267

THE DESIGN OF FLEXIBLE HEAT EXCHANGER NETWORKS

by

Alan Clive Beautyman, B.Sc.

A thesis submitted for the degree of
Doctor of Philosophy of the University of London
and for the Diploma of Imperial College.

Department of Chemical Engineering and Chemical Technology,
Imperial College of Science and Technology,
London SW7.

September, 1984.

ABSTRACT

A method is presented for the design of flexible heat exchanger networks, that is networks which are able to handle a specified bounded range of operating conditions in an economical way. Flexibility is an economically important and, at present, relatively neglected aspect of network design.

The design method developed in the present work is based on computer modelling and optimization techniques and is an improvement on existing methods. A number of pinched and unpinched network design problems are used to demonstrate the new method. The method has the advantage of guaranteeing operation of the designed flexible network at all operating points, not just extreme points of the specified operating range.

In some cases it has been found that the desired flexibility may be imparted by the quantitative overdesign of certain exchangers whereas in other cases the network structure has been found to be deficient. Methods for detecting and dealing with these two cases are given.

It is demonstrated that the correct placement of bypasses for control purposes in a network is crucial in ensuring flexible operation and good control characteristics. Where possible rules are given to aid correct placement.

The design method is extended to include cases in which the uncertain variables are non-independent thus ensuring that a network is not required to handle impossible (or highly unlikely) operating conditions. The economic benefits of such an approach are demonstrated.

Techniques are also presented for the design of flexible networks which have design flexibility, that is have more than the minimum number of units or have stream-splits.

As a result of the main work a new method has been developed for correcting the standard log-mean temperature difference formulation for use in the computer models.

ACKNOWLEDGEMENTS

I am indebted to the Science and Engineering Research Council for financial support during the course of this work and also to British Petroleum International plc who funded the research programme of which this work was a part.

I would also like to thank Alan Cornish, my supervisor, for his help over the past three years.

TABLE OF CONTENTS

		PAGE NO.
ABSTRACT		2
NOMENCLATURE		14
UNITS		15
1.	INTRODUCTION	16
2.	DESIGNING FOR UNCERTAINTY	22
	2.1 Introduction	22
	2.2 General Methods for Design Under Uncertainty	23
	2.3 Review of the Work of Grossmann and Co-Workers	31
	2.4 Heat Exchanger Network Synthesis	35
	2.5 Review of the Work of Morari and Co-Workers	37
	2.6 Requirements of a Design Scheme for Flexible Heat Exchanger Networks	40
3.	DEGREES OF FREEDOM ANALYSIS	41
	3.1 Introduction	41
	3.2 An Example Problem	41
	3.3 Interpretation of Degrees of Freedom	44
	3.4 General Calculation of Degrees of Freedom	44
	3.5 Badly Posed Problems	46
	3.5.1 Streams not connected via stream/stream heat exchangers to a utility exchanger	46
	3.5.2 Networks featuring an overspecified exchanger	49
	3.6 Applications of Degrees of Freedom Analysis	53
	3.6.1 Unpinched problems with minimum number of units	53
	3.6.2 Pinched problems	55
	3.7 Conclusions from Degrees of Freedom Analysis	56

PAGE NO.

4. DEVELOPMENT OF A NEW DESIGN METHOD FOR FLEXIBLE HEAT EXCHANGER NETWORKS 63

4.1 Introduction 63
4.2 Physical Constraints 63
4.3 Imposed Constraints 65
4.4 Incorporating the Feasibility Constraint into a Design Procedure 67

5. COMPUTER IMPLEMENTATION OF THE DESIGN SCHEME 73

5.1 Introduction 73
5.2 Network Modelling 73
5.2.1 Network representation 73
5.2.2 First stage of analysis 74
5.2.3 Equation solving 76
5.2.3.1 Choice of equations to solve by iterative methods 76
5.2.3.2 Solution procedure for the general case 81
5.2.3.3 Equation solving routine used 82
5.2.4 Modelling of infeasible operation 83
5.3 Constraints 85
5.3.1 Modelling optimizing control 85
5.4 Use of the Program for Cornerpoint Checking 86
5.5 Maximization of Constraint Functions 87

6. INVESTIGATION OF THE DESIGN PROBLEM 94

6.1 Introduction 94
6.2 Infeasible Operation 94
6.2.1 Sizing infeasibilities 94
6.2.2 Structural infeasibilities 98
6.2.3 Unbypassed exchangers and stream-splits 99
6.2.4 Summary 100

		PAGE NO.
6.3	Form of Constraint Functions	101
	6.3.1 Example Problem 6.1	101
	6.3.2 Example Problem 6.2	102
	6.3.3 Example Problem 6.3	104
	6.3.3.1 Morari and co-workers' treatment of the problem	104
	6.3.3.2 A new approach	105
6.4	Utility Use Constraint	106
	6.4.1 Example Problem 6.4	107
6.5	Summary	110
7.	DESIGN EXAMPLES	131
7.1	An Unpinched Problem	131
7.2	A Combined Network	147
7.3	Pinched Problem 1	155
7.4	Pinched Problem 2	157
7.5	Pinched Problem 3	161
7.6	Pinched Problem 4	167
7.7	Conclusions from Design Examples	172
8.	ADDITIONAL TOPICS	174
8.1	Non-Independent Inlet Variations	174
	8.1.1 Example problem	176
8.2	Networks which have Design Flexibility	182
	8.2.1 Example problem	182
	8.2.1.1 Formulation of problem	184
	8.2.2 General design method for a network with design flexibility	187
8.3	Large Operating Ranges	200

		PAGE NO.
8.4	Additions and Improvements to the Design Method	203
	8.4.1 Additional uncertain variables	203
	8.4.2 Heat exchanger models	204
	8.4.3 Physical properties	205
	8.4.4 A short-cut design method	205
9.	CONCLUSIONS	207
APPENDIX A	Redundant Utility Exchangers	210
APPENDIX B	Log-Mean Temperature Difference Formulation	214
APPENDIX C	Use of Program HERON for Cornerpoint Checking	221
APPENDIX D	Use of Program HERON1 for Constraint Maximization	231
APPENDIX E	A Reprint of the Paper "The Design of Flexible Heat Exchanger Networks"	236
REFERENCES		256
APPENDIX F	Microfiche Listing of HERON	
APPENDIX G	Microfiche Listing of HERON1	

LIST OF TABLES

TABLE		PAGE NO.
4.1	Interpretation of the general nomenclature for the specific treatment of heat exchanger networks used in the present work.	70
7.1	Area and cost of networks designed for the range 0.9 to 1.0 of nominal F.Cp.	137
7.2	The cost of the networks designed for the five operating ranges considered.	138
7.3	Stream data used in a combined network example.	150
7.4	Comparison of the utility use of networks (a) and (b).	164
8.1	Cost of networks under consideration.	179
8.2	The optimum U.A1 and U.A2 values for three sets of operating conditions.	189
8.3	A comparison of the costs of a network using a stream-split with the best network not using a stream-split.	190

