186	E1310261	1.000000	X1070167	E1050109	1.000000
X1060186	E1350256	1.000000	X1070167	E1240292	4.576571
X1060186	E4000101	.050000	X1070167	E1240291	1.000000
X1060199	E1410178	1.000000	X1070168	E1240292	-1.000000
X1060199	E1290257	-.950000	X1070168	E1350275	1.000000
X1070104	E1350275	-1.000000	X1070168	E4000101	.050000
X1070133	E1200282	-1.000000	X1070173	E1350275	1.000000
X1070133	E1200283	18.570003	X1070173	E1290276	1.000000
X1070133	E1130128	1.000000	X1070173	E4000101	.041000
X1070134	E1130128	1.000000	X1070175	E1300277	1.000000
X1070134	E1210284	1.000000	X1070175	E1300278	-22.795055
X1070134	E1210285	-18.570003	X1070175	E1280159	1.000000
X1070135	E1220286	1.000000	X1070176	E1310279	1.000000
X1070135	E1220287	-40.963808	X1070176	E1310280	-72.666306
X1070135	E1130128	1.000000	X1070176	E1280159	1.000000
X1070136	E1230289	18.570003	X1070182	E1360172	1.000000
X1070136	E1130128	1.000000	X1070182	E1300277	-1.000000
X1070136	E1230288	-1.000000	X1070183	E1350275	1.000000
X1070137	E1130128	1.000000	X1070183	E1300278	1.000000
X1070137	E1240291	-1.000000	X1070183	E4000101	.035[illegible]00
X1070137	E1240292	40.963808	X1070184	E1380174	1.000000
X1070154	E1200282	1.000000	X1070184	E1310281	-1.000000
X1070154	E1270158	1.000000	X1070185	E1310281	32.432176
X1070155	E1200282	1.000000	X1070185	E1050109	1.000000
X1070155	E1370173	1.000000	X1070185	E1310279	-1.000000
X1070155	E1200283	22.795055	X1070186	E1350275	1.000000
X1070156	E1200283	-1.000000	X1070186	E1310280	1.000000
X1070156	E1350275	1.000000	X1070186	E4000101	.050000
X1070156	E4000101	.040000	X1070199	E1410178	1.000000
X1070157	E1210284	-1.000000	X1070199	E1290276	-.950000
X1070157	E1270158	1.000000	X1080104	E1350294	-1.000000
X1070158	E1210285	1.000000	X1080133	E1130128	1.000000
X1070158	E1350275	1.000000	X1080133	E1200301	-1.000000
X1070158	E4000101	.035000	X1080133	E1200302	18.570003
X1070159	E1370173	1.000000	X1080134	E1210304	-18.570003
X1070159	E1220286	-1.000000	X1080134	E1210303	1.000000
X10701[illegible]	E1220287	1.000000	X1080134	E1130128	1.000000
X1070160	E1350275	1.000000	X1080135	E1130128	1.000000
X1070160	E4000101	.035000	X1080135	E1220306	-40.963808
X1070161	E1380174	1.000000	X1080135	E1220305	1.000000
X1070161	E1230290	-1.000000	X1080136	E1130128	1.000000
X1070162	E1230288	1.000000	X1080136	E1230307	-1.000000
X1070162	E1270158	1.000000	X1080136	E1230308	18.570003
X1070163	E1230290	16.768744	X1080137	E1240311	40.963808
X1070163	E1230288	1.000000	X1080137	E1240310	-1.000000
X1070163	E1230289	26.340818	X1080137	E1130128	1.000000
X1070163	E1050109	1.000000	X1080154	E1200301	1.000000

LISTING VIII.10 - CONTINUED

X1080154	E1270158	1.000000	X1080185	E1050109	1.000000
X1080155	E1370173	1.000000	X1080185	E1310298	-1.000000
X1080155	E1200302	22.795055	X1080185	E1310300	32.432176
X1080155	E1200301	1.000000	X1080186	E1350294	1.000000
X1080156	E1350294	1.000000	X1080186	E1310299	1.000000
X1080156	E4000101	.040000	X1080186	E4000101	.050000
X1080156	E1200302	-1.000000	X1080199	E1290295	-.950000
X1080157	E1270158	1.000000	X1080199	E1410178	1.000000
X1080157	E1210303	-1.000000	X1090104	E1350313	-1.000000
X1080158	E1210304	1.000000	X1090133	E1200320	-1.000000
X1080158	E1350294	1.000000	X1090133	E1200321	18.570003
X1080158	E4000101	.035000	X1090133	E1130128	-1.000000
X1080159	E1220305	-1.000000	X1090134	E1210323	-18.570003
X1080159	E1370173	1.000000	X1090134	E1130128	1.000000
X1080160	E1220306	1.000000	X1090134	E1210322	1.000000
X1080160	E1350294	1.000000	X1090135	E1220324	1.000000
X1080160	E4000101	.035000	X1090135	E1220325	-40.963808
X1080161	E1230309	-1.000000	X1090135	E1130128	1.000000
X1080161	E1380174	1.000000	X1090136	E1130128	1.000000
X1080162	E1230307	1.000000	X1090136	E1230326	-1.000000
X1080162	E1270158	1.000000	X1090136	E1230327	18.570003
X1080163	E1230307	1.000000	X1090137	E1240329	-1.000000
X1080163	E1230309	16.768744	X1090137	E1130128	1.000000
X1080163	E1050109	1.000000	X1090137	E1240330	40.963808
X1080163	E1230308	26.340818	X1090154	E1200320	1.000000
X1080164	E1230308	-1.000000	X1090154	E1270158	1.000000
X1080164	E1350294	1.000000	X1090155	E1200321	22.795055
X1080164	E4000101	.050000	X1090155	E1370173	1.000000
X1080165	E1380174	1.000000	X1090155	E1200320	1.000000
X1080165	E1240312	-1.000000	X1090156	E1200321	-1.000000
X1080166	E1240310	1.000000	X1090156	E1350313	1.000000
X1080166	E1370173	1.000000	X1090156	E4000101	.040000
X1080167	E1240312	17.888390	X1090157	E1270158	1.000000
X1080167	E1240311	4.576571	X1090157	E1210322	-1.000000
X1080167	E1050109	1.000000	X1090158	E1210323	1.000000
X1080167	E1240310	1.000000	X1090158	E1350313	1.000000
X1080168	E1350294	1.000000	X1090158	E4000101	.035000
X1080168	E1240311	-1.000000	X1090159	E1370173	1.000000
X1080168	E4000101	.050000	X1090159	E1220324	-1.000000
X1080173	E1350294	1.000000	X1090160	E1350313	1.000000
X1080173	E1290295	1.000000	X1090160	E1220325	1.000000
X1080173	E4000101	.041000	X1090160	E4000101	.035000
X1080175	E1300296	1.000000	X1090161	E1380174	1.000000
X1080175	E1300297	-22.795055	X1090161	E1230328	-1.000000
X1080175	E1280159	1.000000	X1090162	E1270158	1.000000
X1080176	E1310298	1.000000	X1090162	E1230326	1.000000
X1080176	E1310299	-72.666306	X1090163	E1230326	1.000000
X1080176	E1280159	1.000000	X1090163	E1230327	26.340818
X1080182	E1300296	-1.000000	X1090163	E1230328	16.768744
X1080182	E1360172	1.000000	X1090163	E1050109	1.000000
X1080183	E1300297	1.000000	X1090164	E1230327	-1.000000
X1080183	E1350294	1.000000	X1090164	E1350313	1.000000
X1080183	E4000101	.035000	X1090164	E4000101	.050000
X1080184	E1310300	-1.000000	X1090165	E1380174	1.000000
X1080184	E1380174	1.000000	X1090165	E1240331	-1.000000

LISTING VIII.10 - CONTINUED

X1090166	E1240329	1.000000	X1100156	E1200340	-1.000000
X1090166	E1370173	1.000000	X1100156	E4000101	.040000
X1090167	E1240329	1.000000	X1100157	E1270158	1.000000
X1090167	E1050109	1.000000	X1100157	E1210341	-1.000000
X1090167	E1240330	4.576571	X1100158	E1350332	1.000000
X1090167	E1240331	17.888390	X1100158	E1210342	1.000000
X1090168	E1350313	1.000000	X1100158	E4000101	.035000
X1090168	E1240330	-1.000000	X1100159	E1220343	-1.000000
X1090168	E4000101	.050000	X1100159	E1370173	1.000000
X1090173	E1290314	1.000000	X1100160	E1220344	1.000000
X1090173	E1350313	1.000000	X1100160	E1350332	1.000000
X1090173	E4000101	.041000	X1100160	E4000101	.035000
X1090175	E1300316	-22.795055	X1100161	E1380174	1.000000
X1090175	E1280159	1.000000	X1100161	E1230347	-1.000000
X1090175	E1300315	1.000000	X1100162	E1230345	1.000000
X1090176	E1310317	1.000000	X1100162	E1270158	1.000000
X1090176	E1310318	-72.666306	X1100163	E1230347	16.768744
X1090176	E1280159	1.000000	X1100163	E1230345	1.000000
X1090182	E1300315	-1.000000	X1100163	E1050109	1.000000
X1090182	E1360172	1.000000	X1100163	E1230346	26.340818
X1090183	E1300316	1.000000	X1100164	E1230346	-1.000000
X1090183	E1350313	1.000000	X1100164	E1350332	1.000000
X1090183	E4000101	.035000	X1100164	E4000101	.050000
X1090184	E1310319	-1.000000	X1100165	E1240350	-1.000000
X1090184	E1380174	1.000000	X1100165	E1380174	1.000000
X1090185	E1050109	1.000000	X1100166	E1240348	1.000000
X1090185	E1310319	32.432176	X1100166	E1370173	1.000000
X1090185	E1310317	-1.000000	X1100167	E1240349	4.576571
X1090186	E1310318	1.000000	X1100167	E1240350	17.888390
X1090186	E4000101	.050000	X1100167	E1240348	1.000000
X1090186	E1350313	1.000000	X1100167	E1050109	1.000000
X1090199	E1290314	-.950000	X1100168	E1350332	1.000000
X1090199	E1410178	1.000000	X1100168	E1240349	-1.000000
X1100104	E1350332	-1.000000	X1100168	E4000101	.050000
X1100133	E1200339	-1.000000	X1100173	E1350332	1.000000
X1100133	E1130128	1.000000	X1100173	E1290333	1.000000
X1100133	E1200340	18.570003	X1100173	E4000101	.041000
X1100134	E1210342	-18.570003	X1100175	E1300334	1.000000
X1100134	E1210341	1.000000	X1100175	E1300335	-22.795055
X1100134	E1130128	1.000000	X1100175	E1280159	1.000000
X1100135	E1220344	-40.963808	X1100176	E1310336	- 1.000000
X1100135	E1130128	1.000000	X1100176	E1310337	-72.666306
X1100135	E1220343	1.000000	X1100176	E1280159	1.000000
X1100136	E1230345	-1.000000	X1100182	E1360172	1.000000
X1100136	E1130128	1.000000	X1100182	E1300334	-1.000000
X1100136	E1230346	18.570003	X1100183	E1300335	1.000000
X1100137	E1240349	40.963808	X1100183	E1350332	1.000000
X1100137	E1130128	1.000000	X1100183	E4000101	.035000
X1100137	E1240348	-1.000000	X1100184	E1380174	1.000000
X1100154	E1270158	1.000000	X1100184	E1310338	-1.000000
X1100154	E1200339	1.000000	X1100185	E1310336	-1.000000
X1100155	E1370173	1.000000	X1100185	E1310338	32.432176
X1100155	E1200340	22.795055	X1100185	E1050109	1.000000
X1100155	E1200339	1.000000	X1100186	E1310337	1.000000
X1100156	E1350332	1.000000	X1100186	E1350332	1.000000

LISTING VIII.10 - CONTINUED

```
    X1100186   E4000101          .050000
    X1100199   E1410178         1.000000
    X1100199   E1290333         -.950000
    W2000106   E4000101          .050000
    W2000106   E1060111         -.253721
    W2000107   E1070115         -.901684
    W2000107   E4000101          .050000
    W2000125   E1250155        -1.000000
    W2000125   E1250154        -1.000000
    W2000125   E4000101          .200000
    Z7000194   E5000100       139.300000
    C5000101   E4000102        -1.000000
    C5000101   E7000000         1.000000
    C5000102   E7000000         1.000000
    C5000102   E4000103        -1.000000
    C5000103   E4000101        -1.000000
    C5000103   E4000102         1.090000
BOUNDS
 LO DEMAND     X1000125        75.600000
 LO DEMAND     X1000126       410.000000
 LO DEMAND     X1000171     26140.000000
 LO DEMAND     X1000174      1100.000000
 LO DEMAND     X1000180        10.000000
 LO DEMAND     X1000181        26.300000
 LO DEMAND     X1000194       139.300000
 LO DEMAND     X1050104       184.000000
 LO DEMAND     X1060104      1288.000000
 LO DEMAND     X1070104       257.500000
 LO DEMAND     X1080104       225.000000
 LO DEMAND     X1090104       456.000000
 LO DEMAND     X1100104       368.000000
ENDATA
```