LIST OF FIGURES

FIGURE		PAGE NO.
3.1	A network used as an example in the degrees of freedom analysis.	59
3.2	A network used to demonstrate a badly posed problem.	59
3.3	A solution given by Linnhoff to problem 6SP1.	59
3.4	Network topology of Figure 3.3.	60
3.5	A network featuring an overspecified exchanger.	60
3.6	A network which requires optimizing control.	60
3.7	Exchanger 3 from the network given in Figure 3.6.	61
3.8	An unpinched minimum number of units network.	61
3.9	A pinched problem.	62
4.1	A network used to identify physical constraints.	71
4.2	An outline design scheme for the design of flexible networks.	72
5.1	The process matrix representation of a simple network.	90
5.2	The usual representation of the network given in Figure 5.1.	91
5.3	Use of the program for cornerpoint checking.	92
5.4	Use of the program for maximizing each network constraint function value.	93
6.1	The describing equations for an exchanger/bypass system.	112
6.2	The result of using different exchanger areas in an exchanger/bypass system required to meet a fixed duty.	113
6.3	Variation of required exchanger split fraction with U.A for an exchanger/bypass system required to meet a fixed duty.	114
6.4	Variation of required exchanger split fraction with an inlet variable for an exchanger/bypass system in a network.	114

FIGURE		PAGE NO.
6.5	An exchanger/bypass system showing structural infeasibilities.	115
6.6	Distinguishing between sizing and structural infeasibilities.	116
6.7	A network solving problem 6SP1 used in Example Problem 6.1.	117
6.8	The variation of required exchanger split fraction 4 with F.Cp1 and T1 in Example Problem 6.1.	118
6.9	The variation of required exchanger split fraction 2 with F.Cp1 and T1 in Example Problem 6.1.	119
6.10	The variation of required exchanger split fraction 2 or 4 with F.Cp1 and T1 in Example Problem 6.1.	120
6.11	A network given by Marselle(1980) used in Example Problem 6.2.	121
6.12	Variation of required exchanger split fraction with T1 for U.A = 3.98 in Example Problem 6.2.	122
6.13	Variation of required exchanger split fraction with T1 for U.A = 8.04 in Example Problem 6.2.	123
6.14	A network posing flexibility problems used in Example Problem 6.3.	124-125
6.15	The variation of the cold end ΔT of exchanger 1 with F.Cp1 in Example Problem 6.3.	126
6.16	The variation of the hot end ΔT of exchanger 1 with F.Cp1 in Example Problem 6.3.	127
6.17	A network for the pinched problem used in Example Problem 6.4.	128
6.18	The variation of total utility use with F.Cp1 in Example Problem 6.4.	129
6.19	Actual total utility use as a percentage greater than theoretical total minimum use in Example Problem 6.4.	130
7.1	Five networks solving problem 5SP1.	139
7.2	Performance of the five networks at point I in the operating range.	140
7.3	Performance of the remaining networks at other operating conditions.	141
7.4	Networks able to handle a range of F.Cp values from 0.9 to 1.0 times nominal.	142

FIGURE		PAGE NO.
7.5	Networks able to handle a range of F.Cp values from 0.95 to 1.0 times nominal.	143
7.6	Infeasible operation of networks for a range of inlet temperatures from 0.9 to 1.0 times nominal.	144
7.7	Networks able to handle a range of inlet temperatures from 0.95 to 1.0 times nominal.	145
7.8	Infeasible operating conditions and networks able to handle F.Cp values from 0.95 to 1.0 times nominal and inlet temperatures from 0.95 to 1.0 times nominal.	146
7.9	The performance of networks A and B at operating points I and II.	151
7.10	A combined network including networks A and B as special cases.	152
7.11	The eight alternative networks formed by combining networks A and B and assigning three bypasses.	153
7.12	The most promising network alternative.	154
7.13	Performance of the optimum network at operating points I and II.	154
7.14	Pinched Problem 1.	156
7.15	The network used in Pinched Problem 2.	159
7.16	The performance of three alternatives at operating point I.	160
7.17	The network used in Pinched Problem 3.	165
7.18	The performance of three alternatives at operating point I.	166
7.19	The network used in Pinched Problem 4.	169
7.20	The performance of the four alternatives at operating point I.	170
7.21	The performance of the four alternatives at operating point I'.	171
8.1	Operating ranges for two uncertain inlet variables.	180
8.2	The cheapest network of those considered in Chapter 7 for problem 5SP1 able to handle independently varying F.Cp values from 0.95 to 1.0 times nominal.	180
8.3	The reduced operating range.	181

FIGURE		PAGE NO.
8.4	The minimum required exchanger areas at operating points I and II.	191
8.5	The location of critical points I and II in the operating range as represented in Figure 8.3.	192
8.6	A possible network including a stream-split for problem 5SP1 with a range of F.Cp values from 0.95 to 1.0 times nominal.	193
8.7	Three designs for nominal conditions.	194
8.8	The performance of the optimum nominal case design at operating point I.	195
8.9	The optimum design for operating point I.	195
8.10	The performance of the optimum network designed for operating point I at operating point II.	196
8.11	The optimum design for operating point II.	196
8.12	The performance of the optimum network designed for operating point II at operating point I.	197
8.13	The performance of the network at three sets of operating conditions for variable U.A1 and U.A2 values.	198
8.14	The performance of the optimum network at three sets of operating conditions.	199
8.15	A network designed for a large operating range.	202
A.1	A network demonstrating redundant utility exchangers.	213
B.1	A comparison of the standard formulation of the log-mean temperature difference with Grossmann's formulation.	219
B.2	A new formulation for the calculation of the log-mean temperature difference.	220
C.1	The network used in the example and the input data required by the program.	224
D.1	The network used in the example and the input data required by the program.	233

NOMENCLATURE

The network representation is as used by Linnhoff and co-workers with the following nomenclature:

TM = Temperature of stream section M

FN = Flowrate of stream N

CpN = Heat capacity of stream N

HE = Heat load of heat exchanger E

UE = Overall heat transfer coefficient for exchanger E

AE = Area of heat exchanger E

CK = Heat load of cooling utility exchanger K

WJ = Heat load of heating utility exchanger J

R = Exchanger split fraction which, for a bypass, is the fraction of the total flowrate passing through the exchanger.

In the work the following variables occur as products of single variables:

F.CpN = The product of the flowrate (F) of stream N and the heat capacity (Cp) of stream N (the heat capacity flowrate)

U.AE = The product of the overall heat transfer coefficient (U) for exchanger E and the area(A)of exchanger E.

UNITS

QUANTITY	UNIT
T	^{o}F
F.Cp	(BTU/hr.^{o}F)x10^{-4}
U.A	(BTU/hr.^{o}F)x10^{-4}
H,C,W	(BTU/hr)x10^{-4}
R	Dimensionless

These units are consistent with those used by previous workers, and in particular with those used in specifying the widely used literature problems, listed by Linnhoff(1979), some of which are used in the present work.

C H A P T E R 1

INTRODUCTION

The methods used to design chemical plants are based, in general, on only a single set of operating conditions, and the assumption is made that these design conditions will be those encountered during operation. This is a good assumption in many cases as during normal operation operating conditions, such as throughput, may be adjusted to meet design specifications. Inevitably, however, certain operating conditions will not always take their design values due to uncertainties in process stream or physical properties. Many such uncertainties will take the form of small variations about the design (or nominal) values which will still be those which are most likely to occur in practice. To ensure that a plant designed for a single set of nominal operating conditions can operate over this range of conditions which occurs in practice various 'rule of thumb' techniques have been developed based on the oversizing of equipment in the hope that it will then be able to handle the range of operating conditions. This, then, was the state of the art for design under uncertain conditions until recent years: an accurate and often complex procedure to design the optimum plant for a single set of nominal conditions followed by an arbitrary overdesign stage in which equipment sizes were increased according to experience. In terms of the accuracy of the two stages there is an obvious dichotomy. However, it is probable that this design method was generally successful in producing a plant which could handle uncertainties of the form described above. There are two main reasons why such a design scheme is no longer attractive, one is economic and the other practical.

Plants are now required to be much more economical in both their capital and operating costs. 'Rule of thumb' oversizing was successful because it provided excess capacity to that which was necessary and hence oversized equipment was being bought but not fully used. A requirement of a new method for design under uncertainty would be to provide just sufficient overdesign to handle the possible range of

uncertain conditions. Stricter economic constraints also now govern the operation of plants. Unfavourable economic conditions may mean that a plant must be run at much less than maximum throughput or, in the case of a crude oil distillation train, that a different feedstock must be used with different physical properties. Thus in this case there is no single set of nominal or most likely operating conditions and a design method must be able to accommodate a large and possibly discontinuous range of operating conditions.

The practical objection to 'rule of thumb' oversizing is that computer based techniques are now available for plant design which can be adapted to design accurately for uncertainty quickly and cheaply.

The stricter economic climate has also led to the development of effective methods for the design of heat exchanger networks which can recover waste heat in a process by transferring heat between hot process streams which need cooling and cold process streams which need heating. The design methods developed have proved successful in producing networks which recover in an economical manner a high proportion of the available waste heat in a process for a single set of operating conditions. The present work is concerned with the design of flexible heat exchanger networks; that is networks which can handle a specified bounded range of operating conditions in an economical way.

In Chapter 2 of this thesis a review of work covering the general problem of design under uncertainty is given followed by a brief review of the standard heat exchanger network synthesis problem and then a review of work relating to the flexible heat exchanger network design problem. The results of two groups of workers are highlighted, those of Grossmann and co-workers relating to general design under uncertainty and those of Morari and co-workers relating to the design of flexible heat exchanger networks. On the basis of this literature review a number of requirements of a new design scheme for flexible heat exchanger networks are identified which could lead to an improvement on existing methods.

One of the requirements is that bypasses for control purposes should be placed correctly in a network, not necessarily following the assumption of previous workers that all exchangers have controlled bypasses. In Chapter 3 a degrees of freedom analysis is presented which yields information about how many bypasses need be included and identifies cases in which incorrect placement of bypasses makes adequate control impossible. A particular problem arises in a network which requires only cooling utilities for some parts of a specified operating range and only heating for others. The extension to the degrees of freedom analysis required in this case is presented in Appendix A.

In Chapter 4 the mathematical framework for a new design method for flexible heat exchanger networks is given. The method is based on the mathematical development of Grossmann and co-workers and aims to design networks which can operate over a given bounded range of operating conditions set by the designer, comprising a range of process stream flowrates, heat capacities and inlet temperatures defined only by lower and upper bounds on the variables. For any possible set of such operating conditions the network must be able to bring the process streams to their required target temperatures. The method proceeds by searching for a set of operating conditions which the network under consideration cannot handle. Network constraint functions, which must take feasible values if the network is to operate adequately, can be optimized (maximized) with respect to the uncertain inlet variables to find operating conditions for which the operation of the network is 'most' infeasible. The network can then be modified to eliminate the infeasibility and the search for other operating conditions producing infeasible operation continued. Thus, in a stepwise procedure, a flexible network is produced. Incorporated in the network model is a constraint function which ensures that a designed network recovers a high proportion of the recoverable heat for each set of operating conditions. This goal of 'high' heat recovery is the same as that used by Linnhoff and co-workers in a 'non-flexible' design method and is used for the same purpose : to ensure an economically good network rather than a economically 'optimum' network which is not necessarily the best alternative.

In Chapter 5 details are given of the computer implementation of part of this design method. The major component of the computer program is a network modeller which will simulate the performance of any given network of heat exchangers, utility exchangers,splitters and mixers. The network modelling program is attached to an optimization routine which carries out the constraint function maximization required by the search procedure. Chapter 5 contains details of the operation of the program including network modelling, equation solving, modelling of optimizing control and constraint function maximization. Two versions of the program exist, HERON for checking the operation of a network at all the extreme points of a given operating range and HERON1 which searches for infeasible operation by the maximization of constraint functions. Examples of the use of these programs are given in Appendices C and D respectively and listings of HERON and HERON1 are included on microfiche as Appendices F and G.

One particular problem in modelling the infeasible operation of a network is to ensure that the log-mean temperature difference relationship, used for exchanger modelling, will return values for any values of temperature approach. A new method for correcting the log-mean temperature difference function to overcome this problem is presented in Appendix B.

Using the computer program a number of fundamental features of the design problem are investigated in Chapter 6. In particular, it is important to know whether a network cannot handle a particular set of operating conditions because of sizing or structural deficiencies, and the work in this chapter resolves this problem. A number of example problems are used to study the form of the constraint functions governing particular networks to determine whether ensuring feasible operation of a network at the extreme points of an operating range guarantees operation at intermediate operating points (it does not). Also considered in Chapter 6 are appropriate formulations of a utility use constraint function.

In Chapter 7 a number of flexible design examples are presented covering both unpinched and pinched problems. The removal of structural infeasibilities by using a combined network is also demonstrated. From these examples a number of important conclusions are drawn relating to the flexible design problem.

Chapter 8 covers a number of additional topics and extensions to the new design method. Firstly, it is important that the required operating range is specified accurately to prevent a network being designed to handle impossible (or highly unlikely) operating conditions. By specifying interactions between uncertain inlet variables a more realistic reduced operating range may be specified and an extension to the basic design scheme to handle this case is presented. An example design problem demonstrates the economic benefits which can result from this type of approach.

A particular problem arises when dealing with networks which have design flexibility (that is those which have more than the minimum number of units or have stream-splits) and a method is presented for correctly sizing the exchangers in such networks to ensure flexibility.

Various topics are not completely covered or resolved in the thesis. In particular the problems of dealing with large operating ranges (which may requre a network to have optimizing or switching control), adapting the design method to include additional uncertain variables (such as target temperatures and heat transfer coefficients) and speeding up the design method by using a short-cut technique are highlighted .

Finally in Chapter 9 the main conclusions resulting from the work are summarised.

Appendix E contains a copy of the paper "The Design of Flexible Heat Exchanger Networks" presented at the First UK National Heat Transfer Conference held in Leeds, July 1984, the proceedings of which

appear in I.Chem.E. Symposium Series No. 86. This paper is based on a summary of part of the work reported in this thesis.

C H A P T E R 2

DESIGNING FOR UNCERTAINTY

2.1 Introduction

In recent years rising energy costs have forced plant designers to attempt to make plants more economical by considering the recovery of waste heat. Heat exchanger networks which transfer heat between hot process streams which need cooling and cold process streams which need heating provide a means of increasing heat recovery and reducing utility use.

Methods for the design of heat exchanger networks for fixed operating conditions which attempt to strike an economic balance between the operating cost of a network, represented by utility use, and the capital cost of the exchangers required are given by Linnhoff (1979), Linnhoff and Flower (1978), Flower and Linnhoff (1980), Linnhoff and Hindmarsh (1983) and the I. Chem. E. User Guide (1982). These design methods have proved successful in practice and the I. Chem. E. User Guide (1982) lists cases in which more economical designs have resulted from their application.