APPENDIX 9 PROGRAM LISTINGS FOR "ODES"

```
(1)  SUBROUTINE INPUT
(2)  SUBROUTINE MAIN
(3)  SUBROUTINE OPTIM
(4)  SUBROUTINE PREPAR
(5)  SUBROUTINE SETVAL
(6)  SUBROUTINE MODULE
(7)  SUBROUTINE TRIAN
(8)  SUBROUTINE CALO
(9)  SUBROUTINE OBJECT
(10) SUBROUTINE OUTPUT
(11) SUBROUTINE TYPE1  (JUNC)     ... JUCTION
(12) SUBROUTINE TYPE2  (ERECIV)   ... ELECTRIC RECEIVER
(13) SUBROUTINE TYPE3  (DEAER)    ... DEAERATOR
(14) SUBROUTINE TYPE4  (PUMP)     ... PUMP
(15) SUBROUTINE TYPE5  (FURN)     ... PYROLYSIS FURNACE
(16) SUBROUTINE TYPE6  (SPLTR)    ... SPLITTER
(17) SUBROUTINE TYPE7  (JUNC1)    ... JUNCTION
(18) SUBROUTINE TYPE8  (WHB)      ... WASTE HEAT BOILER
(19) SUBROUTINE TYPE9  (FDRUM)    ... FLASH DRUM
(20) SUBROUTINE TYPE10(TURBG)     ... GAS TURBINE
(21) SUBROUTINE TYPE11(TURBB)     ... BACK PRESSURE TURBINE
(22) SUBROUTINE TYPE12(VALVE)     ... VALVE
(23) SUBROUTINE TYPE13(TURBC)     ... CONDENSING TURBINE
(24) SUBROUTINE TYPE14(HEATP)     ... STEAM FOR PROCESS HEATING
(25) SUBROUTINE TYPE16(SCOND)     ... SURFACE CONDENSER
(26) SUBROUTINE TYPE17(BOILR)     ... BOILER
(27) SUBROUTINE TYPE21(GTURBB)    ... TUBO-GENERATOR
(28) SUBROUTINE TYPE26(DEAER1)    ... DEAERATOR
(29) SUBROUTINE TYPE27(EQUAL)     ... EQUAL
(30) SUBROUTINE TYPE28(WTHEAT)    ... SOFTNERS
(31) SUBROUTINE TYPE28(SDRUM)     ... STEAM DRUM
(32) SUBROUTINE TYPE30(HEATR)     ... HEATER
(33) SUBROUTINE TYPE29(MOTOR)     ... MOTOR
```

```
(34) SUBROUTINE COST2       ... COST FOR ELECTRIC RECEIVER
(35) SUBROUTINE COST3       ... COST FOR DEAERATOR
(36) SUBROUTINE COST4       ... COST FOR PUMP
(37) SUBROUTINE COST11      ... COST FOR BUCK PRESSURE TURBINE
(38) SUBROUTINE COST17      ... COST FOR BOILER
(39) SUBROUTINE COST21      ... COST FOR GENERATOR TURBINE
(40) SUBROUTINE COST28      ... COST FOR SOFTNERS
(41) SUBROUTINE COST29      ... COST FOR MOTOR
(42) SUBROUTINE COST30      ... COST FOR HEATER
(43) SUBROUTINE ENTS
(44) SUBROUTINE ENTH
(45) SUBROUTINE HLATE
(46) SUBROUTINE TBP
(47) SUBROUTINE ENTV
(48) SUBROUTINE ENTL
(49) SUBROUTINE ENTRO
(50) SUBROUTINE CAL
(51) SUBROUTINE FRANDN
(52) SUBROUTINE AMAX
(53) SUBROUTINE AMIN
(54) SUBROUTINE NEWTON
(55) FUNCTION   EFFTG
(56) FUNCTION   EFFB
(57) FUNCTION   EFFTC
(58) SUBROUTINE FDELH
(59) FUNCTION   EFGTB
(60) FUNCTION   EFFTB
```

```
      SUBROUTINE INPUT (IS,IE,NST,NEN,NIS,NIE,NV,MAXM,NC,ICON)
C     ** SURPROGRAM TO READ INPUT DATA
      DIMENSION IS(NST,NIS),IE(NEN,NIE),ICON(NC,3)
      DIMENSION IDATA(15),TITLE(8),DATA(4)
      COMMON /SYS1/SNEN(1)/SYS2/LENGS(1)/SYS3/NM,KYWRD(1)
      COMMON /OBJ1/JIN,IIN(1)/OBJ2/JOUT,IOUT(1)/SIM2/IL(1)/SIM6/NI(1)
      COMMON /SIM7/NJ(1)/SIM9/NS,ISP(1)/OPT4/RLI(1)/OPT5/RUI(1)
      COMMON /OPT11/NCD,CNST(1)/OPT10/IX(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM,I1,I2,IPOINT
      DATA NS,N,INV,ISEQ,IPOINT,I2,I1/6*0,1/
      NV=0
      READ(5,200) TITLE
      WRITE(6,201)TITLE
200   FORMAT(8A10)
201   FORMAT(1H1,10X,8A10)
      READ(5,300) ISIM,MAXST,MAXEQ,IUNIT
300   FORMAT(5I5)
      WRITE(6,301) ISIM,MAXST,MAXEQ,IUNIT
301   FORMAT(1H0,16X,*ISIM*,5X,*MAXST*,5X,*MAXEQ*,5X,*IUNIT*/11X,5I10/)
      M=MAXST
      READ(5,101) (IS(I,1),I=1,MAXST)
101   FORMAT(16I5)
      WRITE(6,206) (IS(I,1),I=1,MAXST)
206   FORMAT(1H0,10X,*STREAM CODE*/(11X,16I5))
      WRITE(6,207)
207   FORMAT(1H0,10X,*SYSTEM NETWORK*)
1     READ(5,100) IDATA
100   FORMAT(A10  ,14I5)
      WRITE(6,104) IDATA
104   FORMAT(11X,A10,14I5)
      IF(IDATA(1).EQ.1H* )GO TO 2
      IF(IDATA(1).EQ.2H**)GO TO 3
      ISEQ=ISEQ+1
      IE(ISEQ,1)=IDATA(2)
      DO 102 I=1,NM
      IF(IDATA(1).EQ.KYWRD(I)) GO TO 103
102   CONTINUE
```

```
      WRITE(6,208) IDATA(1)
208   FORMAT(1H0,*KEYWORD ERROR *,A10)
103   IE(ISEQ,2)=I
      IE(ISEQ,3)=IPOINT
      IPOINT=IPUINT+IDATA(3)
      DO 10 I=4,NIE
10    IE(ISEQ,I)=IDATA(I)
      J1=IS(IE(ISEQ,6),1)
      IF(J1.EQ.8.O.J1.EQ.9) GO TO 1
      IF(IDATA(1).EQ.KYWRD(1)) GO TO 13
      IF(IDATA(1).NE.KYWRD(6)) GO TO 1
13    NS=NS+1
      ISP(NS)=ISEQ
      GO TO 1
2     I1=ISEQ
      GO TO 1
3     I2=ISEQ
C     FINDING INPUT STREAM AND OUTPUT STREAM FOR SYSTEM
      DO 31 I=I1,I2
      J=IE(I,4)
      DO 31 K=1,J
      NI(IE(I,K+5))=1.
31    CONTINUE
      JOUT=0
      DO 35 I=1,M
      IF(NI(I).EQ.1) GO TO 35
      JOUT=JOUT+1
      IOUT(JOUT)=I
35    CONTINUE
      DO 33 I=I1,I2
      J=IE(I,5)
      DO 33 K=1,J
      J1=IE(I,4)+K+5
      NJ(IE(I,J1))=1.
33    CONTINUE
      JIN =0
```

```
      DO 36 I=1,M
      IS(I,2)=IPOINT
      IF(NJ(I).EQ.1) GO TO 36
      JIN =JIN +1
      IIN(JIN) =I
36    IPOINT=IPOINT+LENGS(IS(I,1))
4     READ(5,100) IDATA
      IF(IDATA(1).EQ.5HEQUIP) GO TO 5
      IF(IDATA(1).EQ.6HSTREAM) GO TO 6
      IF(IDATA(1).EQ.   4HIVAR) GO TO 7
      IF(IDATA(1).EQ.6HCONSTR) GO TO 8
      IF(IDATA(1).EQ.    3HEND) GO TO 9
5     J1=IDATA(2)
      WRITE(6,205) J1
205   FORMAT(1H0,10X,*EQUIPMENT DATA*,I6)
      DO 30 I=1,J1
      READ(5,110) IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
110   FORMAT(A10,I5,4(I5,F10.2))
      WRITE(6,400)IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
400   FORMAT( 11X    ,A10,I5,4(I5,F10.2))
      DO 50 IW=1,ISEQ
      IF(IDATA(2).EQ.IE(IW,1)) GO TO 51
50    CONTINUE
51    J=1
52    K=IE(IW,3)+IDATA(J+2)
      SNEN(K)=DATA(J)
      IF(IDATA(J+3).EQ.0) GO TO 30
      J=J+1
      GO TO 52
30    CONTINUE
      GO TO 4
6     J1=IDATA(2)
      WRITE(6,202) J1
202   FORMAT(1H0,10X,*STREAM DATA*,I9)
      DO 40 I=1,J1
      READ(5,110) IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
      WRITE(6,400)IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
```

```
      J=1
      IF(IDATA(3).EQ.1.AND.IS(IDATA(2),1).GT.3) GO TO 62
61    K=IS(IDATA(2),2)+IDATA(J+2)
      SNEN(K)=DATA(J)
63    IF(IDATA(J+3).EQ.0) GO TO 40
      J=J+1
      GO TO 61
62    SNEN(IS(IDATA(2),2)+1)=DATA(1)
      INV=INV+1
      IL(INV)=IDATA(2)
      GO TO 63
40    CONTINUE
      GO TO 4
7     NV=IDATA(2)
      MAXM=IDATA(3)
      WRITE(6,203) MAXM
203   FORMAT(1H0,10X,*INDEPENDENT VARIABLES*,I9)
      DO 70 I=1,NV
      READ(5,120) (IDATA(K),K=1,4),RLI(I),RUI(I)
120   FORMAT(10X,4I5,2F10.2)
      WRITE(6,210) (IDATA(K),K=1,4),RLI(I),RUI(I)
210   FORMAT(11X,4I5,2F10.2)
      IF(IDATA(2).EQ.2) GO TO 71
      IF(IDATA(4).NE.1) GO TO 76
      INV=INV+1
      IL(INV)=IDATA(3)
76    IX(I)=IS(IDATA(3),2)+IDATA(4)
      GO TO 70
71    DO 74 IW=11,ISEQ
      IF(IDATA(3).EQ.IE(IW,1)) GO TO 75
74    CONTINUE
75    IX(I)=IE(IW,3)+IDATA(4)
70    CONTINUE
      GO TO 4
8     NCD=IDATA(2)
      WRITE(6,204)
204   FORMAT(1H0,10X,*CONSTRAINTS*)
```

```
      DO 80 I=1,NCD
      READ(5,121) (ICON(I,K),K=1,2),(IDATA(K),K=1,3),CNST(I)
      WRITE(6,211)(ICON(I,K),K=1,2),(IDATA(K),K=1,3),CNST(I)
121   FORMAT(10X,5I5, F10.2)
211   FORMAT(11X,5I5,F10.2)
      IF(IDATA(1)-1) 81,82,83
81    ICON(I,3)=0
      GO TO 80
82    ICON(I,3)=IS(IDATA(2),2)+IDATA(3)
      GO TO 80
83    DO 84 IW=I1,ISEQ
      IF(IDATA(2).EQ.IE(IW,1)) GO TO 85
84    CONTINUE
85    ICON(I,3)=IE(IW,3)+IDATA(3)
80    CONTINUE
      GO TO 4
9     RETURN
      END
```

```
      SUBROUTINE MAIN   (A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICON,NC,XA,NV,NN)
C     SUBPROGRAM TO ORGANIZE BASIC MODULES FOR AN OPTIMAL DESIGN
C     OF ARBITRARY ENERGY SYSTEMS
C
      DIMENSION      A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE),XA(NV,NN)
      DIMENSION      ICON(NC,3)
      COMMON /SIM1/   F(1)/SIM2/IL(1)/SIM6/NI(1)/SIM7/NJ(1)/SIM8/NK(1)
      COMMON /SYS1/SNEN(1)/OBJ4/ECON(1)
      COMMON /OPT1/MAXM,X(1)/OPT2/RMIN(1)/OPT3/RMAX(1)/OPT4/RLI(1)
      COMMON /OPT5/RUI(1)/OPT6/G,FUNC(1)/OPT7/XO(1)/OPT8/XR(1)/OPT9/R(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM,I1,I2,IPOINT
      DATA II,IC,KOUNT,ICHECK,IPRINT /4*0,1/
C
      IF(ISIM.LT.4) II=1
C     OPTIMIZATION ROUTINE
10    CALL OPTIM(NV,X,U,RMIN,RMAX,RLI,RUI,G,NN,KOUNT,MAXM,IPRINT,ICHECK,
     1           II,XA,XR,XO,FUNC,R)
C     SET UP DESIGN VARIABLES THROUGH INDEPENDENT VARIABLES AND SPLITERS
      CALL SETVAL(KOUNT,ICHECK,IS,IE,NST,NEN,NIS,NIE,NV,NC,ICON,NN)
      IF(ICHECK.LT.0) GO TO 10
      LOOP=0
      N=0
30    IF(I1.EQ.1) GO TO 20
C     LEVEL-1 IMPLIES MASS BALANCE CALCULATION
      LEVEL=1
      CALL MODULE(LEVEL,1,I1,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICHECK)
      IF(ICHECK.LT.0) GO TO 10
C     LEVEL-2 IMPLIES ENERGY BALANCE CALCULATION
20    LEVEL=2
      CALL MODULE(LEVEL,I1,I2,A,IS,IE,NST,NEN, NEQ,NIS,NIE,ICHECK)
      IF(ICHECK.LT.0) GO TO 10
      IF(ISIM.LT.4) GO TO 11
      IF(KOUNT.NE.1) GO TO 31
C     CHECK THE DEGREE OF FREEDOM
11    I=M-INV-N
      IF(I) 91,92,93
```

```
91    I=-I
      WRITE(6,2000) I
2000  FORMAT(1H0,I5,        - * EXTRA VARIABLES WERE SPECIFIED*)
      CALL EXIT
93    WRITE(6,3000) I
3000  FORMAT(1H0,I5,* MORE VARIABLES SHOULD BE SPECIFIED*)
      CALL EXIT
92    CONTINUE
      DO 85 I=1,M
      F(I)=SNEN(IS(I,2)+1)
85    NI(I)=I
C     SOLVE SIMULTANEOUS LINEAR EQUATIONS
      CALL PREPAR(F,M,N,NST,INV,NI,NJ,NK,IL)
      IC1=IC+1
      IC2=IC
      CALL TRIAN(F,M,N,NST,NJ,NK,IC2)
      GO TO 32
31    CONTINUE
      DO 88 I=1,M
88    F(I)=0.
      DO 86 I=1,INV
86    F(NK(N+I))=SNEN(IS(NK(N+I),2)+1)
      CALL CALO(IC1,IC2,F)
32    IF(ISIM.NE.2) GO TO 94
98    DO 95 I=1,M
      IF(SNEN(IS(I,2)+1).LE.0.1E-10) GO TO 95
      DEF=ABS(SNEN(IS(I,2)+1)-F(I))/SNEN(IS(I,2)+1)
      IF(DEF.GT.0.001) GO TO 96
95    CONTINUF
      GO TO 90
96    IF(LOOP.GE. 9) GO TO 90
      N=0
      LOOP=LOOP+1
      CALL MODULE(LEVEL,I1,I2,A,IS,IE,NST,NEN, NEQ,NIS,NIE,ICHECK)
      DO 97 I=1,M
      IF(F(I).LT.0. ) GO TO 9
97    SNEN(IS(I,2)+1)=F(I)
```

```
      CALL CALO(IC1,IC2,F)
      GO TO 90
9     ICHECK=-1
      GO TO 10
94    DO 87 I=1,M
      IF(F(I).LT.0. ) F(I)=0.1E-10
87    SNEN(IS(I,2)+1)=F(I)
90    IF(ISIM.LT.3) RETURN
C     LEVEL-3 IMPLIES SIZING OF EQUIPMENT AND COST ESTIMATION
      LEVEL=3
      CALL MODULE(LEVEL,I1,I2,A,IS,IE,NST,NEN, NEQ,NIS,NIE,ICHECK)
      IF(ICHECK.LT.0) GO TO 10
C     CALCULATE OBJECTIVE FUNCTION
      CALL        OBJECT(U       ,IS,IE,NST,NEN,NIS,NIE)
      WRITE(6,7000) KOUNT,U,(X(I),I=1,NV),ECON(3),ECON(4)
7000  FORMAT(1H ,10X,I10,7F10.2/(21X,7F10.2))
      IF(II.EQ.0) GO TO 10
      CALL        OUTPUT(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE OPTIM(N,X,U,RMIN,RMAX ,RLI,RUI,G,NN,KOUNT,MAXM,IPRINT,
     1                ICHECK,II,XA,XR,XO,FUNC,R)
C     SUBPROGRAM OF OPTIMIZATION BY COMPLEX METHOD
      DIMENSION X(1),RMIN(1),RMAX(1),RLI(1),RUI(1)
     1          ,XA(N,NN),FUNC(1),XR(1),XO(1),R(1)
      DATA ICNTR,ISOS,K,ALPHA,BETA /0,0,0,1.,0.5/
      IF(II.EQ.1) RETURN
      IF(KOUNT.EQ.0) GO TO 30
      IF(KOUNT.GT.NN) GO TO 60
      IF(ICHECK.GT.0) GO TO 70
      IF(ISOS.GT.10) GO TO 71
      ISOS=ISOS+1
      GO TO 90
71    WRITE(6,1000)
1000  FORMAT(1H0,*INITIALIZATION FAILED*)
      CALL EXIT
70    FUNC(K)=U
      IF(KOUNT.EQ.NN) GO TO 110
30    KOUNT=KOUNT+1
      K=K+1
      ISOS=0
90    CALL FRANUN(R,N,0)
      DO 20 I=1,N
20    X(I)=RLI (I)+(RUI (I)-RLI (I))*R(I)
40    DO 50 I=1,N
50    XA(I,K)=X(I)
      ICHECK=1
      RETURN
60    IF(ICHECK.EQ.-1) GO TO 250
      FUNC(K)=U
      IF(ICNTR.NE.2) GO TO 110
      I=K+1
      IF(I.EQ.NN.AND.I.EQ.IMAX) GO TO 110
      IF(K.EQ.NN) GO TO 110
      KOUNT=KOUNT+1
```

```
100   IF(KOUNT.GE.MAXM) GO TO 120
111   K=K+1
      IF(K.EQ.IMAX) GO TO 111
      DO 130 I=1,N
130   X(I)=XA(I,K)
      ICHECK=1
      RETURN
110   UMIN=AMIN(NN,FUNC,IMIN)
      UMAX=AMAX(NN,FUNC,IMAX)
      CRIT=ABS((UMAX-UMIN)/UMAX)
      IF(CRIT.LT.G) GO TO 310
      IF(KOUNT.LT.MAXM) GO TO 170
120   WRITE(6,1100)
1100  FORMAT(1H0,*ITERATION DID NOT CONVERGE*)
      GO TO 310
170   KOUNT=KOUNT+1
      IF(KOUNT.EQ.NN) GO TO 220
      IF(IMINO.NE.IMIN) GO TO 220
      IF(ICNTR-1) 200,210,220
220   USTOR =FUNC(IMIN)
      DO 230 I=1,N
      XO(I)=XA(I,IMIN)
      XR(I)=0.
      DO 240 J=1,NN
240   XR(I)=XR(I)+XA(I,J)
      XR(I)=(XR(I)-XA(I,IMIN))/(NN-1)
230   X(I)=(1.0+ALPHA)*XR(I)-ALPHA*XO(I)
      K=IMIN
      IMINO=IMIN
      ICNTR=0
      ISOS=0
140   DO 160 I=1,N
      IF(X(I).GT.RMAX(I)) GO TO 250
      IF(X(I).LT.RMIN(I)) GO TO 250
160   CONTINUE
      ICHECK=1
      GO TO 40
```

```
200   ICNTR=1
250   DO 260 I=1,N
260   X(I)=BETA*X(I)+(1.-BETA)*XR(I)
      K=IMIN
      ISOS=ISOS+1
      IF(ISOS.LE.20) GO TO 140
      WRITE(6,1300)(XR(I),RMIN(I),RMAX(I),I=1,N)
1300  FORMAT(1H0,*CENTROID IS OUT OF RANGE*/(1X,3E15.5))
      GO TO 310
210   IF(UMIN.GT.USTOR) GO TO 280
      DO 290 I=1,N
290   XA(I,IMIN)=XO(I)
280   DO 400 J=1,NN
      IF(J.EQ.IMAX) GO TO 400
      DO 300 I=1,N
300   XA(I,J)=(XA(I,IMAX)+XA(I,J))/2.
400   CONTINUE
      K=0
      ICNTR=2
      GO TO 100
310   K=IMAX
      DO 320 I=1,N
320   X(I)=XA(I,IMAX)
      II=1
      RETURN
      END
```

```
      SUBROUTINE PREPAR(F,M,N ,L,INV,NI,NJ,NK,IL)
C     PRETREAMENT TO SOLVE A SET OF SIMULTANEOUS LINEAR EQUATIONS
      DIMENSION F(1),NI(1),NJ(1),NK(1),IL(1)
      JW1=L+2
      NN=N
C     SORTING OF UNKOWN AND KNOWN VARIABLES
      DO 98 I=1,INV
      J=M-I+1
      IT=NI(J)
      DO 99 K=I,INV
      IF(IT.EQ.IL(K)) GO TO 89
99    CONTINUE
      NI(IL(I))=NI(J)
      NI(J)=IL(I)
      GO TO 98
89    IL(K)=IL(I)
98    CONTINUE
C     REARRANGEMENT OF LOCATION OF VARIABLES AND EQUATIONS
      DO 50 I=1,N
50    NJ(I)=I
      DO 51 I=1,M
51    NK(I)=NI(I)
60    K=1
61    DO 30 I=1,NN
      IF(NN.EQ.1) GO TO 80
      ISUM=0
      DO 40 J=1,NN
      IJ=NK(I)
      IB=(    NJ(J )-1)*L+IJ    +JW1
      IF(ABS(F(IB)).LE.0.1E-50) GO TO 40
      ISUM=ISUM+1
      ISM=0
      DO 41 JI=1,NN
      IJ=NK(JI)
      IB=(NJ(J)-1)*L+IJ+JW1
      IF(ABS(F(IB)).LE.0.1E-50) GO TO 41
      ISM=ISM+1
41    CONTINUE
      IF(ISM.NE.1) GO TO 44
      II= I
```

```
      JJ=J
      GO TO 70
44    IF(ISUM.EW.1) GO TO 42
      IF(ISM.GE.ISMO) GO TO 40
42    ISMO=ISM
      JJ=J
40    CONTINUE
      II=I
      IF(ISUM.EW.1) GO TO 70
      IF(ISUM.EW.0) GO TO 62
      IF(K.EQ.1) GO TO 45
      IF(ISUM -  ISUMO) 45,46,30
46    IF(ISMO.GE.ISMOO) GO TO 30
45    JO=JJ
      IO=II
      ISUMO=ISUM
      ISMOO=ISMO
      K=0
30    CONTINUE
      JJ=JO
      II=IO
      GO TO 70
62    WRITE(6,1111)
1111  FORMAT(1HO,*MATRIX IS WRONG*)
      WRITE(6,2222) ISUM,NN
      DO 700 I=1,M
700   WRITE(6,2222) NI(I),NK(I),NJ(I)
2222  FORMAT(1H ,10I5)
      CALL EXIT
70    III=NK(NN)
      JJJ=NJ(NN)
      NK(NN)=NK(II)
      NJ(NN)=NJ(JJ)
      NK(II)=III
      NJ(JJ)=JJJ
      NN=NN-1
      GO TO 60
80    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SETVAL(K,ICHECK,IS,IE,NST,NEN,NIS,NIE,NV,NC,ICON,N)
C     SUBPROGRAM TO SET INDEPENDENT VARIABLES INTO DESIGN VARIABLES
      DIMENSION          IS(NST,NIS),IE(NEN,NIE),ICON(NC,3)
      COMMON /SYS1/SNEN(1)/SIM9/NS,ISP(1)
      COMMON /OPT1/MAXM,X(1)/OPT9/R(1)/OPT10/IX(1)/OPT11/NCD,CNST(1)
      IF(NV.EQ.0) GO TO 70
      DO 10 I=1,NV
   10 SNEN(IX(I))=X(I)
   20 IF(NC.EQ.0) GO TO 70
      DO 30 I=1,NCD
      IF(ICON(I,1)-1) 40,50,60
   40 SNEN(ICON(I,3))=X(ICON(I,2))-CNST(I)
      GO TO 30
   50 IF(X(ICON(I,2)).LT.SNEN(ICON(I,3))) GO TO 30
      IF(K.GT.N) GO TO 51
      CALL FRANUN(R,1,0)
      X(ICON(I,2))=SNEN(ICON(I,3))-0.1*R(I)
      SNEN(IX(ICON(I,2)))=X(ICON(I,2))
      GO TO 30
   51 ICHECK=-1
      RETURN
   60 IF(X(ICON(I,2)).GT.SNEN(ICON(I,3))) GO TO 30
      IF(K.GT.N) GO TO 51
      CALL FRANUN(R,1,0)
      X(ICON(I,2))=SNEN(ICON(I,3))+0.1*R(I)
      SNEN(IX(ICON(I,2)))=X(ICON(I,2))
      GO TO 30
   30 CONTINUE
   70 DO 80 I=1,NS
      J=IE(ISP(I),5)+IE(ISP(I),4)-1
      J3=IE(ISP(I),6)
      DO 80 J1=1,J
      J4=IE(ISP(I),J1+6)
      DO 80 J2=2,4
   80 SNEN(IS(J4,2)+J2)=SNEN(IS(J3,2)+J2)
      RETURN
      END
```

```
      SUBROUTINE MODULE(L,I1,I2,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** SURPROGRAM TO CALL ALL UNIT MODULES
      COMMON /GENRL/NCP,M,N,IEQ
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      DO 100 IEQ=I1,I2
      IF(ICH.LT.0) RETURN
      K=IE(IEQ,2)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,
     1       23,24,25,26,27,28,29,30),K
1     CALL TYPE1 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
2     CALL TYPE2 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
3     CALL TYPE3 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
4     CALL TYPE4 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
5     CALL TYPE5 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
6     CALL TYPE6 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
7     CALL TYPE7 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
8     CALL TYPE8 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
9     CALL TYPE9 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
10    CALL TYPE10(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
11    CALL TYPE11(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
12    CALL TYPE12(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
13    CALL TYPE13(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
14    CALL TYPE14(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
15    CALL TYPE15(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
16    CALL TYPE16(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
17    CALL TYPE17(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
18    CALL TYPE18(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
19    CALL TYPE19(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
20    CALL TYPE20(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
21    CALL TYPE21(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
22    CALL TYPE22(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
23    CALL TYPE23(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
24    CALL TYPE24(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
25    CALL TYPE25(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
26    CALL TYPE26(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