It is important, however, that heat recovery is not achieved at the expense of the operability and controllability of the network. The present work is concerned with the design of flexible heat exchanger networks, that is networks which can handle a range of operating conditions in an economical way.

General descriptions of the types of uncertain variables which can produce a range of operating conditions are given by Rudd and Watson (1968), Weisman and Holzman (1972a) and White (1982). For example, variable conditions may be produced by changes in process stream flowrates or temperatures or in heat transfer coefficients. For heat exchanger networks the most important uncertain variables are the inlet process stream flowrates, heat capacities and temperatures. These uncertainties may arise from feedstock or product changes in the

overall process or from the need to operate at less than maximum throughput, and because of this the range of possible inlet conditions may be large. It is likely that only upper and lower limits for the uncertain variables will be known as they do not represent small variations about nominal or most likely values, for which a probability distribution function would apply, but also potentially large and long lasting step changes in variables. It is also possible that an uncertain variable may take values which fall into discontinuous ranges, for example this may be the case in a crude oil preheat train if the crude is changed to one with a different specific heat capacity.

In developing a new method for the design of heat exchanger networks able to handle a range of operating conditions it is first necessary to consider general methods for the design of systems under uncertainty.

2.2 General Methods for Design Under Uncertainty

In the past provision was made for uncertain variables in the design of a chemical plant by arbitrary 'rule of thumb' oversizing of equipment. For example, Kittrell and Watson (1966) note that it was common practice to simply multiply the calculated equipment size by a factor of 1.1 to 2.0 with the choice influenced by the experience of the designer in similar cases and by intuition. The limitations of this approach are that nothing can be guaranteed about the resulting design; too much oversizing leads to unnecessary excess capability and capital cost: too little oversizing may lead to inadequate operation and an inability to meet product specifications with a subsequent economic penalty.

An early method for the accurate quantitative design of a system to handle uncertain conditions economically is given by Dantzig (1955) (1956), (1963) with particular reference to transportation and allocation problems. Common practice had been to put plenty of 'slack' and 'fat' into the system to ensure that capabilities were above requirements and Dantzig notes that this could lead to far from optimal

action. The design method proposed by Dantzig is formulated as a two-stage linear programming problem, the general form of which was followed by many subsequent workers and is described in its particular application to chemical plant design by, for example, Malik and Hughes (1973).

Malik and Hughes (1973) recognise the design of a chemical process as being inherently a two-stage problem: a design stage in which plant structure and equipment sizes are chosen, and an operating stage in which operating conditions are adjusted to yield optimal performance. The objective of the design stage is to fix a set of design parameters which will optimize future capability for optimized operation.

A method for determining a 'safety' or oversizing factor for a system which is more accurate than arbitrary oversizing is the confidence level approach described by Buckley (1950) and in a more complex formulation applicable to the general linear programming problem by Charnes and Cooper (1959). In this method the designer attempts to strike a compromise between excess cost and flexible operation by setting a particular level of reliability which the system must attain. For example a chemical plant might be required to operate satisfactorally for 95% of the time. This method requires that the probability distribution functions of the unknown variables be known. Often, as in the work of Charnes and Cooper, normal distribution are assumed with the means being nominal operating conditions and standard deviations being estimated. The confidence level approach is similar in philosophy to the usual treatment of failure rates and repair time as described by Smith (1981).

In an application of this method to heat exchangers Berryman and Himmelblau (1971) apply a 90% confidence level to the detailed design of single exchangers. Al-Zakri and Bell (1981a), (1981b) use this approach for the prediction of an oversizing factor for each heat exchanger in a crude preheat train for which all inlet flowrates, temperatures and heat capacities are uncertain, as are the heat transfer coefficients. The confidence level set is that all outlet temperatures must be within specified ranges for 95% of the time.

The major problem in this approach is in setting an appropriate confidence level as no direct economic data are considered. Lashmet and Szczepanski (1974) apply the method to distillation column overdesign as does Saletan (1969) who observes that plausible economic variations can lead to economic confidence levels of 50% to 90% for the same problem indicating that a more accurate approach to the overdesign of complex plant is needed than the imposition of a single predetermined confidence level figure. Freeman and Gaddy (1975) conclude that in some cases estimating an appropriate confidence level may be possible but in others an attempt to find an economically optimum confidence level should be made. Methods for determining cost-optimum oversizing factors form the bulk of the work which has been carried out on design under uncertain conditions for chemical plants.

The first stage of such a procedure is to produce an economic function representing the value of the plant for any particular set of uncertain variables. This will include an assessment of the capital cost of the plant, allowing for depreciation over its expected life, which is dependent only on the design parameters chosen, and also the operating cost which will generally be dependent on both the uncertain and design variables. The operating cost can include the cost of feedstocks and utilities, the value of any products and a cost penalty for those sets of operating conditions producing inadequate operation. The cost penalty due to inadequate operation may not simply represent loss of revenue from products but also may include an assessed cost for loss of good will and similar factors. For examples of how to construct such plant value functions see Saletan and Caselli (1963) and Weisman and Holzman (1972a),(1972b).

As there are generally assumed to be an infinite number of sets of conditions defined by the uncertain variables, with each set representing a potentially different plant value, a method of calculating the expected value of the plant over all the uncertain variables is needed. Analytic integration over the range of uncertain variables to find the expected value of the plant, as proposed by

Kittrell and Watson (1966) and Schlosser (1979), may be possible if the probability distribution functions of the uncertain variables are known (for example, normal, triangular or rectangular distributions have been used) but for large systems the complex interactions between variables make analytic integration impracticable.

Most methods use a stochastic approach such as that presented by Berryman and Himmelblau (1973) based on Monte Carlo methods as described by Tayyabkhan and Richardson (1965) and Hess and Quigley (1963). Using this method a series of random sets of uncertain variables, conforming to the required probability distribution functions, are generated and the performance of the plant modelled for each set of variables as in the work of Gaddy and Culberson (1973). The value of the plant can be calculated for each set of conditions and if sufficient sets are used a good numerical approximation to the expected value of the plant can be calculated.

The most rigorous economic treatment of design for uncertain conditions is termed the Here-and-Now Problem by Malik and Hughes (1979) or the two-stage programming problem by Avriel and Wilde (1969). This treatment is based on the work of Mangasarian (1964) and Mangasarian and Rosen (1964) which applies to the general nonlinear programming problem. In the Here-and-Now problem the expected value of the plant is maximized with respect to the design variables. This is found to be computationally very expensive as Monte Carlo methods must be used to calculate the expected value of the plant for each set of design variables tried by the maximization procedure. However, some workers have considered the direct solution of this problem, for example Saletan and Caselli (1963), Kittrell and Watson (1966) and Saletan (1969).

In order to reduce the computational effort involved a number of simpler economic objectives have been developed and these are summarized by Malik and Hughes (1979), Watanabe and Nishimura (1973) and Avriel and Wilde (1969).

The most commonly used economic formulations are:

1) The Most-Likely Problem
   The plant is designed to have a maximum value (calculated from the plant value function) assuming that all uncertain variables take their most likely (nominal) values i.e. no uncertainty is taken account of.
2) The Expectation Problem.
   The expected value of the plant designed in the Most-Likely Problem is calculated for the given probability distributions of uncertain variables and allowing for operational optimization.
3) The Wait-and-See Problem.
   The expected value of the plant value function is calculated for the given probability distributions of uncertain variables but assuming that for each set of uncertain variables the plant value is maximized with respect to operational and design variables. This is a hypothetical calculation as in practice the design variables are invariant.

Work by Mangasarian (1964), Mangasarian and Rosen (1964) and Madansky (1959) has shown that for many systems the expected plant values defined by these economic formulations obey the following relationship:

$$\text{Wait-and-See} > \text{Here-and-Now} > \text{Expectation}$$

As the Wait-and-See and Expectation expected plant values are easier to calculate than the Here-and-Now expected plant value, design methods based on one or a combination of these formulations and the confidence level approach have been proposed by Malik and Hughes (1979), Doering and Gaddy (1979) and Avriel and Wilde (1969). The strategies proposed often require experience and intuition to combine the objectives correctly but close approximations to the Here-and-Now design can be produced.

The theoretical Wait-and-See objective function, used to calculate the maximum expected plant value, representing as it does the maximum possible value of any design for each set of uncertain variables, has been used to define economic objectives for suboptimal designs. Formulations which are used by, for example, Wen and Chang (1968) and Watanabe and Nishimura (1973) include:

1) Minimize (with respect to the design variables) the maximum deviation of the plant value from the maximum plant value given by the Wait-and-See objective function. This deviation is termed the sensitivity and, if normalized, takes account of the magnitude of the values.

2) Minimize expected normalized (or relative) sensitivity

A Simpler objective which does not require the calculation of Wait-and-See function values is:

3) Minimize the maximum deviation of the plant value from the value of the plant designed in the Most-Likely Problem.

Strategies (1) and (2) are described by Rohrer and Sobral (1965). A from of strategy (1) which uses absolute rather than normalized deviations is termed the minimax regret strategy by Savage (1954).

A different definition of sensitivity is used by Chen, Erickson and Fan (1970) and Seinfeld and McBride (1970). In their work the uncertain variables which strongly affect the profitability of a given designed plant are identified by calculating the derivative of the plant value function with respect to each uncertain variable. A design is sought which restricts the magnitude of these derivatives to acceptable values.

Derivatives are also used in the sensitivity analyses of Takamatsu, Hashimoto and Ohno (1970), Dittmar and Hartmann (1976), Takamatsu, Hashimoto and Shioya (1973) and Bingzhen, Yourum, Xiaorong and Pingkai (1982). In their work an optimized nominal case design forms the basis of a sensitivity analysis in which the outputs from the system must be kept within given bounds. Partial derivatives are used

to identify which uncertain variables strongly affect the outputs. Takamatsu, Hashimoto and Ohno (1970) apply the method to heat exchanger network design with the objective of minimizing the excess added area required to bring the outlet stream temperatures within given bounds. It is found that different amounts of extra area must be added to each exchanger, indicating that a single overdesign factor would have been inappropriate. It can be seen that the objective of keeping outlet temperatures within given bounds is the same as that used by Al-Zakri and Bell (1981a), (1981b) in a confidence level approach, and that the design methods, which are both based on computer simulation of a plant designed for nominal operating conditions, are similar.

There are a number of reasons why many of the design methods described so far may not be applied directly to the particular flexible heat exchanger network design problem addressed in the present work. No accurate probability distributions for the uncertain variables can be set as they are subject not only to small scale variations about nominal or most likely values but also to potentially large, long-lasting and upredictable step changes due to feedstock or throughput changes. The formulation of reliable and accurate plant value functions is also difficult as feedstock and product values can change rapidly (as for example crude oil prices). Flexible networks which have been designed to be cost optimum on the basis of present economic data may prove to be far from optimum cost networks in the future after unexpected economic changes, thus demonstrating the design procedure to produce a cost 'optimum' network to have been needlessly overexact in its setting of an economic target.

However, the major reason why more of the design methods discussed so far is applicable to a flexible heat exchanger network design problem is that they aim to solve the sizing problem but ignore the structural problem. The assumption is made that any desired degree of flexibility may be given to a plant simply by the appropriate oversizing of equipment. This is generally the case for reactors or distillation columns but is not true for heat exchanger networks. As will be demonstrated, a network designed for nominal operating

conditions may not be able to handle a different set of conditions even with infinite oversizing of the individual heat exchangers.

Relatively few workers have tackled the sizing and structural problems for uncertain design. Johns, Marketos and Rippin (1976), (1978) deal with the case where all uncertainties are resolved after the start-up of the plant and allow for the fact that plant modifications may be made then. In the case of heat exchanger networks all uncertainties are not resolved. Powers and Mayer (1975) also recognise that uncertain variables may make it necessary to add or remove complete parts of a process.

For the standard design problem in which no uncertain variables are considered, attempts have been made to optimize for structure in addition to size by the use of structural parameters, see Westerberg and McGalliard (1972) and Heydweiller and Fan (1982). In their work a general heat exchanger network is used which must contain the optimum network as a special case. "Structural Sensitivity Analysis" is used to indicate which exchangers are to be removed. It is unclear whether this method is workable for more than small networks.

This general approach is also proposed by Grossmann (1977) who investigates the optimization of a completely general network to produce a solution in which zero areas and flows define a particular optimal subnetwork. Grossmann concludes that no efficient solution to this nonlinear programming problem is possible due to the problem of multiple local optima. No efficient solution to the equivalent flexible problem which includes uncertain variables using either structural sensitivity or direct optimization is therefore likely to be possible either. Grossmann proposes a compromise in which the structural and sizing problems are treated separately.

Another objective which has been used for design under uncertainty is termed the Permanently Feasible Problem by Malik and Hughes (1979). This design method has the advantage that only lower and upper bounds on the uncertain variables need be known rather than full probability distribution functions. The final design must be feasible for all values of uncertain variables, which is equivalent to setting a

confidence level of 100%. This objective is applied by Friedman and Reklaitis (1975a), (1975b) to the linear programming problem and has occasionally been applied to chemical plant design, for example by Gaddy and Freeman (1974) in the design of a pollution control system where tight design is required by law. In a simplified application of this design target Nishida, Ichikawa and Tazaki (1972), (1974) imply that a design may be made fully feasible by designing for a single 'worst' set of uncertain variables, as do many of the workers who use a form of sensitivity analysis. The most substantial work on this topic is by Grossmann and co-workers who demonstrate that there may be more than one such critical set of conditions and also consider variables which may be manipulated while a plant is running to produce optimized operation, a factor ignored by many workers.

2.3 Review of the Work of Grossmann and Co-Workers

The approach used by Grossmann and co-workers is a development of design methods presented for mechanical and allocation problems by Bandler (1974) and later Kwak and Haug (1976) and Polak and Sangiovanni-Vincentelli (1979).

The basic design strategy as presented by Grossmann (1977), Grossmann and Sargent (1979), Halemane and Grossmann (1981b), (1982) and Halemane (1982a), (1982b) is as follows.

For any chemical plant the following vectors may be defined:

d = equipment design parameters which are fixed values
x = state variables which can be determined by solving the set of equations representing the process system
z = control variables which may be adjusted during operation to meet utility and operating targets
θ = uncertain variables

It is assumed that the uncertain variables are bounded therefore a region D, the disturbance range, is defined which contains all possible sets of values of θ:

$$D = \{\theta \mid \theta^L < \theta < \theta^U\} \qquad (2.1)$$

where θ^L and θ^U are the lower and upper bounds on θ. The equations which govern the operation of a system are of the form:

$$h'(d, z, x, \theta) = 0 \qquad (2.2)$$

To ensure that in practice a system can operate feasibly certain constraints must be satisfied.