27    CALL TYPE27(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
28    CALL TYPE28(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
29    CALL TYPE29(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH) $ GO TO 100
30    CALL TYPE30(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
100   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRIAN(F,M,N,L,NJ,NK,IR)
C     SUBPROGRAM TO SOLVE A SET OF SIMULTANEOUS LINEAR EQUATIONS
      DIMENSION F(1),NJ(1),NK(1)
      COMMON /TRI/ISUM,ICL
      JW1=L+2
      JW2=L+1
      ICL=IR
      DO 10 K=1,N
      I1=K+1
      IF(K.EQ.1) GO TO 21
      J1=K-1
      DO 20 I=K,M
      ISUM=0
      DO 90 J=1,J1
      IB=(NJ (J)-1)*L+NK(I)+JW1
      IC=(NJ (K)-1)*L+NK(J)+JW1
      IF(ABS(F(IB)*F(IC)).LE.0.1E-50) GO TO 90
        IF(ISUM.NE.0) GO TO 93
      CALL CAL(11,JW1,0,0,F)
93    IF(F(IB).EQ.1.) GO TO 91
      IF(F(IC).EQ.1.) GO TO 92
      CALL CAL(5,JW2,IB,IC,F)
      CALL CAL(1,JW1,JW1,JW2,F)
      GO TO 90
91    CALL CAL(1,JW1,JW1,IC,F)
      GO TO 90
92    CALL CAL(1,JW1,JW1,IB,F)
90    CONTINUE
      IF(ISUM.EQ.0) GO TO 20
      IB=(NJ (K)-1)*L+NK(I)+JW1
      IF(ABS(F(IB)).LE.0.1E-50) GO TO 95
      CALL CAL(3,IB,IB,JW1,F)
      GO TO 20
95    CALL CAL(10,IB,JW1,0,F)
20    CONTINUE
```

```
      IF(I1.GT.N) GO TO 10
21    DO 100 I=I1,N
      ISUM=0
      IF(K.EQ.1) GO TO 111
      DO 110 J=1,J1
      IB=(NJ (J)-1)*L+NK(K)+JW1
      IC=(NJ (I)-1)*L+NK(J)+JW1
      IF(ABS(F(IB)*F(IC)).LE.0.1E-50) GO TO 110
        IF(ISUM.NE.0) GO TO 107
      CALL CAL(11,JW1,0,0,F)
107   IF(F(IB).EQ.1.) GO TO 101
      CALL CAL(5,JW2,IB,IC,F)
      CALL CAL(1,JW1,JW1,JW2,F)
      IF(F(IC).EQ.1.) GO TO 102
      GO TO 110
101   CALL CAL(1,JW1,JW1,IC,F)
      GO TO 110
102   CALL CAL(1,JW1,JW1,IB,F)
110   CONTINUE
111   IB=(NJ (I)-1)*L+NK(K)+JW1
      IC=(NJ (K)-1)*L+NK(K)+JW1
      IF(ABS(F(IC)).LE.0.1E-50) GO TO 999
      IF(ISUM.EQ.0) GO TO 103
      IF(ABS(F(IB)).LE.0.1E-50) GO TO 104
      CALL CAL(3,IB,IB,JW1,F)
106   IF(F(IC).EQ.1.) GO TO 100
      CALL CAL(7,IB,IB,IC,F)
      GO TO 100
999   WRITE(6,1000)
      DO 2224 J=1,M
2224  WRITE(6,2222)        NK(J),NJ(J)
2222  FORMAT(1H ,10I5)
      CALL EXIT
1000  FORMAT(1H0,*PIVOT ERROR *)
```

```
104    IF(F(IC).EQ.1.) GO TO 105
       CALL CAL(8,IB,JW1,IC,F)
       GO TO 100
105    CALL CAL(10,IB,JW1,0,F)
       GO TO 100
103    IF(ABS(F(IB)).GT.0.1E-50) GO TO 106
100    CONTINUE
10     CONTINUE
       DO 120 K=1,N
       IJ=N-K+1
       IJ1=IJ+1
       ISUM=0
       DO 130 J=IJ1,M
       IB=(NJ (IJ)-1)*L+NK(J)+JW1
       IF(ABS(F(IB)).LE.0.1E-50) GO TO 130
         IF(ISUM.NE.0) GO TO 131
       CALL CAL(11,JW1,0,0,F)
131    CALL CAL(5,JW2,NK(J),IB,F)
       CALL CAL(1,JW1,JW1,JW2,F)
130    CONTINUE
       IF(ISUM.EQ.0) GO TO 123
       IC=(NJ (IJ)-1)*L+NK(IJ)+JW1
       IF(F(IC).EQ.1.) GO TO 132
       CALL CAL(8,NK (IJ),JW1,IC,F)
       GO TO 120
132    CALL CAL(10,NK (IJ),JW1,0,F)
       GO TO 120
123    F(NK(IJ))=0.
120    CONTINUE
       IR=ICL
       WRITE(6,2222) ICL
80     CONTINUE
       RETURN
       END
```

```
      SUBROUTINE CALO(I1,I2,F)
C     ** SUBPROGRAM TO COMPUTE ESSENTIAL OPERATIONS FOR SOLVING SIMULTA-
C                  NEOUS EQUATIONS
      DIMENSION F(1)
      COMMON /SIM2/IL(1)/SIM3/II(1)/SIM4/IJ(1)/SIM5/IK(1)
      DO 12 M=I1,I2
      L=IL(M)
      I=II(M)
      J=IJ(M)
      K=IK(M)
      GO TO (1,2,3,4,5,6,7,8,9,10,11),L
1     F(I)=F(J)+F(K)
      GO TO 12
2     F(I)=-F(J)+F(K)
      GO TO 12
3     F(I)= F(J)-F(K)
      GO TO 12
4     F(I)=-F(J)-F(K)
      GO TO 12
5     F(I)= F(J)*F(K)
      GO TO 12
6     F(I)=-F(J)*F(K)
      GO TO 12
7     F(I)= F(J)/F(K)
      GO TO 12
8     F(I)=-F(J)/F(K)
      GO TO 12
9     F(I)= F(J)
      GO TO 12
10    F(I)=-F(J)
      GO TO 12
11    F(I)=0.
12    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE OBJECT(U       ,IS,IE,NST,NEN,NIS,NIE)
C     CALCULATE OBJECTIVE FUNCTION
      DIMENSION                 IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/OBJ1/JIN,IIN(1)/OBJ2/JOUT,IOUT(1)
      COMMON /OBJ3/COST(1)/OBJ4/ECON(1)/OBJ5/ICRIT
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM,I1,I2,IPOINT
C
C     ICRIT 1 ... VENTURE COST
C     ICRIT 2 ... VENTURE PROFIT
C     ICRIT 3 ... ANNUAL OPERATING COST
C     ECON(1) ... APPARENT PROFIT
C     ECON(2) ... PRODUCT SALES INCOME
C     ECON(3) ... OPERATING COST
C     ECON(4) ... TOTAL ERECTED COST
C     ECON(5) ... OPERATING DAYS
C     ECON(6) ... LANG FACTER
C     ECON(7) ... DEPRECIATION+RATE OF RETURN*(1+IW)/(1-TAX)
C                IW .. RATIO OF WORKING CAPITAL TO INVESTMENT
C     ECON(8) ... INCOME TAX
C     ECON(9) ... LABOR AND SUPERVISION
C
      DO 100 I=2,4
100   ECON(I)=0.
      GO TO (1,2,3),ICRIT
2     DO 10 I=1,JOUT
10    ECON(2)=ECON(2)+COST(IS(IOUT(I),1))*SNEN(IS(IOUT(I),2)+1)*ECON(5)
1     DO 30 I=I1,I2
30    ECON(4)=ECON(4)+SNEN(IE(I,3)+1)*ECON(6)
3     DO 20 I=1,JIN
20    ECON(3)=ECON(3)+COST(IS( IIN(I),1))*SNEN(IS( IIN(I),2)+1)*ECON(5)
      ECON(3)=ECON(3)+ECON(9)
      ECON(1)=(ECON(2)-ECON(3)-ECON(4)*ECON(7))
      U=ECON(1)
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM TO PRINT OUT COMPUTATIONAL RESULTS
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SYS3/NM,KYWRD(1)/OBJ1/JIN,IIN(1)
      COMMON /OBJ2/JOUT,IOUT(1)/OBJ3/COST(1)/OBJ4/E(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM,I1,I2,IPOINT
      WRITE(6,5000)
5000  FORMAT(1H0,10X,*STREAM DATA FLOW RATE*)
      DO 200 I=1,M
      IF(I.EQ.M) GO TO 210
      K=IS(I+1,2)-IS(I,2)
      GO TO 200
210   K=IPOINT-IS(I,2)
200   WRITE(6,4000)I,(SNEN(IS(I,2)+J),J=1,K)
4000  FORMAT(1H ,10X,I10,7F10.2/(11X,8F10.2))
      WRITE(6,8000) I2
8000  FORMAT(1H0,10X,*EQUIPMENT DATA*,I6,6X,*COST*)
      DO 100 I=1,I2
      IF(I.EQ.I2) GO TO 110
      K=IE(I+1,3)-IE(I,3)
      GO TO 100
110   K=IS(1,2)-IE(I,3)
100   WRITE(6,1000) KYWRD(IE(I,2)),IE(I,1),(SNEN(IE(I,3)+J),J=1,K)
1000  FORMAT(1H ,10X,A10,I10,6F10.2/(31X,7F10.2))
      WRITE(6,7002)
      WRITE(6,7001) E(1)
      WRITE(6,7000) E(4)
      WRITE(6,6000) E(3)
7000  FORMAT(1H0,10X,*TOTAL ERECTED COST *,F10.1)
7002  FORMAT(1H1,10X,*OPTIMAL SOLUTION    *)
7001  FORMAT(1H0,10X,*OBJECTIVE FUNCTION *,F10.1)
6000  FORMAT(1H0,10X,*OPERATING COST     *,F10.1)
```

```
      DO 1 I=1,7
1     E(I+10)=0.
      DO 10 I=1,JIN
      IF(IS( IIN(I),1).LE.3.O.IS( IIN(I),1).GE.11) GO TO 10
      J=IS( IIN(I),1)+7
      E(J)=E(J)+COST(J-7)*SNEN(IS(IIN(I),2)+1)*E(5)
10    CONTINUE
      E(18)=E(15)+E(16)
      WRITE(6,20) E(12)
      WRITE(6,80) E(11)
      WRITE(6,40) E(14)
      WRITE(6,50) E(18)
      WRITE(6,60) E(13)
      IF(E(17).GT.0.1E-10) WRITE(6,90) E(17)
      WRITE(6,70) E(9)
20    FORMAT(1H0,11X,*FUEL GAS             *,F10.1)
40    FORMAT(1H0,11X,*COOLING WATER        *,F10.1)
50    FORMAT(1H0,11X,*ELECTRICITY          *,F10.1)
60    FORMAT(1H0,11X,*INDUSTRIAL  WATER    *,F10.1)
70    FORMAT(1H0,11X,*LABOR $ SUPERVISION*,F9.1)
80    FORMAT(1H0,11X,*FUEL OIL             *,F10.1)
90    FORMAT(1H0,11X,*BOILER FEED WATER    *,F10.1)
      RETURN
      END
```

```
      SUBROUTINE TYPE1 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** JUNCTION(JUNC) .. SAME CONDITIONS ARE SET FOR ALL INPUT STREAMS
C     IE(IEQ,4)             ... NO. OF INPUT  STREAMS
C     IE(IEQ,5)             ... NO. OF OUTPUT STREAMS
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) RETURN
      N=N+1
      I=IE(IEQ,4)
      DO 10 J=1,I
      K=IE(IEQ,J+5)
10    A(K,N)=1.
      J1=IE(IEQ,5)
      DO 30 J=1,J1
      K=IE(IEQ,I+J+5)
30    A(K,N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE2 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR ELECTRIC RECEIVER (ERECIV)
C     IE(IEQ,4)          ... NO. OF INPUT  STREAMS
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ
      COMMON /SYS1/SNEN(1)
      J=IE(IEQ,4)
      IF(L.EQ.3) GO TO 10
      N=N+1
      DO 20 I=1,J
      I1=IE(IEQ,I+5)
20    A(I1,N)=1.
      J1=IE(IEQ,5)
      DO 30 I=1,J1
      I1=IE(IEQ,I+J+5)
30    A(I1,N)=-1.
10    SUM=0.
      DO 40 I=1,J
      I1=IE(IEQ,I+5)
40    SUM=SUM+SNEN(IS(I1,2)+1)
      SNEN(IE(IEQ,3)+2)=SUM
      CALL COST2(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE3 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** DEAERATOR (DEAER) WITH TWO INPUT STREAMS AND NO HEAT BALANCE
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF DEAERATOR
      COMMON /GENRL/NCP,M,N,IEQ
      COMMON /SYS1/SNEN(1)
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
      N=N+1
      I=IE(IEQ,6)
      A(I,N)=1.
      I=IE(IEQ,7)
      A(I,N)=1.
      A(I3,N)=-1.
      RETURN
10    SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      CALL COST3(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE4 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** FOR FOR PUMP (PUMP)
C     SNEN(IS(I3,2)+3)     ... OUTLET PRESSURE
C     EFP                  ... PUMP EFFICIENCY
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,ID,IUNIT
      COMMON /SYS1/SNEN(1)
      DATA /EFP/ 0.65
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
      N=N+1
      A(I1,N)= 1.
      A(I3,N)=-1.
      N=N+1
      A(I2,N)= 1.
      A(I1,N)=-(SNEN(IS(I3,2)+3)-SNEN(IS(I1,2)+3))*0.0271/EFP
      IF(IUNIT.EQ.2) A(I1,N)=A(I1,N)*0.0318
      RETURN
10    SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      CALL COST4(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPES (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** PYROLYSIS FURNACE (FURN)
C     SNEN(IE(IEQ,3)+2) ... HEAT DUTY SPECIFIED
C     EFF,              ... FURNACE EFFICIENCY
C     HVL               ... HEATING VALUE OF FUEL GAS (BTU/MLB)
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      DATA HVL,EFF/0.291,0.85/
      IF(L.EQ.3) GO TO 10
      N=N+1
      I=IE(IEQ,6)
      A(I,N)=1.
      I=IE(IEQ,6)
      A(I,N)=-1.
      SNEN(IS(IE(IEQ,7),2)+1)=SNEN(IE(IEQ,3)+2)/HVL/EFF
      IF(LOOP.GT.1) RETURN
      INV=INV+1
      IL(INV)=IE(IEQ,7)
      RETURN
10    CALL COSTS(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE6 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** SPLITTER (SPLTR)
C     IE(IEQ,5)           ... NO. OF OUTPUT STREAMS
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) RETURN
      N=N+1
      I=IE(IEQ,5)
      DO 10 J=1,I
      K=IE(IEQ,J+6)
10    A(K,N)=1.
      K=IE(IEQ,6)
      A(K,N)=-1.
      RETURN
      END
      SUBROUTINE TYPE7 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR JUNCTION (JUNC1)
C     IE(IEQ,4)           ... NO. OF INPUT  STREAMS
C     IE(IEQ,5)           ... NO. OF OUTPUT STREAMS
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) RETURN
      N=N+1
      I=IE(IEQ,4)
      DO 10 J=1,I
      K=IE(IEQ,J+5)
10    A(K,N)=1.
      WRITE(6,100) A(K,N),K,N
100   FORMAT(1H ,F15.5,2I10)
      J1=IE(IEQ,5)
      DO 30 J=1,J1
      K=IE(IEQ,I+J+5)
30    A(K,N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE8 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** WASTE HEAT BOILER(WHB)
C     I1                     ... BOILER FEED WATER
C     I2                     ... WASTE HEAT AVAILABLE
C     I3                     ... STEAM GENERATED
C     I4                     ... BLOW-DOWN WATER
C     SNEN(IE(IEQ,3)+2)  ... STEAM GENERATED/FEED
C     SNEN(IE(IEQ,3)+3)  ... HEAT DUTY AVAILABLE
C     SNEN(IS(I1,2)+4)   ... FRACTION OF VAPOR CONTAINED IN THE FEED
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),9E(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM
      IF(L.EQ.3) GO TO 10
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      N=N+1
      A(I1,N)=-1.
      A(I3,N)= 1.
      A(I4,N)=1.
      N=N+1
      A(I3,N)= 1.
      A(I1 ,N)=-SNEN(IE(IEQ,3)+2)
      II=IS(I1,1)-9
      IF(II.EQ.2) E1=SNEN(IS(I1,2)+4)
      CALL ENTH(II,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      E2=SNEN(IE(IEQ,3)+2)
      CALL ENTH(2 ,SNEN(IS(I3,2)+2),SNEN(IS(I2,3)+3),E2)
      SNEN(IS(I1,2)+1)=SNEN(IE(IEQ,3)+3)/(E2-E1)*0.1E4
      IF(LOOP.GT.1) RETURN
      INV=INV+1
      IL(INV)=I1
      RETURN
10    CALL COST8(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE9 (L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** FLASH DRUM(FDRUM)
C     I1                   .... INPUT STREAM
C     I2                   ... OUTPUT STREAM 1
C     I3                   ... OUTPUT STREAM 2
C     SNEN(IE(IEQ,3)+2) ... SEPARATION RATIO TO STREAM I2
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) GO TO 10
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      N=N+1
      A(I1,N)= 1.
      A(I2,N)=-1.
      A(I3,N)=-1.
      N=N+1
      A(I2 ,N)=1.
      A(I1 ,N)=-SNEN(IE(IEQ,3)+2)
      RETURN
10    CALL COSTY(IS,IE,NST,NEN,NIS,NIS)
      RETURN
      END
```

```
      SUBROUTINE TYPE10(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** GAS TURBINE(TURBG)
C     I1                  ... FUEL GAS OUTLET
C     I2                  ... INTERNAL POWER DEMAND
C     I3                  ... POWER GENERATED
C     I4                  ... FUEL GAS OUTLET
C     DELH                ... POWER AVAILABLE FROM FUEL GAS(KW/MLB)
C     SNEN(IE(IEQ,3)+3)   ... POWER CONSUMED / FUEL GAS
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM
      IF(L.EQ.3) GO TO 10
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      N=N+1
      A(I2,N)= 1.
      A(I3,N)= 1.
      CALL FDELH(SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I4,2)+2),
     1           SNEN(IS(I4,2)+3),DELH)
      A(I1,N)=-EFFTG(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3)
     1          ,SNEN(IS(I4,2)+3))*DELH
      IF(IUNIT.EQ.1) A(I1,N)=2.205*A(I1,N)
      N=N+1
      A(I2,N)= 1.
      A(I1 ,N)=-SNEN(IE(IEQ,3)+3)
      N=N+1
      A(I1,N)= 1.
      A(I4,N)=-1.
      SNEN(IS(I3,2)+1)=SNEN(IE(IEQ,3)+2)
      IF(LOOP.GT.1) RETURN
      INV=INV+1
      IL(INV)=I3
      RETURN
   10 CALL COST10(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE11(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** BACK TURBINE(TURBB)
C     I1                   ... STREAM NO. OF INLET STEAM
C     I2                   ... STREAM NO. OF OUTLET STEAM
C     I3                   ... STREAM NO. OF OUTPUT POWER
C     E1                   ... INLET ENTHALPY
C     E2                   ... OUTLET ENTHALPY AFTER ISENTROPIC EXPANSION
C     A                    ... COEFFICIENT MATRIX FOR A SIMULTANEOUS EQUATION
C     L-3                  ... SIZING AND COST ESTIMATION
C     L-2                  ... ENERGY AND MASS BALANCE
C     N                    ... SEQUENCE NUMBER OF BALANCE EQUATIONS
C     IUNIT-1              ... C.G.S UNIT
C     IUNIT-2              ... BTU-LB UNIT
C     SNEN(IS(I1,2)+2)     ... TEMPERATURE OF INLET STEAM
C     SNEN(IS(I1,2)+3)     ... PRESSURE    OF INLET STEAM
C     ENTS(SUBROUTINE)     ... CALCULATE A CONDITION AFTER ISENTROPIC EXPANSI(
C     ENTH(SUBROUTINE)     ... CALCULATE STEAM ENTHALPY
C     EFFTB(FUNCTION)      ... TURBINE EFFICIENCY
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP
      I1=IE(IEQ,6)
      I2=IE(IFQ,7)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
C     STEAM BALANCE
      N=N+1
      A(I1,N)= 1.
      A(I2,N)=-1.
```

```
C     MODEL FOR POWER GENERATED BY STEAM EXPANSION
      N=N+1
      A(I3,N)= 1.
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1        T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      A(I1,N)=-(E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1        SNEN(IS(I1,2)+3))
C     UNIT CONVERSION
      IF(IUNIT.EQ.1) A(I1,N)=A(I1,N)*3.968
      A(I1,N)=A(I1,N)*0.293
      IF(A(I1,N).GT.0.) ICH=-1
      RETURN
C     COSTIMATION FOR TURBINE
10    CALL COST11(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
      SUBROUTINE TYPE12(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** VALVE
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) RETURN
      N=N+1
      I =IE(IEQ,6)
      A(I,N)=1.
      I =IE(IEQ,7)
      A(I,N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE13(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** CONDENSING TURBINE(TURBC)
C     I1                    ... STREAM NO. OF INLET STEAM
C     I2                    ... STREAM NO. OF CONDENSED STEAM
C     I3                    ... STREAM NO. OF POWER OUTPUT
C     SNEN(IS(I1,2)+2)      ... PRESSURE OF OUTLET STEAM (IN HG)
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
      N=N+1
      A(I1,N)= 1.
      A(I2,N)=-1.
      N=N+1
      A(I3,N)= 1.
      P2=SNEN(IS(I2,2)+3)/29.92
      IF(IUNIT.EQ.2) P2=P2*14.70
      CALL ENTS(2,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),P2,E2,
     1          SNEN(IS(I2,2)+2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      A(I1,N)=-(E1-E2)*EFFTC(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1         SNEN(IS(I1,2)+2))
      IF(IUNIT.EQ.1) A(I1,N)=A(I1,N)*3.968
      A(I1,N)=A(I1,N)*0.293
      RETURN
   10 CALL COST13(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE14(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** STEAM FOR PROCESS HEATING(HEATP)
C     SNEN(IE(IEQ,3)+2) ... STEAM DEMAND SPECIFIED FOR PROCESS HEATING
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM
      IF(L.EQ.3) RETURN
      N=N+1
      I =IE(IEQ,7)
      A(I,N)=-1.
      I =IE(IEQ,6)
      A(I,N)=1.
      SNEN(IS(I,2)+1)=SNEN(IE(IEQ,3)+2)
      IF(LOOP.GT.1) RETURN
      INV=INV+1
      IL(INV)=I
      RETURN
      END
```

```
      SUBROUTINE TYPE16(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** SURFACE CONDENSER(SCOND) ... MASS BALANCE ONLY
C     SNEN(IS(I1,2)+2)   ... INLET
C     SNEN(IE(IEQ,3)+2)  ... COOLING WATER / CONDENSATE (MLB/MLB)
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) GO TO 10
      N=N+1
      I =IE(IEQ,8)
      A(I,N)=-1.
      I =IE(IEQ,6)
      A(I,N)=1.
      N=N+1
      A(I,N)=SNEN(IE(IEQ,3)+2)
      I =IE(IEQ,7)
      A(I,N)=-1.
      N=N+1
      A(I,N)=1.
      I =IE(IEQ,9)
      A(I,N)=-1.
      RETURN