If g' is the set of all (J) such constraints then for feasible operation:

$$g'(d, z, x, \theta) < 0 \qquad (2.3)$$

or as $x = x(d,z,\theta)$ the state variables can be eliminated and feasible operation of the system is represented by:

$$g(d, z, \theta) < 0 \qquad (2.4)$$

It is required that the plant must operate feasibly over the entire disturbance range and it has been proved by Polak and Sangiovanni-Vincentelli (1979) and Halemane and Grossmann (1981a), (1981b), (1982) that to ensure this the following must apply:

$$\max_{\theta \in D} \min_{z} \max_{j \in J} g_j(d, z, \theta) < 0 \qquad (2.5)$$

In general terms the 'worst' value of all the individual constraint values which can be found in the disturbance range must be feasible. The control variables z are used to try to maintain feasibility.

In the work of Grossmann and co-workers, an expected cost function is minimized subject to this feasibility constraint producing a problem of the form:

$$\min_{d} \; \max_{\theta \in D} \{ \min_{z} C(d, z, \theta) \mid g(d, z, \theta) \leq 0 \} \qquad (2.6)$$

$$\text{s.t.} \quad \max_{\theta \in D} \; \min_{z} \; \max_{j \in J} \; g_j(d, z, \theta) \leq 0$$

Much of the work of Grossmann (1977), Grossmann and Sargent (1979), Halemane and Grossmann (1981b), (1982), Grossmann, Halemane and Swaney (1982a), (1982b), (1983), Grossmann and Halemane (1982) and Halemane (1982a), (1982b) has been in reformulating and developing solution methods for this problem, which is very difficult to solve directly.

A number of related topics are raised when the uncertain design problem is formulated in this way.

1) The feasibility constraint (2.5) can be used to calculate the 'size' of the feasible region for any particular system and as such can be used as a metric for operational flexibility, see Swaney and Grossmann (1982).

   This flexibility index, which is effectively one number representing the flexibility of a system, can be used to produce designs with the optimal degree of flexibility if the required disturbance range need not be predetermined. The index also can be used to compare and choose between otherwise similar designs.

2) For systems with zero degrees of freedom no optimizing control variables z are present. The feasibility constraint (2.5) then reduces to:

$$\max_{\theta \in D} \; \max_{j \in J} \; g_j(d, \theta) \leq 0 \qquad (2.7)$$

   So that, as shown by Grossmann, Halemane and Swaney (1982a), (1982b), (1983), the constraints may be treated individually:

$$\max_{\theta^j \in D} \; g_j(d, \theta^j) \leq 0 \quad \text{for each } j \in J \qquad (2.8)$$

If additional degrees of freedom are required the procedure is much more difficult to carry out as it is not possible to treat the constraints individually.

3) As noted in most of the work of Grossmann and co-workers and proved by, for example, Halemane and Grossmann (1981b), (1982), if the constraint functions $g_j(d,z,\theta)$ are jointly convex (or montonic) in z and θ then the problem

$$\max_{\theta \in D} \min_{z} \max_{j \in J} g_j(d,z,\theta) \qquad (2.9)$$

has its global and local solutions at extreme points of the polyhedral region $D = \{\theta | \theta^L \leq \theta \leq \theta^U\}$.

It follows that if the constraints are convex or monotonic feasibility of operation for every value of $\theta \in D$ can be guaranteed by considering in the design all the vertices of the polyhedron D. Grossmann and Sargent (1978) and Grossmann (1977) present a method based on analysing the gradient of each constraint function which will select the vertices of D which maximize individual constraint functions, thus reducing the number of critical points which must be considered to guarantee full feasibility (N.B. It is possible that there will be more than one such critical point)

The aim of Grossmann and co-workers is to produce a cost optimum design. As in other methods for uncertain design the method is applicable to sizing equipment but not directly to determining a structure. However, the formulation of the feasibility constraint (2.8) does prove useful in developing a new design scheme.

A number of the design methods discussed have been applied to particular formulations of the flexible heat exchanger network problem, for example by Al-Zakri and Bell (1981a), (1981b), Takamatsu, Hasimoto and Ohno (1970) and Grossmann (1977), but none are generally applicable to the problem as they neglect structural considerations. To review the major

work which does consider the structural problem, that of Morari and co-workers, it is necessary to briefly review standard 'non-flexible' heat exchanger network synthesis methods.

2.4 Heat Exchanger Network Synthesis

The standard non-flexible heat exchanger network synthesis problem is as follows: given a set of hot process streams to be cooled and another set of cold process streams to be heated, with process stream flowrates, heat capacities and inlet and target temperatures specified, plus a set of available heating and cooling utilities, develop the least cost network of exchangers to accomplish the required heating and cooling.

The following example illustrates a typical non-flexible design problem statement:

Stream	F.Cp	T_i	T_t	Type
1	1.2	500	100	Hot
2	1.6	450	150	Hot
3	0.4	100	400	Cold

where T_i = Stream inlet temperature
T_t = Stream target temperature

Many design methods have been proposed to solve this problem. For a review and assessment of the relative merits of most of the available methods see Nishida, Stephanopoulos and Westerberg (1981) and Hanson (1984).

It is clear that the methods which have proved the most successful and widely used to date are those of Linnhoff and co-workers as presented by Linnhoff (1979), Linnhoff and Flower (1978), Flower and Linnhoff (1980), Linnhoff and Hindmarsh (1983) and by Linnhoff et al in the I.Chem.E. User Guide (1982).

Much of the power of this work lies in a preanalysis target-setting stage which is partly based on work by Hohmann (1971).

By setting a minimum allowable approach temperature, ΔT_{min}, the minimum possible utility use required by a network can be calculated directly from the problem data, thus setting a heat recovery target for a designed network to achieve.

The minimum number of exchangers (or, more accurately, process stream matches) needed may also be calculated and is generally one less than the total number of process and utility streams.

It has been found that a network meeting these two targets of minimum utility use and number of matches for any particular problem is likely to be close to the economic optimum network, provided that a good choice of ΔT_{min} is used. ΔT_{min} may be regarded as providing a trade-off between capital and operating costs; a lower ΔT_{min} value leads to lower utility use but larger exchanger areas are required. In general ΔT_{min} is set to a relatively low value as, due to the high cost of energy, networks featuring low utility use are economically attractive.

Also in the preanalysis heat recovery "bottlenecks" or pinch points may be identified which prevent full heat integration. If a stream system is pinched a network must feature both heating and cooling utilities.

In the most recent design method given by Linnhoff and Hindmarsh (1983) and the I.Chem.E. User Guide (1982) the aim is to produce networks which feature minimum utility use (based on ΔT_{min}) achieved by the fewest possible exchangers. If a pinch is present the essential matches and stream splits required immediately around the pinch can be identified. Away from the pinch no fixed rules for matching streams are given, possibly enabling a number of different networks to be generated.