10    CALL COST16(IS,IE,NST,NEN,NIS,N9E)
      RETURN
      END
```

```
      SUBROUTINE TYPE17(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR BOILER (BOILR)
C     I1                    ... BOILER FEED WATER
C     I2                    ... POWER FOR DRAFT FAN
C     I3                    ... FUEL REQUIRED
C     I4                    ... STEAM GENERATED
C     I5                    ... BLOW DOWN WATER
C     SNEN(IE(IEQ,3)+4)     ... BLOW DOWN RATIO
C     SNEN(IE(IEQ,3)+5)     ... HEATING VALUE OF FUEL
C     SNEN(IE(IEQ,3)+6)     ... POWER FOR DRAFT FAN / B.F.W
C     T                     ... BOILING POINT AT BOILER PRESSURE
C     EFFB(FUNCTION)        ... BOILER EFFICIENCY
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      I5=IE(IEQ,10)
      IF(L.EQ.3) GO TO 10
C     MASS BALANCE
      N=N+1
      A(I4,N)= 1.
      A(I1,N) =SNEN(IE(IEQ,3)+4)-1.
```

```
      N=N+1
      A(I5,N)= 1.
      A(I1,N) =-SNEN(IE(IEQ,3)+4)
C     ENERGY BALANCE
      N=N+1
      CALL ENTH(3,SNEN(IS(I4,2)+2),SNEN(IS(I4,2)+3),E)
      A(I4,N)=-E
      CALL TBP(SNEN(    IS(I4,2)+3),T)
      CALL ENTH(1,T                          ,SNEN(IS(I4,2)+3),E)
      A(I5,N)=-E
      CALL ENTH(1,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E)
      A(I1,N)=E
      A(I3,N) = EFFB(SNEN(IS(I3,2)+1),SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3))
     1         *SNEN(IE(IEQ,3)+5)
      N=N+1
      A(I2,N)= 1.
      A(I1,N) =-SNEN(IE(IEQ,3)+6)
   10 SNEN(IE(IEQ,3)+2)=SNEN(IS(I4,2)+1)
      CALL COST17(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE21(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR TUBO-GENERATOR (GTURBB)
C     I1                       ... INLET STEAM
C     I2                       ... EXHAUSTED STEAM
C     I3                       ... POWER OUTPUT
C     SNEN(IE(IEQ,3)+4)        ... POWER CONSUMED INTERNALLY / POWER GENERATED
C     EFGTB(FUNCTION)          ... EFFICIENCY OF GENERATOR TURBINE
C     E1                       ... INLET ENTHALPY
C     E2                       ... OUTLET ENTHALPY AFTER ISENTROPIC EXPANSION
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
      N=N+1
      A(I1,N)= 1.
      A(I2,N)=-1.
      N=N+1
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1          T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      A(I1,N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1          SNEN(IS(I1,2)+3))
      IF(IUNIT.EQ.1) A(I1,N)=A(I1,N)*3.968
      A(I1,N)=A(I1,N)*0.293
      A(I3,N)=-1./(1.-SNEN(IE(IEQ,3)+4))
   10 SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      CALL COST21(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE26(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** DEAERATOR (DEAER1) HAVING 3 INPUT STREAMS AND HEAT BALANCE
C     II - 1              ... PURE WATER
C     II - 2              ... WET STEAM
C     II - 3              ... STEAM
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF DEAERATOR
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      IF(L.EQ.3) GO TO 10
C     MASS BALANCE
      N=N+1
      I=IE(IEQ,6)
      A(I,N)=1.
      I=IE(IEQ,7)
      A(I,N)=1.
      I=IE(IEQ,8)
      A(I,N)=1.
      I=IE(IEQ,9)
      A(I,N)=-1.
C     HEAT BALANCE
      N=N+1
      DO 20 J=1,3
      II=IS(IE(IEQ,J+5),1)-9
      I1=IE(IEQ,J+5)
      CALL ENTH(II,SNEN(IS(I1     ,2)+2),SNEN(IS(I1     ,2)+3),E)
   20 A(I1 ,N)=E
      CALL ENTH(1,SNEN(IS(IE(IEQ,9),2)+2),SNEN(IS(IE(IEQ,9),2)+3),E)
      I=IE(IEQ,9)
      A(I,N)=-E
      RETURN
   10 I=IE(IEQ,9)
      SNEN(IE(IEQ,3)+2)=SNEN(IS(I ,2)+1)
      CALL COST3(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE27(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR EQUAL (EQUAL)
      COMMON /GENRL/NCP,M,N,IEQ
      COMMON /SYS1/SNEN(1)
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      IF(L.EQ.3) RETURN
      N=N+1
      I=IE(IEQ,6)
      A(I,N)=1.
      I=IE(IEQ,7)
      A(I,N)=-1.
      RETURN
      END
      SUBROUTINE TYPE28(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     PROGRAM FUR SOFTNERS (WTREAT)
C     I1                     ... MAKE-UP FEED WATER
C     I2                     ... STEAM CONSUMED FOR WASHING
C     I3                     ... TREATED WATER (OUTPUT STREAM)
C     SNEN(IE(IEQ,3)+2)  ... CAPACITY OF SOFTNERS
C     SNEN(IE(IEQ,3)+4)  ... RATIO OF STEAM CONSUMPTION TO FEED
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      IF(L.EQ.3) GO TO 10
      N=N+1
      A(I1,N)= 1.
      A(I3,N)=-1.
      N=N+1
      A(I2,N)= 1.
      A(I1,N) =-SNEN(IE(IEQ,3)+4)
      RETURN
   10 SNEN(IE(IEQ,3)+2)=SNEN(IS(I3,2)+1)
      CALL COST28(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE2[illegible](L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     ** FOR STEAM DRUM (SDRUM)
C     I1                     ... BOILER FEED WATER
C     I2                     ... STEAM GENERATED
C     I3                     ... BLOW-DOWN WATER
C     II - 1                 ... PURE WATER
C     II - 2                 ... WET STEAM
C     II - 3                 ... STEAM
C     SNEN(IE(IEQ,3)+2)      ... STEAM GENERATED/FEED
C     SNEN(IE(IEQ,3)+3)      ... HEAT DUTY SPECIFIED
C     SNEN(IS(I1,2)+4)       ... FRACTION OF VAPOR CONTAINED IN THE FEED
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)/SIM2/IL(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM
      IF(L.EQ.3) GO TO 10
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      N=N+1
      A(I1,N)= 1.
      A(I2,N)=-1.
      A(I3,N)=-1.
      N=N+1
      A(I1 ,N)=-SNEN(IE(IEQ,3)+2)
      II=IS(I1,1)-9
      IF(II.EQ.2) E1=SNEN(IS(I1,2)+4)
      CALL ENTH(II,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      A(I2,N)= 1.
      E2=SNEN(IE(IEQ,3)+2)
      CALL ENTH(2 ,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),E2)
      SNEN(IS(I1,2)+1)=SNEN(IE(IEQ,3)+3)/(E2-E1)*0.1E4
      IF(LOOP.GT.1) RETURN
      INV=INV+1
      IL(INV)=I1
      RETURN
   10 CALL COST7(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE TYPE30(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR HEATER (HEATR)
C     I1                   ... INLET STEAM FOR HEATING
C     I2                   ... BOILER FEED WATER
C     I3                   ... OUTPUT STREAM 1(CONDENSED WATER)
C     I4                   ... OUTPUT STREAM 2(B.F.W. HEATED)
C     SNEN(IE(IEQ,3)+2)  ... HEAT TRANSFER AREA
C     SNEN(IE(IEQ,3)+4)  ... HEAT TRANSFER COEFFICIENT
C     SNEN(IS(I1,2)+2)   ... TEMPERATURE OF INLET STEAM
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      IF(L.EQ.3) GO TO 10
C     MASS BALANCE
      N=N+1
      A(I2,N)= 1.
      A(I4,N)=-1.
      N=N+1
      A(I1,N)= 1.
      A(I3,N)=-1.
C     HEAT RALANCE
      N=N+1
      CALL ENTH(1,SNEN(IS(I4,2)+2),SNEN(IS(I4,2)+3),E)
      A(I4,N)=E
      CALL ENTH(1,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),E)
      A(I2,N)=-E
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E)
      A(I1,N)=-E
      CALL ENTH(1,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),E)
      A(I3,N)= E
      RETURN
```

```
10    CONTINUE
      SNEN(IE(IEQ,3)+2)=1000.*SNEN(IS(I2,2)+1)*(SNEN(IS(I4,2)+2)-SNEN(IS
     1                 (I2,2)+2))/SNEN(IE(IEQ,3)+4)/((SNEN(IS(I1,2)+2)-
     1                  SNEN(IS(I4,2)+2))-(SNEN(IS(I3,2)+2)-SNEN(IS(I2,2
     1        )+2)))    *ALOG((SNEN(IS(I1,2)+2)-SNEN(IS(I4,2)+2))/(SNEN(
     1                  IS(I3,2)+2)-SNEN(IS(I2,2)+2)))/(SNEN(IS(I4,2)+2)
     1                  -SNEN(IS(I2,2)+2))
      CALL COST30(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END


      SUBROUTINE TYPE29(L,A,IS,IE,NST,NEN,NEQ,NIS,NIE,ICH)
C     SUBPROGRAM FOR MOTOR (MOTOR)
C     EFM                  ... MOTOR EFFICIENCY
C
      DIMENSION A(NST,NEQ),IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,INV,LOOP,ISIM
      DATA EFM/0.95/
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      IF(L.EQ.3) GO TO 10
      N=N+1
      A(I1,N)= EFM
      A(I2,N)=-1.
10    SNEN(IE(IEQ,3)+2)=SNEN(IS(I2,2)+1)
      CALL COST21(IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE COST2 (IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR ELECTRIC RECEIVER
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF ELECTRIC RECEIVER
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
       COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.90*0.551
      RETURN
      END
      SUBROUTINE COST3 (IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR DEAERATOR
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF DEAERATOR
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
       COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.80*0.03
      RETURN
      END
      SUBROUTINE COST4 (IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR PUMP
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF PUMP
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
       COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.60*0.072
      RETURN
      END
```

```
      SUBROUTINE COST11(IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR BUCK PRESSURE TURBINE
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF BUCK PRESSURE TURBINE
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.70*0.524
      RETURN
      END
      SUBROUTINE COST17(IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR BOILER
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF BOILER
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.68*67.1
      RETURN
      END
      SUBROUTINE COST21(IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR GENERATOR TURBINE
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF GENERATOR TURBINE
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.72*1.33
      RETURN
      END
```

```
      SUBROUTINE COST28(IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR SOFTNERS
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF SOFTNERS
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
       COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.80*0.05
      RETURN
      END
      SUBROUTINE COST29(IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR MOTOR
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF MOTOR
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
       COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.5*1.106
      RETURN
      END
      SUBROUTINE COST30(IS,IE,NST,NEN,NIS,NIE)
C     COST FUNCTION FOR HEATER
C     SNEN(IE(IEQ,3)+1) ... EQUIPMENT COST
C     SNEN(IE(IEQ,3)+2) ... CAPACITY OF HEATER
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
       COMMON /GENRL/NCP,M,N,IEQ
      SNEN(IE(IEQ,3)+1) =SNEN(IE(IEQ,3)+2)**0.90*0.079
      RETURN
      END
```

```
      SUBROUTINE ENTS(I,TEMP1,PRESS1,PRESS2,E,T)
C     CALCULATE OUTLET CONDITION OF TURBINE BY ISENTROPIC EXPANSION
C     I *** 1 ...   STEAM CONDITION IS UNKNOWN AT TURBINE OUTLET
C     I *** 2 ...   STEAM CONDITION IS WET     AT TURBINE OUTLET
C     I *** 3 ...   STEAM CONDITION IS DRY     AT TURBINE OUTLET
      COMMON /GENRL/ID(5),IUNIT
      CALL ENTRO(TEMP1,PRESS1,S1)
      CALL TBP(PRESS2,T)
      IF(I.EQ.3) GO TO 60
      IF(I.EQ.2) GO TO 40
50    CALL ENTRO(T,PRESS2,SV)
      DELS=ABS(SV-S1)/S1
      IF(DELS.LE.0.003) GO TO 30
      IF(SV.GT.S1) GO TO 40
60    KOUNT=0
      T=T+20.
10    CALL NEWTON(T,DELS,35.0,TEMP1,KOUNT)
      CALL ENTRO(T,PRESS2,S2)
      DELS=ABS(S1-S2)/S1
      IF(KOUNT.EQ.-1) GO TO 30
      IF(DELS.GT.0.003) GO TO 10
      GO TO 30
40    CALL ENTH(1,T,PRESS2,E)
      CALL HLATE(T,DELH)
      IF(IUNIT.EQ.1) GO TO 20
      TT=(T-32.)/1.8
      E=E*0.556
      DELH=DELH*0.556
      GO TO 21
20    TT=T
21    TT=TT+273.16
      X=(TT*S1-E)/DELH
      E=E+X*DELH
      IF(IUNIT.EQ.1) RETURN
      E=E/0.556
      RETURN
30    CALL ENTH(3,T,PRESS2,E)
      RETURN
      END
```

```
      SUBROUTINE ENTH(I,TEMP,PRESS,E)
C     SUBPROGRAM TO COMPUTE ENTHALPY FOR H2O
C     I - 1                ... ENTHALPY FOR WATER
C     I - 2                ... ENTHALPY FOR WET STEAM
C     I - 3                ... ENTHALPY FOR STEAM
C
      IF(I-2) 10,20,30
10    CALL ENTL(TEMP,E)
      RETURN
20    CALL ENTL(TEMP,EL)
      CALL HLATE(TEMP,DELH)
      E=EL+DELH*E
      RETURN
30    CALL ENTV(TEMP,PRESS,E)
      RETURN
      END
      SUBROUTINE HLATE(TEMP,DELH)
C     SUBPRGRAM TO OBTAIN LATENT HEAT OF STEAM
      COMMON /GENRL/ID(5),IUNIT
      T=TEMP
      IF(IUNIT.EQ.2) GO TO 10
      T=T*1.8+32.
      T=T+460.
10    DELH=0.329762E6-0.2105665RE4*T+0.5641524E1*T*T-0.74549032E-2*T**3
     1    +0.4843594lE-5*T**4-0.12422089E-8*T**5
      DELH=DELH/18.
      IF(IUNIT.EQ.2) RETURN
      DELH=DELH*0.556
      RETURN
      END
      SUBROUTINE TBP(PRESS,T)
C     SUBPRGRAM TO OBTAIN BOILING POINT OF STEAM
      COMMON /GENRL/ID(5),IUNIT
      P=PRESS
      IF(IUNIT.EQ.2) GO TO 10
      P=P*14.72
10    T=6872.990297/(14.356947-ALOG(P))+82.724257-460.
      IF(IUNIT.EQ.2) RETURN
      T=(T-32.)/1.8
      RETURN
      END
```

```
      SUBROUTINE ENTV(TEMP,PRESS,E)
C     SUBPRGRAM TO OBTAIN VAPOR ENTHALPY OF STEAM
      COMMON /GENRL/ID(5),IUNIT
      T=TEMP
      P=PRESS
      IF(IUNIT.EQ.1) GO TO 10
      T=(T-32.)/1.8
      P=P/14.70
10    T=T+273.16
      P=P*10330.
      E=479.74+46.007*T/100.+.50418596*(T/100.)**2-11.262055*ALOG(T/100.
     1 )-.7560784E-2*P/(T/100.)**2.7-(11.064326*P+.359629E-4*P*P
     1 +.191179E-29*P**6)/
     1       (T/100.)**8.4-.28003E-11*P**6/(T/100.)**30.5
     1  +.77392744E-13*P**21/(T/100.)**147
      IF(IUNIT.EQ.1) RETURN
      E=E/0.556
      RETURN
      END
      SUBROUTINE ENTL(TEMP,E)
C     SUBPRGRAM TO OBTAIN LIQUID ENTHALPY OF STEAM
      COMMON /GENRL/ID(5),IUNIT
      DATA TBASE/492./
      T=TEMP
      IF(IUNIT.EQ.2) GO TO 10
      T=T*1.8+32.
10    T=T+460.
      E=0.272736E1*(T-TBASE)+0.75731363E-1*(T*T-TBASE*TBASE)/2.
     1 -0.12518875E-3*(T**3-TBASE**3)/3.+0.6937168E-7*(T**4-TBASE**4)/4.
      E=E/18.
      IF(IUNIT.EQ.2) RETURN
      E=E*0.556
      RETURN
      END
```

```
      SUBROUTINE ENTRO(TEMP,PRESS,S)
C     SUBPRGRAM TO OBTAIN ENTROPY OF STEAM
      COMMON /GENRL/ID(5),IUNIT
      T=TEMP
      P=PRESS
      IF(IUNIT.EQ.1) GO TO 10
      T=(T-32.)/1.8
      P=P/14.70
10    T=T+273.16
      P=P*10330.
      S=2.29540+0.34618*ALOG(T/100.)+0.01832*(T/100.)-0.14047/(T/100.)
     1 -0.110213*ALOG(P)-4.22399E-5*P/(T/100.)**3.7-(8.59692E-2*P
     1 +6.2067E-7*P*P-5.93457E-34*P**6)/(T/100.)**9.4-1.90481E-14*P**6/
     1 (T/100.)**31.5-4.31662E-47*P**26/(T/100.)**148
      RETURN
      END
      SUBROUTINE FRANDN(A,N,M)
C     ** RANDOM NUMBER GENERATOR
      DIMENSION A(1)
C     B  IS A MACHINE-DEPENDENT CONSTANT AND B=2.0**(I/2+1)+3.0
C     WHERE  I = NUMBER OF BITS IN AN INTEGER WORD (I=47 FOR CDC6400)
      B=262147.
      X=M
      X=X/0.8719467
      IF(X.NE.0.0)Y=AMOD(ABS(X),3.18967)
      DO 10 K=1,N
      DO 11 J=1,2
11    Y=AMOD(B*Y,1.0)
      A(K)=Y
C     AVOID Y=0. AND Y=1. TO PREVENT DIVIDING INTO ZERO
10    IF(Y.EQ.0.0.OR.Y.EQ.1.0)Y=0.182818285
      RETURN
      END
```

```
      SUBROUTINE CAL(L,I,J,K,F)
C     STORE ESSENTIAL OPERATIONS IN SOLVING SIMULTANEOUS EQUATIONS
      DIMENSION F(1)
      COMMON /SIM2/IL(1)/SIM3/II(1)/SIM4/IJ(1)/SIM5/IK(1)
      COMMON /TRI/  ISUM  ,ICL
      ICL=ICL+1
      IL(ICL)=L
      II(ICL)=I
      IJ(ICL)=J
      IK(ICL)=K
      ISUM=1
      GO TO (1,2,3,4,5,6,7,8,9,10,11),L
1     F(I)=F(J)+F(K)
      GO TO 12
2     F(I)=-F(J)+F(K)
      GO TO 12
3     F(I)= F(J)-F(K)
      GO TO 12
4     F(I)=-F(J)-F(K)
      GO TO 12
5     F(I)= F(J)*F(K)
      GO TO 12
6     F(I)=-F(J)*F(K)
      GO TO 12
7     F(I)= F(J)/F(K)
      GO TO 12
8     F(I)=-F(J)/F(K)
      GO TO 12
9     F(I)= F(J)
      GO TO 12
10    F(I)=-F(J)
      GO TO 12
11    F(I)=0.
12    CONTINUE
      RETURN
      END
```

```
      FUNCTION AMAX (N,X,IMAX)
C     ** MAXIMUM FINDING
      DIMENSION X(1)
      IMAX=1
      W=X(1)
      DO 10 I=2,N
      IF(W.GE.X(I)) GO TO 10
5     W=X(I)
      IMAX=I
10    CONTINUE
      AMAX =W
      RETURN.
      END
      FUNCTION AMIN (N,X,IMIN)
C     ** MINIMUM FINDING
      DIMENSION X(1)
      IMIN=1
      W=X(1)
      DO 10 I=2,N
      IF(W.LE.X(I)) GO TO 10
5     W=X(I)
      IMIN=I
10    CONTINUE
      AMIN =W
      RETURN
      END
```

```
      SUBROUTINE NEWTON(X,F,XL,XU,N)
C     SUBPROGRAM FOR ROOT-FINDING BY NEWTON RAPHSON METHOD
      IF(N-1) 10,20,30
   10 N=1
      XOLD=X
      RETURN
   20 N=2
      FOLD=F
      X=X+XOLD*0.001
      RETURN
   30 N=N+1
      IF(N.GT.30) GO TO 100
      DERF=(F-FOLD)/(X-XOLD)
      DE=ABS(DERF)
      IF(DE  .LT.0.1E-10) GO TO 52
      DX=F/DERF
      FOLD=F
      XOLD=X
      X=X-DX
   40 XXX=(X-XL)*(X-XU)
      IF(XXX.LT.0.   ) RETURN
      X=(X+XOLD)/2.0
      IF(M.GT.10) GO TO 50
      GO TO 40
50    WRITE(6,2000)
2000  FORMAT(1H0,10X,*THE SOLUTION DID NOT EXIST*)
      GO TO 51
  100 WRITE(6,1000)
 1000 FORMAT(1H0,10X,27H** ITERATION OVERLIMITED **)
51    WRITE(6,3000) X,FOLD,XOLD,XL,XU,N
3000  FORMAT(1H0,*X,FOLD,XOLD,XL,XU,N*,5E15.5,I3)
52    N=-1
      RETURN
      END
```

```
      FUNCTION EFFTG(F,T1,P1,P2)
C     EFFICIENCY FOR GAS TURBINE
      EFFTG=0.268
      RETURN
      END
      FUNCTION EFFB(F,T,P)
C     EFFICIENCY FOR BOILER
      EFFB=0.90
      RETURN
      END
      FUNCTION EFFTC(F,TEMP,PRESS)
C     EFFICIENCY FOR CONDENSING TURBINE
      IF(F.GT.1000.) GO TO 10
      EFFTC=0.5
      RETURN
10    IF(F.GT.5000.) GO TO 20
      EFFTC=0.7
      RETURN
20    EFFTC=0.80
      RETURN
      END
      SUBROUTINE FUELH(T1,P1,T2,P2,DELH)
C     ENTHALPY DIFFERENCE AVAILABLE FROM FUEL GAS
      DELH=111.
      RETURN
      END
      FUNCTION EFGTB(F,T,P)
C     EFFICIENCY FOR GENERATOR TURBINE
      EFGTB=0.93*0.75
      RETURN
      END
      FUNCTION EFFTB(F,TEMP,PRESS)
C     EFFICIENCY FOR BACK PRESSURE TURBINE
      IF(F.GT.1000.) GO TO 10
      EFFTB=0.4
      RETURN
10    IF(F.GT.5000.) GO TO 20
      EFFTB=0.55
      RETURN
20    EFFTB=0.7
      RETURN
      END
```