The methods of Linnhoff and co-workers are undoubtedly effective in designing heat exchanger networks for fixed operating conditions. The designed networks may or may not be able to operate over a range of conditions, as this is not an objective considered in the design method. As pointed out by Linnhoff and Hindmarsh (1983) the final choice between a number of possible networks must be made not only on economic grounds but also on the basis of other process objectives, such as flexibility.

A design method which extends the work of Linnhoff and co-workers to the flexible design problem is that of Morari and co-workers.

2.5 Review of the Work of Morari and Co-workers

The basic design strategy described by Marselle (1980), Marselle, Morari and Rudd (1982), Morari, Lenhoff, Marselle and Rudd (1979) and Morari, Marselle and Rudd (1980) is as follows.

The form of the flexible design problem is the same as that used in the present work. The following example illustrates a typical flexible design problem statement:

Stream	F.Cp	T_i	T_t	Type
1	1.2-1.4	500-450	100	Hot
2	1.6-1.9	450-300	150	Hot
3	0.4-0.9	100-110	400	Cold

The F.Cp and T_i values are treated as uncertain variables with only lower and upper bounds specified. E.g. the F.Cp value of stream 1 can take any value between 1.2 and 1.4. Target temperatures are assumed to be fixed.

The combination of the ranges of F.Cp and T_i defines the operating (or disturbance) range, D, of uncertain variables as in the work of

Grossmann and co-workers discussed previously. Morari and co-workers term the vertices or extreme points of the operating range 'cornerpoints'.

If all F.Cp and T_i values are assumed to be variable the total number of cornerpoints is given by:

$$\text{Total No. Cornerpoints} = 4^{(\text{Number of streams})} \qquad (2.10)$$

A heuristic design procedure is developed in which four cornerpoints are selected corresponding to particular sets of inlet conditions which would submit a network to a severe test. They are:

A - Maximum exchanger capacity
B - Maximum cooling required
C - Maximum heating required
D - Maximum area required

Optimal network structures are designed for cases A, B, C and D and 'nominal'operating conditions using the techniques of Linnhoff and Hindmarsh (1983). The networks all must be able to recover maximum heat based on a particular minimum temperature approach, ΔT_{min}, which has been chosen. The five structures found are combined to form a 'combined design' in which all five appear as special cases.

When this topology has been determined Morari, Marselle and Rudd (1980) claim that the optimal sizing of the heat exchanger areas to cope with all inlet perturbations is a "relatively standard nonlinear programming problem" although Grossmann and co-workers have shown that the problem is far from trivial. It is claimed by Morari, Marselle and Rudd (1980) that this heuristic design procedure works well but Saboo and Morari (1984) find that the four selected cornerpoints are insufficient and that a generally valid subset of critical cornerpoints cannot be determined independently of a network structure. Morari, Marselle and Rudd (1980) do state that in some cases the design should be checked for all the cornerpoints.

It can be seen that the work of Grossmann and co-workers has a direct application here. Feasible operation for all the cornerpoints will only guarantee feasiblity for all intermediate conditions if the constraints governing the network are convex or monotonic. The fact that this may not always be the case is demonstrated by Morari (1982b), (1983) and Saboo and Morari (1984) who provide an example of a network which is cornerpoint feasible but is not feasible for intermediate conditions.

Saboo and Morari (1984) identify a class of problems for which cornerpoint feasiblity is sufficient to ensure full feasibility. Two of the necessary conditions are:

1) The operating range consists of inlet temperature variations only.
2) The networks do not feature stream-splits.

These two conditions alone restrict the number of problems for which cornerpoint feasibility can be safely applied.

Morari and co-workers' definition of resilience, which requires that for all sets of uncertain inlet conditions a network must be able to recover the maximum amount of heat based on a fixed value of ΔT_{min}, is not satisfactory as will be discussed and demonstrated in Chapters 4 and 7. The rigid setting of a single heat recovery target also leads to a more serious problem which is the assumption implicit in all their work, and stated explicitly in that part of it which also deals with control structure synthesis, that <u>all</u> exchangers in a network must be fitted with a controlled bypass. As actual exchanger areas are ignored in the structural design procedure the implicit assumption made is that the exchangers have flexible areas. Thus, when the exchanger sizes are set to the maximum required values for the operating range it will follow that for any particular set of inlet conditions one or more of the exchangers may be oversized. In order to ensure that the outlet streams meet the prescribed target temperatures and the particular heat recovery set by ΔT_{min} despite this oversizing, bypasses must be

installed on all exchangers. This may not only produce control problems but also does not allow for advantage to be taken of oversized exchangers to recover the maximum amount of heat possible. These factors are discussed in Chapter 7.

Morari (1982a), (1982b), (1983) and Grossmann and Morari (1983) also provide a flexibility or resilience index full details of which appear in an as yet unpublished paper by Morari and Woodcock (1983).

In a different approach to the structural problem Parkinson, Liebman, Pedersen and Templeman (1982) have applied a branch and bound tree searching procedure based on the thermodynamic work of Raghavan (1977) with some success. However, they only check for feasibility at one set of inlet conditions thus not guaranteeing full flexibility, even though the set of conditions used is problem dependent and does provide a severe test for the network.

On the basis of this literature review a number of requirements can be given for a new design scheme for flexible heat exchanger networks if it is to be an improvement on existing methods.

2.6 Requirements of a Design Scheme for Flexible Heat Exchanger Networks

1) Ability to deal with both the sizing and structural problem.
2) Optimal balancing of capital and operating cost.
3) Correct assignment of bypasses to the network not assuming that all exchangers are bypassed, thus requiring the consideration of actual exchanger areas at an early stage.
4) Ensuring that the network can handle the full operating range not just cornerpoints.

C H A P T E R 3

DEGREES OF FREEDOM ANALYSIS

3.1 Introduction

One of the requirements of a design scheme for flexible heat exchanger networks, identified in Chapter 2, is that bypasses must be placed correctly in a network for control purposes. Morari and co-workers' assumption that all exchangers are fitted with a controlled bypass need not necessarily be followed. A degrees of freedom analysis can be used to give information about how many bypasses need be included for control purposes and also to identify cases in which incorrect bypass placement makes adequate control impossible.

Marselle (1980) gives a very brief degrees of freedom analysis assuming that each exchanger can be used as a control device, which is only true if each exchanger is fitted with a bypass. The only conclusion reached by Marselle is that for all practical purposes the degrees of freedom for any heat exchanger network are greater than or equal to zero. As part of the purpose of the present work is to examine the assumption that all exchangers must be bypassed it is necessary to develop a new degrees of freedom analysis in which bypasses and exchangers can be considered separately and not as combined units.

3.2 An Example Problem

The method of analysis may be demonstrated using as an example the network given in Figure 3.1.

Heat capacity values, Cp, are assumed to be independent of temperature and overall heat transfer coefficients, U, are assumed to be invariant. Thus the F.Cp value remains constant along any unsplit process stream.

The equations governing this network are as follows:

Equations for Exchanger 1

$$H1 = (T4 - T5) \cdot F2 \cdot Cp2 \quad (3.1)$$

$$H1 = (T9-T8).F3.Cp3 \quad (3.2)$$

$$H1 = U1.A1.\ t(T4,T5,T8,T9) \quad (3.3)$$

Where t(T4,T5,T8,T9) is some function of those temperatures: for pure countercurrent single shell pass single tube pass exchangers a log-mean temperature difference relationship or a more complex function for multipass exchangers.