APPENDIX 10 PROGRAM LISTINGS FOR "OSES" AND "OPES"

```
(1)  SUBROUTINE MAIN
(2)  SUBROUTINE MODULE
(3)  SUBROUTINE CAPCTY
(4)  SUBROUTINE DEMAND
(5)  SUBROUTINE OBJECT
(6)  SUBROUTINE OUTPUT
(7)  SUBROUTINE TYPE1  (JUNC)     ... JUCTION
(8)  SUBROUTINE TYPE2  (ERECIV)   ... ELECTRIC RECEIVER
(9)  SUBROUTINE TYPE3  (DEAER)    ... DEAERATOR
(10) SUBROUTINE TYPE4  (PUMP)     ... PUMP
(11) SUBROUTINE TYPE6  (SPLTR)    ... SPLITTER
(12) SUBROUTINE TYPE9  (FDRUM)    ... FLASH DRUM
(13) SUBROUTINE TYPE10(TURBG)     ... GAS TURBINE
(14) SUBROUTINE TYPE11(TURBB)     ... BACK PRESSURE TURBINE
(15) SUBROUTINE TYPE13(TURBC)     ... CONDENSING TURBINE
(16) SUBROUTINE TYPE14(TURBC1)    ... EXTRACTIVE CONDENSING TURBINE
(17) SUBROUTINE TYPE16(SCOND)     ... SURFACE CONDENSER
(18) SUBROUTINE TYPE17(BOILR)     ... BOILER
(19) SUBROUTINE TYPE17(BOILR)     ... BOILER FOR POLYSAR"S EXPANSION STUDY
(20) SUBROUTINE TYPE18(FTURBI)    ... EXTRACTIVE BACK TURBINE
(21) SUBROUTINE TYPE21(GTURBB)    ... TUBO-GENERATOR
(22) SUBROUTINE TYPE22(GTURB1)    ... EXTRACTIVE GENERATOR TURBINE
(23) SUBROUTINE TYPE24(GTURBC)    ... CONDENSING GENERATOR TURBINE
(24) SUBROUTINE TYPE24(GTURBC)    ... CONDENSING GENERATOR TURBINE
                                      FOR POLYSAR"S EXPANSION STUDY
(25) SUBROUTINE TYPE25(GTURC1)    ... EXTRACTIVE CONDENSING GENERATOR TURBINE
(26) SUBROUTINE TYPE27(EQUAL)     ... EQUAL
(27) SUBROUTINE TYPE26(SELECT)    ... DRIVER SELECTION
(28) SUBROUTINE TYPE28(WTHEAT)    ... SOFTNERS
(29) SUBROUTINE TYPE29(MOTOR)     ... MOTOR
(30) SUBROUTINE TYPE30(HEATR)     ... HEATER
(31) SUBROUTINE LPSORT
(32) FUNCTION   EFFTH
(33) FUNCTION   EFGTH
```

```
      SUBROUTINE MAIN (IS,IE,NST,NEN,NIS,NIE,NC,ICON)
C     SUBPROGRAM TO ORGANISE BASIC MODULES FOR SYNTHESIS AND EXPANSION PROBLEMS
      DIMENSION IS(NST,NIS),IE(NEN,NIE),ICON(NC,3)
      DIMENSION IDATA(15),TITLE(8),DATA(4)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ,IDI,KD,IPRIOD
      COMMON /OBJ1/JIN,IIN(1)/OBJ5/JEQ,IEX(1)
      COMMON /SYS1/SNEN(1)/SYS2/LENGS(1)/SYS3/NM,KYWRD(1)
      COMMON /SYS4/NJ(1)/SYS5/NS,ISP(1)/LP5/CNST(1)/OBJ6/IW,ICAP(1)
      DATA NS,N,M,M1,M2,MEQ,ISEQ,IPOINT,JEQ,JIN,IW/11*0/
      READ(5,200) TITLE
      WRITE(6,201)TITLE
200   FORMAT(8A10)
201   FORMAT(1H0,10X,8A10)
      READ(5,300)       MAXST,MAXEQ,IUNIT,IPRIOD,KD
300   FORMAT(5I5)
      WRITE(6,301)       MAXST,MAXEQ,IUNIT,IPRIOD,KD
301   FORMAT(1H0,15X,*MAXST*,5X,*MAXEQ*,5X,*IUNIT*,4X,*IPRIOD*,3X,
     1       *SIMCODE*/11X,5I10)
      READ(5,101) (IS(I,1),I=1,MAXST)
101   FORMAT(16I5)
      WRITE(6,206) (IS(I,1),I=1,MAXST)
206   FORMAT(1H0,10X,*STREAM CODE*/(11X,16I5))
      WRITE(6,207)
207   FORMAT(1H0,10X,*SYSTEM NETWORK*)
      DO 1 K=1,MAXEQ
      READ(5,100) IDATA
100   FORMAT(A10  ,14I5)
      WRITE(6,104) IDATA
104   FORMAT(11X,A10,14I5)
      ISEQ=ISEQ+1
      IE(ISEQ,1)=IDATA(2)
      DO 102 I=1,NM
      IF(IDATA(1).EQ.KYWRD(I)) GO TO 103
102   CONTINUE
      WRITE(6,208) IDATA(1)
208   FORMAT(1H0,*KEYWORD ERROR *,A10)
```

```
103   IE(ISEQ,2)=I
      IE(ISEQ,3)=IPOINT
      IPOINT=IPOINT+IDATA(3)
      DO 10 I=4,NIE
10    IE(ISEQ,I)=IDATA(I)
C     STORE EQ. NO. OF SPLITTER AND JUNC
      J1=IS(IE(ISEQ,6),1)
      IF(J1.EQ.8.O.J1.EQ.9) GO TO 1
      IF(IDATA(1).EQ.4HJUNC  ) GO TO 3
      IF(IDATA(1).NE.5HSPLTR ) GO TO 1
3     NS=NS+1
      ISP(NS)=ISEQ
1     CONTINUE
C     FINDING INPUT STREAM
      DO 33 I=1,ISEQ
      J=IE(I,5)
      DO 33 K=1,J
      J1=IE(I,4)+K+5
      NJ(IE(I,J1))=1.
33    CONTINUE
      DO 36 I=1,MAXST
      IS(I,2)=IPOINT
      IF(NJ(I).EQ.1) GO TO 36
      JIN =JIN +1
      IIN(JIN) =I
36    IPOINT=IPOINT+LENGS(IS(I,1))
C
4     READ(5,100) IDATA
      IF(IDATA(1).EQ.1H*) GO TO 2
      IF(IDATA(1).EQ.5HEQUIP) GO TO 5
      IF(IDATA(1).EQ.6HSTREAM) GO TO 6
      IF(IDATA(1).EQ.6HCONSTR) GO TO 8
      IF(IDATA(1).EQ.   3HEND) GO TO 9
5     J1=IDATA(2)
      WRITE(6,205) J1
205   FORMAT(1H0,10X,*EQUIPMENT DATA*,I6)
```

```
      DO 30 I=1,J1
      READ(5,110) IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
110   FORMAT(A10,I5,4(I5,F10.2))
      WRITE(6,400)IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
400   FORMAT( 11X    ,A10,I5,4(I5,F10.3))
      DO 50 II= 1,ISEQ
      IF(IDATA(2).EQ.IE(II,1)) GO TO 51
50    CONTINUE
51    J=1
52    J2=IDATA(J+2)
      K=IE(II,3)+J2
      SNEN(K)=DATA(J)
      IF(J.GT.3) GO TO 30
      IF(IDATA(J+3).EQ.0) GO TO 30
      J=J+1
      IF(IDATA(1).EQ.6HSELECT ) GO TO 52
      IF(J2.EQ.2) CALL CAPCTY(IPRIOD,IDATA(2),DATA(J),DATA(J-1))
      GO TO 5?
30    CONTINUE
C     PARAMETER SETTING FOR STREAM AROUND SPLITTER
      IF(NS.EQ.0) GO TO 4
      DO 31 I=1,NS
      J=IE(ISP(I),5)+IE(ISP(I),4)-1
      J3=IE(ISP(I),6)
      DO 31 J1=1,J
      J4=IE(ISP(I),J1+6)
      DO 31 J2=2,4
31    SNEN(IS(J4,2)+J2)=SNEN(IS(J3,2)+J2)
      GO TO 4
6     J1=IDATA(2)
      L=IDATA(3)*100
      WRITE(6,202) J1 ,IDATA(3)
202   FORMAT(1H0,10X,*STREAM DATA*,I9,* PERIOD-*,I3)
      DO 40 I=1,J1
      READ(5,110) IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
      WRITE(6,400)IDATA(1),IDATA(2),(IDATA(K+2),DATA(K),K=1,4)
```

```
      J=1
61    J2=IDATA(J+2)
      K=IS(IDATA(2),2)+J2
      SNEN(K)=DATA(J)
      IF(J2.EQ.1) GO TO 62
63    IF(J.GT.3) GO TO 40
      IF(IDATA(J+3).EQ.0) GO TO 40
      J=J+1
      GO TO 61
62    CALL DEMAND(L,IDATA(2),DATA(J))
      GO TO 63
40    CONTINUE
      GO TO 4
2     M=0
      CALL MODULE(L,IS,IE,NST,NEN,NIS,NIE)
      GO TO 4
8     NC=IDATA(2)
      WRITE(6,204)
204   FORMAT(1H0,10X,*CONSTRAINTS*)
      DO 80 I=1,NC
      READ(5,121) (ICON(I,K),K=1,3),CNST(I)
      WRITE(6,211)(ICON(I,K),K=1,3),CNST(I)
80    CONTINUE
121   FORMAT(3I5,F10.2)
211   FORMAT(11X,3I5,F10.2)
      CALL CONSTR(IPRIOD,IS,IE,NST,NEN,NIS,NIE)
9     CALL OBJECT(IPRIOD,IS,IE,NST,NEN,NIS,NIE)
      CALL OUTPUT(IPRIOD,IS,IE,NST,NEN,NIS,NIE)
      RETURN
      END
```

```
      SUBROUTINE MODULE(L,IS,IE,NST,NEN,NIS,NIE)
C     ** SUBPROGRAM TO CALL ALL UNIT MODULES
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ
      DO 100 I  =1,ISEQ
      IEQ=I
      K=IE(IEQ,2)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,
     1       23,24,25,26,27,28,29,30),K
1     CALL TYPE1 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
2     CALL TYPE2 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
3     CALL TYPE3 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
4     CALL TYPE4 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
5     CALL TYPE5 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
6     CALL TYPE6 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
7     CALL TYPE7 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
8     CALL TYPE8 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
9     CALL TYPE9 (L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
10    CALL TYPE10(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
11    CALL TYPE11(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
12    CALL TYPE12(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
13    CALL TYPE13(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
14    CALL TYPE14(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
15    CALL TYPE15(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
16    CALL TYPE16(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
17    CALL TYPE17(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
18    CALL TYPE18(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
19    CALL TYPE19(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
20    CALL TYPE20(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
21    CALL TYPE21(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
22    CALL TYPE22(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
23    CALL TYPE23(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
24    CALL TYPE24(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
25    CALL TYPE25(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
26    CALL TYPE26(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
27    CALL TYPE27(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
28    CALL TYPE28(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
29    CALL TYPE29(L   ,IS,IE,NST,NEN,NIS,NIE)   $ GO TO 100
30    CALL TYPE30(L   ,IS,IE,NST,NEN,NIS,NIE)
100   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CAPCTY(L,K,D1,D2)
C     MODELING FOR DEMAND-SUPPLY AND SUPPLY-CAPACITY RELATIONS
C     KD - 0                 ... EXPANSION PROBLEM
C     KD - 1                 ... SYNTHESIS PROBLEM
C     ID                     ... STREAM NO.,THE ENERGY RATE OF WHICH REPRESENTS
C                                A CAPACITY OF THE UNIT CONSIDERED
C                                IF THIS IS ZERO,THE CAPACITY IS NOT REPRESENTED
C                                IN TERMS OF ENERGY RATE
C     IEX(JEX)               ... UNITS CONSIDERED FOR EXPANSION OR SYNTHESIS
C     IW,ICAP(IW)            ... NO. OF UNITS WHOSE CAPACITY IS NOT REPRESENTED
C                                IN TERMS OF ENERGY RATE AND THE UNIT NO.
C
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ,ID1,KD,IPRIOD
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /OBJ5/JEQ,IEX(1)/OBJ6/IW,ICAP(1)
      J2=K*10000
      ID=D1
      IF(KD.NE.0       ) GO TO 70
      DO 10 J=1,L
      M1=M1+1
      MEQ=MEQ+1
      J1=J*100
      I2=J+1
      DO 20 I=1,I2
      N=N+1
      IROW(N)=M1    +2000000+J2
      ICOL(N)=K+J1+(I-1)*10000+3000000
      RDATA(N)=1.
   20 CONTINUE
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      IF(ID.LT.1) GO TO 50
      ICOL(N)=ID+J1+1000000
      GO TO 60
   50 ICOL(N)=K+J1+2000000
   60 RDATA(N)=-1.
   10 CONTINUE
```

```
      DO 80 J=1,L
      N=N+1
      IROW(N)=6000000
      ICOL(N)=K+J*100+9000000
   80 RDATA(N)=U2
      DO 30 I=1,L
      I2=I*100
      DO 30 J=I ,L
      N=N+1
      M1=M1+1
      MEQ=MEQ+1
      IROW(N)=M1        +3000000+J2
      ICOL(N)=K+I2+4000000
      RDATA(N)=-1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=K+I2*100+J*100+3000000
   30 RDATA(N)=1.
      GO TO 90
   70 IF(L.EQ.1) GO TO 90
      DO 100 J=1,L
      M1=M1+1
      MEQ=MEQ+1
      J1=J*100
      N=N+1
      IROW(N)=M1+3000000+J2+J1
      IF(ID.LT.1) GO TO 110
      ICOL(N)=ID+J1+1000000
      GO TO 120
  110 ICOL(N)=K+J1+2000000
  120 RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
```

```
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=K+4000000+100
100   RDATA(N)=-1.
90    JEQ=JEQ+1
      IEX(JEQ)=K
      IF(ID.NE.1) GO TO 40
      IW=IW+1
      ICAP(IW)=K
40    RETURN
      END


      SUBROUTINE DEMAND(L,NSTRM,DATA)
C     SPECIFING EACH DEMAND REQUIRED
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      N=N+1
      IROW(N)=L+5000000
      ICOL(N)=NSTRM+L+7000000
      RDATA(N)=DATA
      RETURN
      END
```

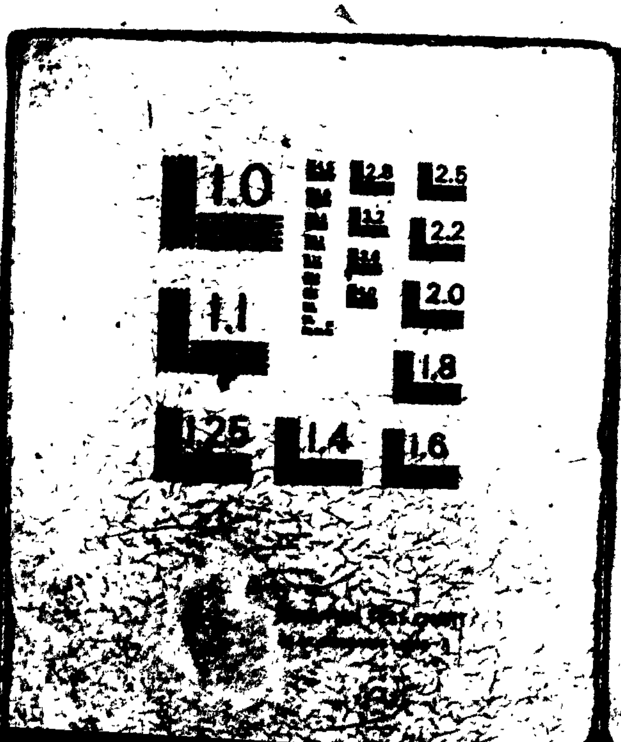
```
      SUBROUTINE OBJECT(K,IS,IE,NST,NEN,NIS,NIE)
C     SET UP MODELS RELATING TO OBJECTIVE FUNCTION
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ,ID1,KK
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /OBJ1/JIN,IIN(1)/OBJ5/JEQ,IEK(1)/OBJ3/COST(1)/OBJ4/ECON(1)
C     OBJECTIVE FUNCTION
      DO 10 J=1,K
      L=J*100
      N=N+1
      IROW(N)=7000000
      ICOL(N)=1+L+5000000
      IF(J.NE.1) GO TO 50
      RDATA(N)=1.
      GO TO 60
50    RDATA(N)=1./(1.+ECON(10))**(J-1)
60    N=N+1
      ICOL(N)=2+L+5000000
      IROW(N)=7000000
10    RDATA(N)=RDATA(N-1)
      MEQ=MEQ+1
      IROW(N)=M2+L +4000000
      ICOL(N)=3+L+5000000
      RDATA(N)=ECON(6)*ECON(7)
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=1+L+5000000
      RDATA(N)=-1.
C     OPERATING COST
      M2=M2+1
      MEQ=MEQ+1
      DO 40 I=1,JIN
      N=N+1
      IROW(N)=M2+L +4000000
      ICOL(N)=IIN(I)+L+1000000
40    RDATA(N)=COST(IS(IIN(I),1))*ECON(5)
```

```
C     ERECTED COST
      DO 20 J=1,K
      M2=M2+1
      MEQ=MEQ+1
      L =J*100
      DO 30 I=1,JEQ
      N=N+1
      IK=IEX(I)
      IF(K.EQ.1.A.KK.NE.0) GO TO 70
      IROW(N)=M2+L +4000000
      ICOL(N)=IEX(I)+L+4000000
      GO TO 30
70    IF(IK.LT.10000) GO TO 31
      IN=IEX(I)/10000
      IK=IEX(I)-IN*10000
31    IF(K.NE.1.A.KK.EQ.0) GO TO 30
      IROW(N)=M2+L+4000000
      ID=SNEN(IE(IK,3)+3)
      IF(ID.EQ.0) GO TO 90
      IF(IEX(I).LT.10000) IN=0
      ICOL(N)=ID+L+1000000+IN*10000
      GO TO 30
90    ICOL(N)=IEX(I)+L+2000000
30    RDATA(N)=SNEN(IE(IK     ,3)+1)
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=3+L+5000000
      RDATA(N)=-1.
C     FIXED COST
      M2=M2+1
      N=N+1
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=2+L+5000000
20    RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(L,IS,IE,NST,NEN,NIS,NIE)
C     GENERATE INPUT DATA TO MPSX
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ,KK ,KD,IPRIOD
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /OBJ5/JEQ,IEX(1)/OBJ6/IW,ICAP(1)
C
      WRITE(6,300)
300   FORMAT(1H1,10X,*LP MODELS FOR OPTIMAL SYNTHESIS OF ENERGY SYSTEM*)
      WRITE(6,400) MEQ,N
400   FORMAT(1H0,10X,*NUMBER OF EQUATION =*,I10//11X,*NUMBER OF COEFF.
     1 =  *,I8/)
      DO 100 I=1,N
      I1=ICOL(I)/1000000
      GO TO (1,2,3,4,5,6,1,2,3),I1
1     WRITE(6,10) IROW(I),ICOL(I),RDATA(I)
      GO TO 100
2     WRITE(6,20) IROW(I),ICOL(I),RDATA(I)
      GO TO 100
3     WRITE(6,30) IROW(I),ICOL(I),RDATA(I)
      GO TO 100
4     WRITE(6,40) IROW(I),ICOL(I),RDATA(I)
      GO TO 100
5     WRITE(6,50) IROW(I),ICOL(I),RDATA(I)
      GO TO 100
6     WRITE(6,60) IROW(I),ICOL(I),RDATA(I)
100   CONTINUE
10    FORMAT(14X,*E*,I7,2X,*X*,I7,2X,F12.6)
20    FORMAT(14X,*E*,I7,2X,*W*,I7,2X,F12.6)
30    FORMAT(14X,*E*,I7,2X,*Y*,I7,2X,F12.6)
40    FORMAT(14X,*E*,I7,2X,*Z*,I7,2X,F12.6)
50    FORMAT(14X,*E*,I7,2X,*C*,I7,2X,F12.6)
60    FORMAT(14X,*E*,I7,2X,*R*,I7,2X,F12.6)
      IF(KK.EQ.0) RETURN
      WRITE(KK,1000)
1000  FORMAT(*NAME*,10X,*ENERGY*)
      WRITE(6 ,1001)
1001  FORMAT(1H0,10X,*NAME*,10X,*ENERGY*)
```

```
      WRITE(KK,1500)
 1500 FORMAT(*ROWS*/1X,*N*,2X,*E7000000*)
      WRITE(6,500)
  500 FORMAT(1H0,10X,*ROWS*/12X,*N*,2X,*E7000000*)
      DO 200 I=1,MEQ
      I1=NEQ(I)/1000000
      IF(I1.NE.3) GO TO 210
      WRITE(6,2100) NEQ(I)
 2100 FORMAT(1H ,11X,*L*,2X,*E*,I7)
      WRITE(KK,1200) NEQ(I)
 1200 FORMAT(1X,*L*,2X,*E*,I7)
      GO TO 200
  210 WRITE(6,2000) NEQ(I)
 2000 FORMAT(1H ,11X,*E*,2X,*E*,I7)
      WRITE(KK,1300) NEQ(I)
 1300 FORMAT(1X,*E*,2X,*E*,I7)
  200 CONTINUE
      WRITE(6,2200)
 2200 FORMAT(1H ,10X,*COLUMNS*)
      WRITE(KK,1400)
 1400 FORMAT(*COLUMNS*)
      K=1
      KD1=KD+1
      DO 220 J=1,KD1
      DO 220 I=1,MAXST
      I2=I+1000000+(J-1)*10000
  220 CALL LPSORT(K,L,I2)
      I1=K-1
      WRITE(6,11)(ICOL(I),IROW(I),RDATA(I),I=1,I1)
   11 FORMAT(15X,*X*,I7,2X,*E*,I7,2X,F12.6)
      WRITE(KK,1210) (ICOL(I),IROW(I),RDATA(I),I=1,I1)
 1210 FORMAT(4X,*X*,I7,2X,*E*,I7,2X,F12.6)
      I1=K
      DO 240 I=1,IW
  240 CALL LPSORT(K,L,ICAP(I)+2000000)
      I2=K-1
      WRITE(6,21)(ICOL(I),IROW(I),RDATA(I),I=I1,I2)
   21 FORMAT(15X,*W*,I7,2X,*E*,I7,2X,F12.6)
      WRITE(KK,1220) (ICOL(I),IROW(I),RDATA(I),I= I1,I2)
 1220 FORMAT(4X,*W*,I7,2X,*E*,I7,2X,F12.6)
```

4 OF/DE 4



```
      I1=K
      I2= L+1
      DO 260 J=1,I2
      J1=(J-1)*10000
      DO 260 I=1,JEQ
260   CALL LPSOHT(K,L,J1+IEX(I)+3000000)
      IF(I1.EQ.K) GO TO 32
      I2=K-1
      WRITE(6,31)(ICOL(I),IROW(I),RDATA(I),I=I1,I2)
31    FORMAT(15X,*Y*,I7,2X,*E*,I7,2X,F12.6)
      WRITE(KK,1230) (ICOL(I),IROW(I),RDATA(I),I= I1,I2)
1230  FORMAT(4X,*Y*,I7,2X,*E*,I7,2X,F12.6)
32    I1=K
      IF(IP.EQ.1.A.KD.NE.0) GO TO 42
      DO 280 J=1,KD1
      DO 280 I=1,JEQ
      IK=IEX(I)
      IF(IK.LT.10000) GO TO 281
      IK=IEX(I)/10000
      IK=IEX(I)-IK*10000
281   I2=IK+4000000+(J-1)*10000
280   CALL LPSOHT(K,L,I2)
      I2=K-1
      WRITE(6,41)(ICOL(I),IROW(I),RDATA(I),I=I1,I2)
41    FORMAT(15X,*Z*,I7,2X,*E*,I7,2X,F12.6)
      WRITE(KK,1240) (ICOL(I),IROW(I),RDATA(I),I= I1,I2)
1240  FORMAT(4X,*Z*,I7,2X,*E*,I7,2X,F12.6)
      I1=K
42    DO 310 I=1,3
310   CALL LPSOHT(K,L,I+5000000)
      I2=K-1
      WRITE(6,51)(ICOL(I),IROW(I),RDATA(I),I=I1,I2)
51    FORMAT(15X,*C*,I7,2X,*E*,I7,2X,F12.6)
      WRITE(KK,1250) (ICOL(I),IROW(I),RDATA(I),I= I1,I2)
1250  FORMAT(4X,*C*,I7,2X,*E*,I7,2X,F12.6)
      I1=K
      CALL LPSOHT(K,L,6000001)
      IF(I1.EQ.K) GO TO 62
      I2=K-1
```

```
      WRITE(6,331)
331   FORMAT(1H ,10X,*RHS*)
      WRITE(KK,2331)
2331  FORMAT(*RHS*)
      WRITE(6,61)(            IROW(I),RDATA(I),I=I1,I2)
61    FORMAT(15X,*R6*,8X,*E*,I7,2X,F12.6)
      WRITE(KK,1260) (            IROW(I),RDATA(I),I= I1,I2)
1260  FORMAT(4X,*R6*,8X,*E*,I7,2X,F12.6)
62    I1=K
      WRITE(6,361)
361   FORMAT(1H ,10X,*BOUNDS*)
      WRITE(KK,1236)
1236  FORMAT(*BOUND*)
      DO 340 J=1,KD1
      DO 340 I=1,MAXST
      I2=I+7000000+(J-1)*10000
340   CALL LPSORT(K,L,I2)
      I2=K-1
      DO 71 I=I1,I2
      ICOL(I)=ICOL(I)-6000000
71    WRITE(6,70) ICOL(I)           ,RDATA(I)
70    FORMAT(1H ,11X,*LO*,1X,*DEMAND*,4X,*X*,I7, 2X,F12.6)
      WRITE(KK,1270) (           ICOL(I),RDATA(I),I= I1,I2)
1270  FORMAT(1X,*LO*,1X,*DEMAND*,4X,      *X*,I7,,2X,F12.6)
      I1=K
      DO 370 I=1,JEQ
370   CALL LPSORT(K,L,IEX(I)+9000000)
      IF(I1.EQ.K) GO TO 93
      I2=K-1
      DO 92 I=I1,I2
      ICOL(I)=ICOL(I)-6000000
92    WRITE(6,90) ICOL(I)           ,RDATA(I)
90    FORMAT(1H ,11X,*UP*,1X,*DEMAND   *,2X,*Y*,I7,2X,F12.6)
      WRITE(KK,1290) (           ICOL(I),RDATA(I),I= I1,I2)
1290  FORMAT(1X,*UP*,1X,*DEMAND  *,2X,*Y*,I7,2X,F12.6)
93    WRITE(6,91)
91    FORMAT(1H ,10X,*ENDATA*)
      WRITE(KK,1291)
1291  FORMAT(*ENDATA*)
      RETURN
      END
```

```
      SUBROUTINE TYPE1 (L,IS,IE,NST,NEN,NIS,NIE)
C     SUBROUTINE FOR JUNCTION
C     IE(IEQ,4)           ... NO. OF INPUT  STREAMS
C     IE(IEQ,5)           ... NO. OF OUTPUT STREAMS
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      M=M+1
      MEQ=MEQ+1
      J=IE(IEQ,4)
      J1=IE(IEQ,1)*10000
      DO 10 I=1,J
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,I+5)+L+1000000
   10 RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      J1=IE(IEQ,5)
      DO 20 I=1,J1
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,I+J+5)+L+1000000
   20 RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE2 (L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR ELECTRIC RECEIVER(ERECIV)
C     IE(IEQ,4)          ... NO. OF INPUT  STREAMS
C     IE(IEQ,5)          ... NO. OF OUTPUT STREAMS
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNFN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
C     POWER BALANCE FOR RECEIVING
      J=IE(IEQ,4)
      J1=IE(IEQ,1)*10000
      M=M+1
      MEQ=MEQ+1
      DO 10 I=1,J
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,I+5)+L+1000000
   10 RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,1)+L+2000000
      RDATA(N)=-1.
C     POWER BALANCE FOR DISTRIBUTING
      M=M+1
      J1=IE(IEQ,5)
      DO 20 I=1,J1
      N=N+1
      IROW(N)=NEQ(MEQ)+1
      ICOL(N)=IE(IEQ,I+J+5)+L+1000000
   20 RDATA(N)=1.
      MEQ=MEQ+1
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,1)+L+2000000
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE3 (L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR DEAERATOR(DEAER ) HAVING 3 INPUT STREAMS AND HEAT BALANCE
C     II - 1                ... PURE WATER
C     II - 2                ... WET STEAM
C     II - 3                ... STEAM
C     SNEN(IS(I1,2)+2)      ... TEMPERATURE OF STREAM I1
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
C     MASS BALANCE
      M=M+1
      MEQ=MEQ+1
      J1=IE(IEQ,1)*10000
      DO 10 J=1,3
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,J+5)+L+1000000
   10 RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,9)+L+1000000
      RDATA(N)=-1.0
C     HEAT BALANCE
      M=M+1
      MEQ=MEQ+1
      DO 20 J=1,3
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,J+5)+L+1000000
      II=IS(IE(IEQ,J+5),1)-9
      I1=IE(IEQ,J+5)
      CALL ENTH(II,SNEN(IS(I1     ,2)+2),SNEN(IS(I1     ,2)+3),E)
```

```
20    RDATA(N)=E
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,9)+L+1000000
      CALL ENTH(1,SNEN(IS(IE(IEQ,9),2)+2),SNEN(IS(IE(IEQ,9),2)+3),E)
      RDATA(N)=-E
      RETURN
      END
```

```
      SUBROUTINE TYPE4 (L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR PUMP (PUMP)
C     IE(IEQ,6)          ... PUMP SUCTION
C     IE(IEQ,7)          ... POWER REQUIRED
C     IE(IEQ,8)          ... PUMP DELIVERY
C     EFP                ... PUMP EFFICIENCY
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      DATA /EFP/ 0.65
      N=N+1
      M=M+1
      MEQ=MEQ+1
      J1=IE(IEQ,1)*10000
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,6)+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,8)+L+1000000
      RDATA(N)=-1.
C     POWER REQUIRED FOR PUMP
      N=N+1
      M=M+1
      MEQ=MEQ+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,7)+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,6)+L+1000000
      RDATA(N)=-(SNEN(IS(IE(IEQ,8),2)+3)-SNEN(IS(IE(IEQ,6),2)+3))*
     1        0.0271/EFP
      IF(IUNIT.EQ.2) RDATA(N)=RDATA(N)*0.0318
      RETURN
      END
```

```
      SUBROUTINE TYPE6 (L,IS,IE,NST,NEN,NIS,NIE)
C     SUBROUTINE FOR SPLITTER (SPLTR)
C     IE(IEQ.5)           ... NO. OF OUTPUT STREAMS
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      M=M+1
      MEQ=MEQ.1
      J=IE(IEQ,5)
      J1=IE(IEQ,1)*10000
      DO 10 I=1,J
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=IE(IEQ,I+6)+L+1000000
   10 RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,6)+L+1000000
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE9 (L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR FLASH DRUM (FDRUM)
C     I1                    ... INPUT STREAM
C     I2                    ... OUTPUT STREAM 1
C     I3                    ... OUTPUT STREAM 2
C     SNEN(IE(IEQ,3)+2) ... SEPARATION RATIO TO STREAM I2
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-SNEN(IE(IEQ,3)+4)
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I3+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=SNEN(IE(IEQ,3)+4)-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE10(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR GAS TURBINE (TURBG)
C     I1                   ... FUEL GAS INLET
C     I2                   ... OUTPUT POWER
C     I3                   ... FUEL GAS OUTLET
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
C     POWER GENERATED / FUEL GAS (KWH/MLB)
      RDATA(N)=-18.9*70.*.98
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)= 1.
      RETURN
      END
```

```
      SUBROUTINE TYPE11(L,IS,IE,NST,NEN,NIS,NIE)
C     PROGRAM FOR BACK TURBINE (TURBB)
C     I1                  ... STREAM NO. OF INLET STEAM
C     I2                  ... STREAM NO. OF OUTLET STEAM
C     I3                  ... STREAM NO. OF OUTPUT POWER
C     E1                  ... INLET ENTHALPY
C     E2                  ... OUTLET ENTHALPY AFTER ISENTROPIC EXPANSION
C     N                   ... SEQUENCE NUMBER OF LP DATA
C     M                   ... NO. OF ENERGY AND MASS BALANCE EQUATIONS
C     MEQ                 ... SEQUENCE NO. OF LP EQUATION
C     IROW(N)             ... INFORMATION FOR ROW IDENTIFICATION
C     ICOL(N)             ... INFORMATION FOR COLUMN IDENTIFICATION
C     NEQ(MEQ)            ... SAME INFORMATION AS IROW,ROW SPECIFICATION
C     SNEN(IS(I1,2)+2)    ... TEMPERATURE OF INLET STEAM
C     SNEN(IS(I1,2)+3)    ... PRESSURE    OF INLET STEAM
C     IUNIT               ... 1 - C.G.S. UNIT , 2 - BTU-LB UNIT
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
C     STEAM BALANCE
      J1=IE(IEQ,1)*10000
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=I1+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+L+1000000
      RDATA(N)=-1.
```

```
C     MODEL FOR POWER GENERATED BY STEAM EXPANSION
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=I3+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+L+1000000
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1          SNEN(IS(I2,2)+2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)=-(E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1          SNEN(IS(I1,2)+3))*0.293
C     UNIT CONVERSION
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      RETURN
      END
```

```
      SUBROUTINE TYPE13(L,IS,IE,NST,NEN,NIS,NIE)
C     PROGRAM FOR CONDENSING TURBINE (TURBC)
C     I1                  ... STREAM NO. OF INLET STEAM
C     I2                  ... STREAM NO. OF COOLING WATER FOR CONDENSER
C     I3                  ... STREAM NO. OF CONDENSED WATER
C     I4                  ... STREAM NO. OF POWER OUTPUT
C     SNEN(IE(IEQ,3)+5)   ... TURBINE OUTLET PRESSURE (IN HG)
C     SNEN(IE(IEQ,3)+6)   ... OUTLET TEMPERATURE OF CONDENSED WATER
C     SNEN(IE(IEQ,3)+7)   ... OUTLET TEMPERATURE OF COOLING WATER
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      J1=IE(IEQ,[illegible])*10000
      J2=L*1000000
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+   J1+J2
      ICOL(N)=I1+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=-1.
```

```
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+   J1+J2
      ICOL(N)=I4+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      P2=SNEN(IE(IEQ,3)+5)/29.92
      IF(IUNIT.EQ.2) P2=P2*14.70
      CALL ENTS(2,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),P2,E2,
     1          SNEN(IS(I3,2)+2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)=-(E1-E2)*EFFTC(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1         SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1+J2
      ICOL(N)=I3+J2
      RDATA(N)=(E2-SNEN(IE(IEQ,3)+6))/(SNEN(IE(IEQ,3)+7)-SNEN(IS(I2,2)+2
     1        ))
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+J2
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE14(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR EXTRUCTIVE CONDENSING TURBINE (TURBC1)
C     I1                    ... STREAM NO. OF INLET STEAM
C     I2                    ... STREAM NO. OF COOLING WATER FOR CONDENSER
C     I3                    ... STREAM NO. OF EXTRACTED STEAM
C     I4                    ... STREAM NO. OF CONDENSED STEAM
C     I5                    ... STREAM NO. OF POWER OUTPUT
C     SNEN(IE(IEQ,3)+5)     ... TURBINE OUTLET PRESSURE (IN HG)
C     SNEN(IE(IEQ,3)+6)     ... OUTLET TEMPERATURE OF CONDENSED WATER
C     SNEN(IE(IEQ,3)+7)     ... OUTLET TEMPERATURE OF COOLING WATER
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      I5=IE(IEQ,10)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I3+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=1.
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ+1
```

```
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I3,2)+3),E2,
     1          T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1          SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      P2=SNEN(IE(IEQ,3)+5)/29.92
      IF(IUNIT.EQ.2) P2=P2*14.70
      CALL ENTS(2,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),P2,E2,SNEN(IS(I4,2)
     1          +2))
      CALL ENTH(3,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),E1)
      RDATA(N)= (E1-E2)*EFFTC(SNEN(IE(IEQ,3)+2),SNEN(IS(I4,2)+2),
     1          SNEN(IS(I4,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I5+J2
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      RDATA(N)=(E2-SNEN(IE(IEQ,3)+6))/(SNEN(IE(IEQ,3)+7)-SNEN(IS(I2,2)+2
     1          ))
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+J2
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE16(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR CONDENSER FOR CONDENSING TURBINE (SCOND)
C     MODELS FOR POLYSAR"S EXPANSION STUDY
C     I1                        ... HOT STREAM TO CONDENSER
C     I2                        ... COOLING WATER TO CONDENSER
C     I3                        ... HOT STREAM OUT
C     I4                        ... COOLING WATER OUT
C     SNEN(IS(I1,2)+2)          ... TEMPERATURE OF HOT STREAM
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
C     MASS BALANCE
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=-1.
```

```
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=-1.
C     HEAT BALANCE
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,1)+L+2000000
      RDATA(N)=-SNEN(IE(IEQ,3)+4)*((SNEN(IS(I1,2)+2)-SNEN(IS(I4,2)+2))-(
     1SNEN(IS(I3,2)+2)-SNEN(IS(I2,2)+2)))/
     1ALOG((SNEN(IS(I1,2)+2)-SNEN(IS(I4,2)+2))/(SNEN(IS(I3,2)+2)-SNEN(
     1IS(I2,2)+2)))/(SNEN(IS(I4,2)+2)-SNEN(IS(I2,2)+2))/1000.
```

```
M=M+1
MEQ=MEQ.1
N=N+1
IROW(N)=M+J1
ICOL(N)=I4+J2
CALL ENTH(1,SNEN(IS(I4,2)+2),SNEN(IS(I4,2)+3),E)
RDATA(N)=E
NEQ(MEQ)=IROW(N)
N=N+1
IROW(N)=NEQ(MEQ)
ICOL(N)=I2+J2
CALL ENTH(1,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),E)
RDATA(N)=-E
N=N+1
IROW(N)=NEQ(MEQ)
ICOL(N)=I1+J2
E=SNEN(IS(I1,2)+4)
RDATA(N)=-E
N=N+1
IROW(N)=NEQ(MEQ)
ICOL(N)=I3+J2
CALL ENTH(1,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),E)
RDATA(N)=E
RETURN
END
```

```
      SUBROUTINE TYPE17(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR BOILER (BOILR)
C     I1                     ... BOILER FEED WATER
C     I2                     ... POWER FOR DRAFT FAN
C     I3                     ... FUEL REQUIRED
C     I4                     ... STEAM GENERATED
C     I5                     ... BLOW DOWN WATER
C     SNEN(IE(IEQ,3)+4)      ... BLOW DOWN RATIO
C     SNEN(IE(IEQ,3)+5)      ... HEATING VALUE OF FUEL
C     SNEN(IE(IEQ,3)+6)      ... POWER FOR DRAFT FAN / B.F.W
C     T                      ... BOILING POINT AT BOILER PRESSURE
C     EFFB(FUNCTION)         ... BOILER EFFICIENCY
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      I5=IE(IEQ,10)
C     MASS BALANCE
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=SNEN(IE(IEQ,3)+4)-1.
```

```
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I5+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-SNEN(IE(IEQ,3)+4)
C     ENERGY BALANCE
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      CALL ENTH(3,SNEN(IS(I4,2)+2),SNEN(IS(I4,2)+3),E)
      RDATA(N)=-E
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I5+J2
      CALL TBP(SNEN(    IS(I4,2)+3),T)
      CALL ENTH(1,T                ,SNEN(IS(I4,2)+3),E)
      RDATA(N)=-E
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      CALL ENTH(1,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E)
      RDATA(N)= E
```

```
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)= EFFB(SNEN(IS(I3,2)+1),SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3))
     1          *SNEN(IE(IEQ,3)+5)
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-SNEN(IE(IEQ,3)+6)
      RETURN
      END
```

```
      SUBROUTINE TYPE17(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR BOILER-1 (BOILR) ... FOR POLYSAR"S EXPANSION STUDY
C     I1                        ... BOILER FEED WATER
C     I2                        ... POWER FOR FORCED DRAFT FAN
C     I3                        ... POWER FOR INDUCED DRAFT FAN
C     I4                        ... FUEL REQUIRED
C     I5                        ... STEAM GENERATED
C     I6                        ... BLOW DOWN WATER
C     SNEN(IE(IEQ,3)+4)         ... BLOW DOWN RATIO
C     SNEN(IE(IEQ,3)+5)         ... HEATING VALUE OF FUEL
C     SNEN(IE(IEQ,3)+6)         ... POWER FOR FORCED DRAFT FAN /STEAM GENERATED
C     SNEN(IE(IEQ,3)+7)         ... POWER FOR INDUCED DRAFT FAN / STEAM GENERATED
C     T                         ... BOILING POINT AT BOILER PRESSURE
C     EFFB(FUNCTION)            ... BOILER EFFICIENCY
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      I5=IE(IEQ,10)
      I6=IE(IEQ,11)
C     MASS BALANCE
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I5+J2
      RDATA(N)=1.
```

```
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=SNEN(IE(IEQ,3)+4)-1.
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I6+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-SNEN(IE(IEQ,3)+4)
C     ENERGY BALANCE
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I5+J2
      CALL ENTH(3,SNEN(IS(I5,2)+2),SNEN(IS(I5,2)+3),E)
      RDATA(N)=-E
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I6+J2
      CALL TBP(SNEN(    IS(I5,2)+3),T)
      CALL ENTH(1,T                ,SNEN(IS(I5,2)+3),E)
      RDATA(N)=-E
```

```
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      CALL ENTH(1,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E)
      RDATA(N)= E
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)= EFFB(SNEN(IS(I3,2)+1),SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3))
     I          *SNEN(IE(IEQ,3)+5)
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I5+J2
      RDATA(N)=-SNEN(IE(IEQ,3)+6)
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I3+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I5+J2
      RDATA(N)=-SNEN(IE(IEQ,3)+7)
      RETURN
      END
```

```
      SUBROUTINE TYPE18(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR EXTRUCTIVE BACK TURBINE (ETURB1)
C     I1                    ... STREAM NO. OF INLET STEAM
C     I2                    ... STREAM NO. OF EXTRACTED STEAM
C     I3                    ... STREAM NO. OF EXHAUSTED STEAM
C     I4                    ... STREAM NO. OF POWER OUTPUT
C     SNEN(IS(I1,2)+2)      ... TEMPERATURE OF INLET STEAM
C     SNEN(IS(I1,2)+3)      ... PRESSURE    OF INLET STEAM
C     EFFTB(FUNCTION)       ... TURBINE EFFICIENCY
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNFN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=1.
```

```
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1          T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1        SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      CALL ENTS(3,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),SNEN(IS(I3,2)+3),E2,
     1          T)
      CALL ENTH(3,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),E1)
      RDATA(N)= (E1-E2)*EFFTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1        SNEN(IS(I3,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE21(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR TUBO-GENERATOR (GTURBB)
C     I1                    ... STREAM NO. OF INLET STEAM
C     I2                    ... STREAM NO. OF OUTLET STEAM
C     I3                    ... STREAM NO. OF OUTPUT POWER
C     SNEN(IE(IEQ,3)+4)     ... POWER CONSUMED INTERNALLY / POWER GENERATED
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      J2=L*1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
```

```
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1          SNEN(IS(I2,2)+2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1         SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=-1./(1.-SNEN(IE(IEQ,3)+4))
      RETURN
      END
```

```
      SUBROUTINE TYPE22(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR TURBO-GENERATOR (GTURB1)
C     I1                   ... STREAM NO. OF INLET STEAM
C     I2                   ... STREAM NO. OF EXTRACTED STEAM
C     I3                   ... STREAM NO. OF EXHAUSTED STEAM
C     I4                   ... STREAM NO. OF POWER OUTPUT
C     SNEN(IE(IEQ,3)+4)    ... POWER CONSUMED INTERNALLY / POWER GENERATED
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=1.
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
```

```
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I2,2)+3),E2,
     1        T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1        SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      CALL ENTS(3,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),SNEN(IS(I3,2)+3),E2,
     1        T)
      CALL ENTH(3,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I3,2)+2),
     1        SNEN(IS(I3,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=-1./(1.-SNEN(IE(IEQ,3)+4))
      RETURN
      END
```

```
      SUBROUTINE TYPE24(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR CONDENSING TURBO-GENERATOR (GTURBC)
C     I1                 ... STREAM NO. OF INLET STEAM
C     I2                 ... STREAM NO. OF COOLING WATER FOR CONDENSER
C     I3                 ... STREAM NO. OF CONDENSED STEAM
C     I4                 ... STREAM NO. OF POWER OUTPUT
C     SNEN(IE(IEQ,3)+5)  ... TURBINE OUTLET PRESSURE (IN HG)
C     SNEN(IE(IEQ,3)+6)  ... OUTLET TEMPERATURE OF CONDENSED WATER
C     SNEN(IE(IEQ,3)+7)  ... OUTLET TEMPERATURE OF COOLING WATER
C     SNEN(IE(IEQ,3)+4)  ... POWER CONSUMED INTERNALLY / POWER GENERATED
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I3+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
```

```
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      IF(IUNIT.EQ.2) P2=P2*14.70
      P2=SNEN(IE(IEQ,3)+5)/29.92
      CALL ENTS(2,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),P2,E2,SNEN(IS(I3,2)
     1          +2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1         SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=-1./(1.-SNEN(IE(IEQ,3)+4))
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I3+J2
      RDATA(N)=(E2-SNEN(IE(IEQ,3)+6))/(SNEN(IE(IEQ,3)+7)-SNEN(IS(I2,2)+2
     1         ))
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+J2
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE24(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR CONDENSING TURBO-GENERATOR (GTURBC)
C     MODELS FOR POLYSAR"S EXPANSION STUDY
C     I1                   ... STREAM NO. OF INLET STEAM
C     I2                   ... STREAM NO. OF CONDENSED STEAM
C     I3                   ... STREAM NO. OF POWER OUTPUT
C     SNEN(IS(I1,2)+2)     ... PRESSURE OF OUTLET STEAM (IN HG)
C     SNEN(IE(IEQ,3)+4)    ... POWER CONSUMED INTERNALLY
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
```

```
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      P2=SNEN(IS(I2,2)+3)/29.92
      IF(IUNIT.EQ.2) P2=P2*14.70
      CALL ENTS(2,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),P2,E2,SNEN(IS(I2,2)
     1          +2))
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1      SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=-1.
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)= 1+L+6000000
      RDATA(N)=SNEN(IE(IEQ,3)+4)
      SNEN(IS(I2,2)+4)=E2
      RETURN
      END
```

```
      SUBROUTINE TYPE25(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR CONDENSING TURBO-GENERATOR (GTURC1)
C
C     I1                     ... STREAM NO. OF INLET STEAM
C     I2                     ... STREAM NO. OF COOLING WATER FOR CONDENSER
C     I3                     ... STREAM NO. OF EXTRACTED STEAM
C     I4                     ... STREAM NO. OF CONDENSED STEAM
C     I5                     ... STREAM NO. OF POWER OUTPUT
C     SNEN(IE(IEQ,3)+5)      ... TURBINE OUTLET PRESSURE (IN HG)
C     SNEN(IE(IEQ,3)+6)      ... OUTLET TEMPERATURE OF CONDENSED WATER
C     SNEN(IE(IEQ,3)+7)      ... OUTLET TEMPERATURE OF COOLING WATER
C     SNEN(IE(IEQ,3)+4)      ... POWER CONSUMED INTERNALLY / POWER GENERATED
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNFN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
      I5=IE(IEQ,10)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I3+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=1.
```

```
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      CALL ENTS(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),SNEN(IS(I3,2)+3),E2,
     1        T)
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I1,2)+2),
     1        SNEN(IS(I1,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      P2=SNEN(IE(IEQ,3)+5)/29.92
      IF(IUNIT.EQ.2) P2=P2*14.70
      CALL ENTS(2,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),P2,E2,SNEN(IS(I4,2)
     1        +2))
      CALL ENTH(3,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),E1)
      RDATA(N)= (E1-E2)*EFGTB(SNEN(IE(IEQ,3)+2),SNEN(IS(I4,2)+2),
     1        SNEN(IS(I4,2)+3))*0.293
      IF(IUNIT.EQ.1) RDATA(N)=RDATA(N)*3.968
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I5+J2
      RDATA(N)=-1./(1.-SNEN(IE(IEQ,3)+4))
      M=M+1
      MEQ=MEQ.1
```

```
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      RDATA(N)=(E2-SNEN(IE(IEQ,3)+6))/(SNEN(IE(IEQ,3)+7)-SNEN(IS(I2,2)+2
     1          ))
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+J2
      RDATA(N)=-1.
      RETURN
      END
      SUBROUTINE TYPE27(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR EQUAL
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      RDATA(N)=-1.
      RETURN
      END
```

```
      SUBROUTINE TYPE26(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR DRIVER SELECTION (SELECT)
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /OBJ5/JEQ,IEX(1)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ,ISEQ,KK ,K ,IPRIOD
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      J=IE(IEQ,4)
      J1=IE(IEQ,1)*10000
      L1=L+1000000
      K=1
30    ID=SNEN(IE(IEQ,3)+K)
      IF(ID.EQ.0) RETURN
CC    ENERGY BALANCE FOR UNIT(SELECT)
      M=M+1
      MEQ=MEQ+1
      K0=K*10000
      K1=(K-1)*10000
      N=N+1
      IROW(N)=M+L1+J1
      NEQ(MEQ)=IROW(N)
      NEO=NEQ(MEQ)
      I1=IE(IEQ,4)+6
      ICOL(N)=IE(IEQ,I1)+L1+K0
      RDATA(N)=-1.
C
      IF(ID.   GT.60) GO TO 24
C     STREAM IDENTIFICATION FOR INTERNAL POWER DEMAND
      L0=(L1+ID*10000)/100
      DO 22 I=1,N
      J5=IROW(I)/100
      IF(L0.EQ.J5) GO TO 23
22    CONTINUE
23    J2=IE(IEQ,I1)+L1
      IF(ICOL(I).EQ.J2) ICOL(I)=ICOL(1)+K0
      I=I+1
```

```
      J5=IROW(I)/100
      IF(L0.EQ.J5) GO TO 23
      GO TO 21
C     SETTING UP EXTERNAL POWER DEMAND
24    DD=SNEN (IE(IEQ,3)+K)
      CALL DEMAND(L,IE(IEQ,I1)+K0,DD)
C
21    DO 10 I=1,J
      N=N+1
      IROW(N)=NEO
      I1=IE(IEQ,I+5)
      ICOL(N)=I1+L1+K0
      RDATA(N)=1.
C     FINDING THE UNIT(I2) WHICH HAS STREAM I1 AS ONE OF OUTPUT STREAM
      DO 33 I2=1,ISEQ
      J2=IE(I2,5)
      DO 33 I3=1,J2
      J3=IE(I2,4)+I3+5
      IF(I1.EQ.IE(I2,J3)) GO TO 34
33    CONTINUE
C     DEMAND-CAPACITY RELATION FOR UNIT(I2)  INVOLVED IN DRIVER SELECTION
34    IF(IPRIOD.EQ.1) GO TO 39
      IF(K.EQ.1) GO TO 38
      M1=M1+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M1+3000000+IE(I2,1)*10000+L
      ID=SNEN(IE(I2,3)+3)
      ICOL(N)=ID+L1+K0
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(I2,1)+4000000+K0+100
      RDATA(N)=-1.
39    IF(K.NE.1) GO TO 32
```

```
      DO 31 I4=1,JEQ
      IF(IEX(I4).NE.IE(I2,1)) GO TO 31
      IEX(I4)=IE(I2,1)+K0
      GO TO 37
31    CONTINUE
32    JEQ=JEQ+1
      IEX(JEQ)=IE(I2,1)+K0
      GO TO 37
38    L0=(3000000+L+IE(I2,1)*10000)/100
      DO 25 I4=1,N
      J5=IROW(I4)/100
      IF(L0.EQ.J5) GO TO 26
25    CONTINUE
      GO TO 37
26    ICOL(I4)=ICOL(I4)+10000
      I4=I4+1
      J5=IROW(I4)/100
      IF(L0.EQ.J5) GO TO 26
C
C     IDENTIFICATION OF MODELS      FOR UNIT(I2) INVOLVED IN DRIVER SELECTIOI
37    L0=(L1+IE(I2,1)*10000)/100
      DO 50 I4=1,N
      J5=IROW(I4)/100
      IF(L0.EQ.J5) GO TO 55
50    CONTINUE
55    II=0
51    IF(K.EQ.1) GO TO 52
      IF(II.EQ.IROW(I4)) GO TO 54
      M=M+1
      MEQ=MEQ+1
      II=IROW(I4)
54    N=N+1
      ICOL(N)=ICOL(I4)+K1
      IROW(N)=M+L1+I2*10000
      RDATA(N)=RDATA(I4)
      NEQ(MEQ)=IROW(N)
      GO TO 53
52    ICOL(I4)=ICOL(I4)+10000
```

```
53    I4=I4+1
      J5=IROW(I4)/100
      IF(LO.EQ.J5) GO TO 51
C     FINDING THE UNIT (I4) WHICH HAS AS OUTPUT STREAM STREAMS WHICH
C     ARE INPUT STREAMSS FOR UNIT(I2) INVOLVED IN DRIVER SELECTION
36    J3=IE(I2,4)
      DO 40 I3=1,J3
      I5=IE(I2,I3+5)
      DO 43 I4=1,ISEQ
      J4=IE(I4,5)
      DO 43 I6=1,J4
      J5=IE(I4,4)+I6+5
      IF(I5.EQ.IE(I4,J5)) GO TO 35
43    CONTINUF
35    LO=(L1+IE(I4,1)*10000)/100
C     IDENTIFICATION OF INPUT STREAMS FOR UNIT(I2) INVOLVED IN DRIVER SELECTI
      DO 42 I4=1,N
      J5=IROW(I4)/100
      IF(LO.EQ.J5) GO TO 41
42    CONTINUE
41    IF(K.EQ.1) GO TO 62
      J2=I5+L1+10000
      IF(ICOL(I4).NE.J2) GO TO 44
      N=N+1
      ICOL(N)=ICOL(I4)+K1
      IROW(N)=IROW(I4)
      RDATA(N)=RDATA(I4)
      GO TO 44
62    J2=I5+L1
      IF(ICOL(I4).NE.J2) GO TO 44
      ICOL(I4)=ICOL(I4)+10000
44    I4=I4+1
      J5=IROW(I4)/100
      IF(LO.EQ.J5) GO TO 41
40    CONTINUE
```

```
C     FINDING THE UNIT (I4) WHICH HAS AS INPUT STREAMS OUTPUT STREAMS
C     FOR UNIT INVOLVED IN DRIVER SELECTION
      J3=IE(I2,5)-1
      IF(J3.EQ.0) GO TO 10
      DO 70 I3=1,J3
      I5=IE(I2,4)+I3+5
      DO 73 I4=1,ISEQ
      J4=IE(I4,4)
      DO 73 I6=1,J4
      J5=I6+5
      IF(IE(I2,I5).EQ.IE(I4,J5)) GO TO 75
73    CONTINUE
75    L0=(L1+IE(I4,1)*10000)/100
C     IDENTIFICATION OF OUTPUT STREAMS FOR UNIT(I2) INVOLVED IN DRIVER SELECT
      DO 72 I4=1,N
      J5=IROW(I4)/100
      IF(L0.EQ.J5) GO TO 71
72    CONTINUE
71    IF(K.EQ.1) GO TO 76
      J2=IE(I2,I5)+L1+10000
      IF(ICOL(I4).NE.J2) GO TO 77
      N=N+1
      ICOL(N)=ICOL(I4)+K1
      IROW(N)=IROW(I4)
      RDATA(N)=RDATA(I4)
      GO TO 77
76    J2=IE(I2,I5)+L1
      IF(ICOL(I4).NE.J2) GO TO 77
      ICOL(I4)=ICOL(I4)+10000
77    I4=I4+1
      J5=IROW(I4)/100
      IF(L0.EQ.J5) GO TO 71
70    CONTINUE
10    CONTINUE
      K=K+1
      IF(K.LE.IE(IEQ,2)) GO TO 30
      RETURN
      END
```

```
      SUBROUTINE TYPE28(L,IS,IE,NST,NEN,NIS,NIE)
C     PROGRAM FOR SOFTNERS (WTREAT)
C     I1                    ... MAKE-UP FEED WATER
C     I2                    ... STEAM CONSUMED FOR WASHING
C     I3                    ... TREATED WATER (OUTPUT STREAM)
C     SNEN(IE(IEQ,3)+4)     ... RATIO OF STEAM CONSUMPTION TO FEED
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      J1=IE(IEQ,1)*10000
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=I1+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+L+1000000
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+J1
      ICOL(N)=I2+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+L+1000000
      RDATA(N)=-SNEN(IE(IEQ,3)+4)
      RETURN
      END
```

```
      SUBROUTINE TYPE29(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR MOTOR'(MOTOR)
C     EFM                      ... MOTOR EFFICIENCY
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /SYS1/SNEN(1)
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      DATA EFM/0.95/
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+L+1000000+IE(IEQ,1)*10000
      ICOL(N)=IE(IEQ,7)+L+1000000
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,6)+L+1000000
      RDATA(N)=-EFM
      RETURN
      END
```

```
      SUBROUTINE TYPE30(L,IS,IE,NST,NEN,NIS,NIE)
C     SUBPROGRAM FOR HEATER (HEATR)
C     I1                     ... INLET STEAM FOR HEATING
C     I2                     ... BOILER FEED WATER
C     I3                     ... OUTPUT STREAM 1(CONDENSED WATER)
C     I4                     ... OUTPUT STREAM 2(B.F.W. HEATED)
C     SNEN(IE(IEQ,3)+4) ... HEAT TRANSFER COEFFICIENT
C     SNEN(IS(I1,2)+2)  ... TEMPERATURE OF INLET STEAM
C
      DIMENSION IS(NST,NIS),IE(NEN,NIE)
      COMMON /GENRL/NCP,M,N,IEQ,MAXST,IUNIT,M1,M2,MEQ
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      COMMON /SYS1/SNEN(1)
      I1=IE(IEQ,6)
      I2=IE(IEQ,7)
      I3=IE(IEQ,8)
      I4=IE(IEQ,9)
C     MASS BALANCE
      J2=L+1000000
      J1=IE(IEQ,1)*10000+J2
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I2+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I4+J2
      RDATA(N)=-1.
C     HEAT BALANCE
      M=M+1
      MEQ=MEQ+1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      RDATA(N)=1.
```

```
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=IE(IEQ,1)+L+2000000
      RDATA(N)=-SNEN(IE(IEQ,3)+4)*((SNEN(IS(I1,2)+2)-SNEN(IS(I4,2)+2))-(
     1SNEN(IS(I3,2)+2)-SNEN(IS(I2,2)+2)))/
     1ALOG((SNEN(IS(I1,2)+2)-SNEN(IS(I4,2)+2))/(SNEN(IS(I3,2)+2)-SNEN(
     1IS(I2,2)+2)))/(SNEN(IS(I4,2)+2)-SNEN(IS(I2,2)+2))/1000.
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I1+J2
      RDATA(N)=1.
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      RDATA(N)=-1.
      M=M+1
      MEQ=MEQ.1
      N=N+1
      IROW(N)=M+J1
      ICOL(N)=I4+J2
      CALL ENTH(1,SNEN(IS(I4,2)+2),SNEN(IS(I4,2)+3),E)
      RDATA(N)=E
      NEQ(MEQ)=IROW(N)
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I2+J2
      CALL ENTH(1,SNEN(IS(I2,2)+2),SNEN(IS(I2,2)+3),E)
      RDATA(N)=-E
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I1+J2
      CALL ENTH(3,SNEN(IS(I1,2)+2),SNEN(IS(I1,2)+3),E)
      RDATA(N)=-E
```

```
      N=N+1
      IROW(N)=NEQ(MEQ)
      ICOL(N)=I3+J2
      CALL ENTH(1,SNEN(IS(I3,2)+2),SNEN(IS(I3,2)+3),E)
      RDATA(N)=E
1000  FORMAT(1H ,F15.5)
      RETURN
      END
      SUBROUTINE LPSORT(K,L,ICODE)
C     SORTING FOR LP DATA
      COMMON /GENRL/NCP,M,N
      COMMON /LP1/IROW(1)/LP2/ICOL(1)/LP3/RDATA(1)/LP4/NEQ(1)
      DO 600 J=1,L
      J1=J*100
      I1=J1+ICODE
      I2=K
      DO 610 J2=I2,N
      IF(I1.NE.ICOL(J2)) GO TO 610
      IF(J2.NE.K) GO TO 611
      K=K+1
      GO TO 610
611   IR=IROW(K)
      IC=ICOL(K)
      RD=RDATA(K)
      IROW(K)=IROW(J2)
      ICOL(K)=ICOL(J2)
      RDATA(K)=RDATA(J2)
      IROW(J2)=IR
      ICOL(J2)=IC
      RDATA(J2)=RD
      K=K+1
610   CONTINUE
600   CONTINUE
      RETURN
      END
```

```
      FUNCTION EFFTB(F,TEMP,PRESS)
C     EFFICIENCY FOR BACK PRESSURE TURBINE (SYNTHESIS STUDY)
      EFFTB=0.5
      RETURN
      END
      FUNCTION EFGTB(F,T,P)
C     EFFICIENCY FOR GENERATOR TURBINE (SYNTHESIS STUDY)
      EFGTB=0.62
      RETURN
      END
```

REFERENCES

AIEE Committee Report, "Application of Probability Method to Generating Capacity Problem", AIEE, Power Apparatus and Systems, 1165 (1961).

Arnstein, R. and O'Connell, L., "What's the Optimal Heat Cycle for Process Utilities", Hydrocarbon Processing, 47 No. 6., 88 (1968).

Baldwin, C.J., Gaven, D.P., and Hoffman, C.H., "Mathematical Model for Use in the Simulation of Power Generation Outage I- Fundamental Condition", AIEE Transactions, Power Apparatus and Systems, December 1251 (1959).

Baldwin, C.J., Billings, J.E., Gaven, D.P., and Hoffman, C.H., "Mathematical Model for Use in the Simulation of Power Generation Outage II - Power System Forced Outage Distribution" ibid, 1258, (1959).

Baldwin, C.J., Gaven, D.P., Hoffman, C.H., and Rose, J.A. "Mathematical Model for Use in The Simulation of Power Generation Outage III - Model for a Large Interconnection" ibid, February 1645 (1960).

Baldwin, C.J., Desalvo, C.A., and Hoffman, C.H. "Load and Capacity Models for Generation Planning by Simulation" ibid, 359 (1960).

Baldwin, C.J., Desalvo, C.A., and Limmer, H.D. "The Effect of Unit Size, Reliability and System Service Quality in Planning Generation Expansion", ibid, 1042 (1961).

Bending, M.J., and Hutchison, H.P. "The Calculation of Steady State Incompressible Flow in Large Networks of Pipes", Ch.E.Sci., 28, 1857 (1973).

Booth, R.R., "Optimal General Planning Considering Unvertainty", AIEE Transactions, Power Apparatus and Systems 91, 70 (1972).

Bouilloud, Ph. "Compute Steam Balance by LP" Hydrocarbon Processing, August, 127, (1969).

Box, M.J., "New Method of Constrained Optimization and a Comparison with Other Methods", Computer Journal 8 No. 1, 42, (1965).

Brennan, M.K., Galloway, C.D., and Kirchmayer, L.K., "Digital Computer Aid Economics - Probabilistic Study of Generation System", AIEE Transactions, Power Apparatus and Systems, 564 (1958).

Cohen, A., and Jensen, L.E., "Digital Program for the Economic Selection of Generating Capacity Additions:, AIEE Transactions, Power Apparatus and Systems, 1628 (1958).

Coleman, J.R., and York, R., "Optimal Plant Design for Growing Market", Ind. & Eng. Chem., 56. No. 1, 29 (1964).

Dain, R.J., and Whitlock, D. "Total Energy wystem Design" Hydrocarbon Processing, July 175, (1969).

Dain, R.J., and Whitlock, D. "The Optimization of Total Energy Systems", British Chem. Eng., 14, No. 9, 477 (1969).

Dale, K.M., Ferguson, W.H., Hoffman C.G., and Rose, J.E. "Production Cost Calculation for System Planning by Operational Gaming Models", AIEE Transactions, Power Apparatus and Systems, 1746 (1960).

Dilliard, J.K., and Sels, H.K. "An Introduction to the Study of System Planning by Operational Gaming Models", ibid, 1284 (1959).

Fitzpatrick, R.J., and Gallagher, J.W. "Determination of an Optimized Generation Expansion Pattern", ibid 1052 (1962).

Fleming, J.B., Lambric, J.R. and Smith M.R. "Energy Conservation in New-Plant Design". Chem. Eng., January 112 (1974).

Fugill, A.P., "Principles and Practice of System Planning", AIEE, Power Apparatus and Systems, 74, 1323 (1955).

Generoso, Jr.E., and Hitchcock, L.B., "Optimizing Plant Expansion - Two Cases", Ind & Eng. Chem. 60, No. 1,12 (1968).

Goto, S., and Matsubara, M., "Optimization of an Extractive Stirred Tank Reactor Coupled with Separators", J.Chem.Eng. (Japan), 5 90 (1972).

Happel, J., "Chemical Process Economics", John Wiley & Sons, New York, (1958)

Harbert, W.D. "Which Tower Goes Where?", Petrol Refiner 36 169 (1957).

Hatakeyama, J. and Symazu, A. "Heat Balance Calculation of Steam Network by Connection Matrix", Netzu Kanri, 22 No. 11, 1219 (1971).

Hendry, J.E., and Hughes, R.R. "Generating Separation Process Flowsheets", Chem.Eng. Progr., 68 No.6, 69 (1972).

Hendry, J.E., Rudd, D.F., and Seader, J.D. "Synthesis in the Design of Chemical Processes", A.I.Ch.E. Journal 19, No. 1, 1, (1973).

Hiraizumi, Y., and Nishimura, H., "Optimization of Heat Exchanger System", Chemical Engineering (Japan), No. 11, 30, (1966).

Hiraizumi, Y., Mori, A., and Nishimura H. "Analysis and Calculation of Process Matrix by Use of Logical Matrix", Kagaku Kogaku (Japan), 33, No. 3, 85, (1969).

Hwa, C.S., "Mathematical Formulation and Optimization of Heat Exchanger Network Using Separable Programming" Proc. Symposium No. 4, p.101, A.I.Ch.E. - I. Chem.E. Joint Meeting (1965).

IBM "Mathematical Programming System/360 Application Description", (1968).

Ichikawa, A., and Fan, L.T., "Necessary Condition for Optimal Process Structure and Evolutionary Search for Optimal Structure", paper presented at Am. Inst. Chem. Engrs. Meeting, Dallas, Texas (1972).

Jackson, R.E., Klomparens, A.J., and Westbrook, G.T. "Long Range Planning" Chem.Enh.Progr. 61, No. 1, 83, (1965).

Jeynes, P.H., and Van Mimwegen, L. "The Criterion of Economic Choice" AIEE. Power Apparatus and Systems. 606, (1958).

Johnson, A.I., and Peters, N. "Energy Systems Modelling", Annual Meeting of the Engineering Institute of Canada, October, 1975.

Kern, D.Q. "Process Heat Transfer", McGraw Hill (1957).

Kesler, M.G., and Parker, R.O. "Optimal Network of Heat Exchange", Chem. Eng. Progr. Symp. Series - Heat Transfer 65, No. 92, 111 (1968).

Kikkawa, Y., and Shoji, Y. "Optimal Design of Utility Systems", Kagaku Sochi, Japan, June, (1968).

King, C.J., "Separation Processes", McGraw-Hill, New York (1971).

King, C.J., Gantz, D.W., and Barnes, F.J. "Systematic Evolutionary Process Synthesis", Ind.Eng.Chem.Process Design Develop., 11, 271 (1972)

Kirchmayer, L.K., et.cl. "An Investigation of the Economic Size of Steam-Electric Generating Units".AIEE Power Apparatus and Systems, 74 600, (1955).

Kist, C., and Thomas, G.J. "Probability Calculation for System Generation Reserves", AIEE, Power Apparatus and Systems, 515 (1958).

Kobayashi, S., Umeda, T., and Ichikawa, A. "Synthesis of Optimal Heat Exchange Systems - An Approach by the Optimal Assignment Problem in Linear Programming", Chem. Eng. Sci. 26 1367 (1971).

Kowalik, J., and Osborne M.R. "Methods for Unconstrained Optimization Problems". American Elsevier Publishing Company Inc., New York, (1968).

Lasdon, L.S. "Optimization Theory for Large Systems", McMillan, New York, (1970).

Lee, K.F., Masso, A.G., and Rudd, D.F. "Branch and Bound Synthesis of Integrated Process Designs" Ind. Eng. Chem. Fundamentals, 9 48 (1970).

Liapis, A.I., Walter H.D., and Zentrum, E.T.H. "The Use of Concepts of Thermodynamic Efficiency in the Synthesis of Heat Exchanger Networks", presented at 4th Annual Research Meeting of the Institution of Chemical Engineers, England (1977).

Lockhart, F.J. "Multi-Column Distillation of Natural Gasoline", Petrol Refiner, 26,104 (1947).

Lopez, J. "An Optimization Package for Chemical Processes", Master Thesis, The University of Western Ontario, London, Ontario, Canada (1975).

Mahalex, C., Nath, R., and Motard, R.L. "Structuring Process Flowsheets" presented at 4th Annual Research Meeting of the Institution of Chemical Engineers, England (1977).

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

McGalliard, R.L., and Westerberg, A.W. "Structural Sensitivity Analysis in Design Synthesis", paper presented at Am. Inst. Chem.Engrs. Meeting, Dallas, Texas (1972).

Menzies, M.A. "Sarnia Energy 2000" Progress Report at The University of Western Ontario, London, Ontario, Canada (1972).

Menzies, M.A. and Johnson, A.I. "Synthesis of Optimal Energy Recovery Networks Using Discrete Methods", Can. J. Chem. Eng., 50,290 (1972)

Miller, R.Jr., "Process Energy Systems", Chem. Eng. May, 130 (1968).

Nelder, J.A., and Mead, R. "A Simplex Method for Minimization", Computer Journal, 1, No. 4, 308 (1965).

Nishida, N., Kobayashi, S., and Ichikawa, A. "Optimal Synthesis of Heat Exchanger Systems". Chem.Eng.Sci., 26, 1841 (1971).

Nishida, N., Liu, Y.A., and Ichikawa, A. "Studies in Chemical Process Design and Synthesis : II", A.I.Ch.E. J., 22 539 (1976).

Nishida, N., Liu, Y.A., and Lapidus, L. "Studies in Chemical Process Design and Synthesis: III", A.I.Ch.E.J., 23, 77 (1977).

Nishimura, H., and Hiraizumi, Y. "Optimal System Pattern for Multicomponent Distillation Systems", Intern. Chem. Eng., 11, 188 (1971).

Nishio, M. "An Analysis on Automatic Solving Techniques of Process Networks", Progress Report No. 6 at The University of Western Ontario, London, Ontario, Canada (1974).

Nishio, M. "Computer Aided Synthesis of Total Energy Systems (9)", Progress Report No. 13 at The University of Western Ontario, London, Ontario, Canada, (1975).

Nishio, M. "An Automatic Solving Technique of Arbitrary Heat Exchange Networks", Progress Report No. 17 at The University of Western Ontario, London, Ontario, Canada (1975)a.

Oatman, E.N., and Hamaut, L.J. "A Dynamic Approach to Generation Expansion Planning", AIEE, Power Apparatus and Systems, 92; 1888, (1973).

Onishi, Y., and Kikkawa, Y. "An Optimal Design of Complex Heat Exchange Systems", Kagaku Sochi, Japan, No. 5,21, (1968).

Petlyuk, F.B., Platonov, V.M.,and Slavinskii, D.M. "Thermodynamically Optimal Method for Separating Multicomponent Mixtures", Int.Chem.Eng., 5 555 (1965).

Ponton, J.W., and Donaldson, R.A.B. "A Fast Method For The Synthesis of Optimal Heath Exchanger Networks". Chem.Eng. Sci., 29 No. 12 2375 (1974).

Powers, G.J. "Heuristic Synthesis in Process Development", Chem. Eng. Prog., 68, No. 8,88 (1972).

Rathore, R.N.S., and Powers, G.J. "A Forward Branching Scheme for the Synthesis of Energy Recovery Systems", Ind. Eng. Process Design Develop. 14 No. 12, 175 (1975).

Rathore, R.N.S., Vanwormer, K.A. and Powers G.J. "Synthesis of Distillation Systems with Energy Integration", A.I.Ch.E.J. 20, No. 5,940 (1974).

Reps, D.N., and Rose, J.A. "Strategy for Expansion of Utility Generation" AIEE, Power Apparatus and Systems, 1710 (1968).

Rod, V., and Marek, J. "Separation Sequences in Multicomponent Rectification", Coll. Czech. Chem. Comm. 24, 3240 (1959).

Rudd, D.F. "The Synthesis of System Designs, I : Elementary Decomposition Theory", A.I.Ch.E.J. 14, 343 (1968).

Schroeder, T.W., and Wilson, G.P. "Economic Selection of Generating Capacity Additions", AIEE, Power Apparatus and Systems, 1133, (1958).

Siirola, J.J., Power, G.J., and Rudd, D.F. "Synthesis of System Designs III : Toward a Process Concept Generator" A.I.Ch.E.J., 17, 677 (1971).

Siirola, J.J., and Rudd, D.F. "Computer Aided Synthesis of Chemical Process Designs", Ind. Eng. Chem. Fundamentals, 10, 353, (1971).

Slack, J.B. "Steam Balance " A New Exact Method", Hydrocarbon Processing, March, 154, (1969).

Slack, J. "Energy Systems in Large Process Plants", Chem. Eng. January, 107 (1972).

Taylor, R.M.H. "Total Energy for Cost Reduction" British Chem. Eng. 14 No. 7, 358 (1969).

Thompson, R.W., and King, C.J. "Systematic Synthesis of Separation Schemes" Paper presented at Am. Inst. Chem. Engrs. Meeting, Dallas, Texas (1972).

Umeda, T. and Nishio, M. "Comparison Between Sequential and Simultaneous Approaches in Process Simulation", Ind. Eng. Chem. Process Design Develop., 11, 153 (1972).

Umeda, T. "Studies of the Optimal design of Chemical Processing Systems", Ph.D. Thesis, Tokyo Institute of Technology, Tokyo, Japan, (1972).

Umeda, T., Hirai, A., and Ichikawa A. "Synthesis of Optimal Processing System by an Intergrated Approach", Chem. Eng. Sc., 27 795 (1972).

Umeda, T., Shindo, A., and Ichikawa A. "Process Synthesis by Task Assignment", Chem. Eng. Sc., 29 2033 (1974).

Umeda, T., and Ichikawa, A. "A Rational Approach to Process Synthesis". Chem. Eng. Sc., 30,699 (1975).

University of Western Ontario, Computer Center "University of Toronto 370/165 Documentation", (1972).

Wells, G.L., and Hodgkinson, M.G. "Heuristics for Energy Savings: presented at 4th Annual Research Meeting of the Institution of Chemical Engineers, England (1977).

Westerberg, A.W., and Stephanopoulos, G. "Studies in Process Synthesis I. Branch and Bound Strategy with List Techniques for the Synthesis of Separation Schemes", Chem. Eng. Sci., 30, No. 8, 963, (1975).

Yamazaki, H. "On Determining the Capacity of Power Plant as an Example of Application of Linear Programming", Kagaku Sochi, Japan, June, 1970.

Yamazaki, H. "Optimal Supply of Electric Power and Steam in Private Power Plant (No.2) - An Example of Application of Separable Programming and Mixed Integer Programming", Netza Kanri, Japan, 22 No. 10, 10 (1970).