Equations for Exchanger 2

$$H2 = (T1- T2).\ F1.Cp1 \quad (3.4)$$

$$H2 = (T7- T6).F3.Cp3.R \quad (3.5)$$

$$H2 = U2.A2.\ t(T1,T2,T6,T7) \quad (3.6)$$

Equations for Utility Exchanger

It is assumed that all utility exchangers, both heating and cooling, are controlled and that the load of a utility exchanger can be altered so as to always bring its process stream to the required target temperature. If this is assumed the only equation required to describe the operation of a utility exchanger is the heat balance.

$$C1 = (T2-T3).F1.Cp1 \quad (3.7)$$

Equations for Mixer

$$T8 = R.T7 + (1-R).T6 \quad (3.8)$$

These eight equations fully describe the network as all other possible equations are implied by them. As noted above the F.Cp values are constant for any unsplit process stream and this has been assumed in the heat balance equations (3.1), (3.2), (3.4), (3.5), (3.7) and (3.8). The fact that the two outlet temperatures from the splitter are equal to the inlet temperature is assumed in equations (3.5), (3.6) and (3.8) and the flowrate balance around the splitter that the outlet

flowrate is equal to R or (1.0-R) (as appropriate) multiplied by the inlet flowrate is assumed in equations (3.5) and (3.8). Thus it is not necessary to give these equations in addition to (3.1) to (3.8).

The number of variables involved in the problem are:

$$\underset{T}{9} + \underset{F}{3} + \underset{Cp}{3} + \underset{U}{2} + \underset{A}{2} + \underset{R}{1} + \underset{H}{2} + \underset{C}{1} = 23$$

In an operating network a number of these variables take values which are either fixed by external process conditions, by physical property relationships or by design. They are:

1) The three inlet temperatures T1, T4, T6
2) The three flowrates F1, F2, F3
3) The three heat capacity values Cp1, Cp2, Cp3
4) The three outlet temperatures, T3, T5, T9, which are required to take fixed values
5) The two overall heat transfer coefficients U1, U2
6) The two exchanger areas A1, A2

Sixteen variables in all are required to take single predetermined values.

Therefore the number of unknown or state variables is:

$$23 - 16 = \underline{7}$$

The degrees of freedom (DF) are defined as:

Degrees of freedom (DF) = No. unknown variables - No. describing equations

∴ In this case DF = 7 - 8 = $\underline{-1}$

Relaxing the assumptions which have been made, if the heat capacities are assumed to be temperature dependent each stream section will have an associated heat capacity value described by a single equation in terms of the other variables. For each extra variable there will be one extra equation therefore the degrees of freedom value will be unaffected. Including a relationship for the overall heat transfer coefficients will also leave the degrees of freedom unaffected.

3.3 Interpretation of Degrees of Freedom

The calculated degrees of freedom for a network gives information about the solution of the network's describing equations. Three cases exist:

1) If DF<0

The set of equations have less unknown variables than equations and therefore in general no solution to the equations exists.

If a network were designed and operated with DF<0 one or more of the variables which were assumed to be fixed would, in fact, be unable to take their required values. This could only mean that one or more of the target temperatures could not be met. Therefore networks in which DF<0 should be avoided. (This is the case, then, with the network in Figure 3.1)

2) If DF = 0

The equations can be solved (for most cases) to find a solution for the intermediate temperatures, the heat loads and the split fractions (i.e. the unknown variables) which satisfies the describing equations.

One unique solution exists.

3) If DF>0

The equations have an infinite number of solutions. The target temperatures can be met for a range of values of intermediate temperatures, heat loads and split fractions.

When DF>0 it is possible to preset values for a number of the unknown variables until the degrees of freedom have been effectively reduced to zero. The values of these preset unknown variables can be altered to produce different solutions to the equations. These preset unknown variables are control variables, as discussed in the work of Grossmann and co-workers (see Chapter 2) and can be used for optimizing control.

3.4 General Calculation of Degrees of Freedom

If the following are defined:

E = No. of stream/stream heat exchangers

V = No. utility exchangers

X = No. mixers

S = No. Splitters

N = No. process streams (i.e. excluding utility streams)

The following values may be calculated for any general network:

Total No. Equations

$$\underbrace{(3.E)}_{\substack{\text{Heat Exchanger}\\ \text{Equations}}} + \underbrace{V}_{\substack{\text{Utility Exchanger}\\ \text{Equations}}} + \underbrace{X}_{\substack{\text{Mixing}\\ \text{Equations}}} \quad (3.9)$$

Total No. Variables

$$\underbrace{(N + 2E + V + X)}_{\text{Temperatures}} + \underbrace{N}_{\text{Flowrates}} + \underbrace{N}_{\substack{\text{Heat}\\ \text{Capacities}}} + \underbrace{E}_{\text{Areas}} + \underbrace{E}_{\substack{\text{Heat Transfer}\\ \text{Coefficient}}} + \underbrace{S}_{\substack{\text{Split}\\ \text{Fractions}}} + \underbrace{(E + V)}_{\text{Heat Loads}} \quad (3.10)$$

No. Predetermined Variables

$$\underbrace{N}_{\substack{\text{Inlet}\\ \text{Temperatures}}} + \underbrace{N}_{\substack{\text{Target}\\ \text{Temperatures}}} + \underbrace{N}_{\substack{\text{Flow-}\\ \text{rates}}} + \underbrace{N}_{\substack{\text{Heat}\\ \text{Capacities}}} + \underbrace{E}_{\text{Areas}} + \underbrace{E}_{\substack{\text{Heat Transfer}\\ \text{Coefficients}}} \quad (3.11)$$

$$\text{Total No. Equations} = 3E + V + X \quad (3.12)$$

$$\text{Total No. Variables} = 5E + 2V + X + S + 3N \quad (3.13)$$

$$\text{No. Predetermined Variables} = 2E + 4N \quad (3.14)$$

$$\begin{aligned}\text{Total No. Unknown Variables} &= \text{Total No. Variables} - \text{No. Predetermined Variables}\\ &= (5E + 2V + X + S + 3N) - (2E + 4N)\\ &= \underline{3E + 2V + X + S - N} \quad (3.15)\end{aligned}$$

$$\begin{aligned}\text{Degrees of Freedom} &= \text{Total No. Unknown variables} - \text{Total No. Equations}\\ &= (3E + 2V + X + S - N) - (3E + V + X)\\ &= \underline{V + S - N} \quad (3.16)\end{aligned}$$

$$\underline{\text{Degrees of Freedom} = \text{No. Utility Exchanger} + \text{No. Splitters} - \text{No. Process Streams}} \quad (3.17)$$

Therefore by adding splitters (in the form of stream-splits or bypasses) to a network it can be ensured that DF>0 which is essential for regulatory control purposes, that is the meeting of target temperatures.

In some cases, however, if DF>0 it does not necessarily follow that the describing equations for a network have a solution. The problem may be badly posed and if this is the case it will be impossible to operate the network so that target temperatures are met.

A number of situations can be identified in which this is the case and these should be avoided.

3.5 Badly Posed Problems

3.5.1 Streams not connected via stream/stream heat exchangers to a utility exchanger (heat source or sink)

For example, consider the network shown in Figure 3.2.

Using equation (3.16) the degrees of freedom for this network can be calculated:

$$DF = 0 + 2 - 2 = 0$$

Therefore generally it would be expected that a single solution to the describing equations exists.

However in this case it is obvious that a solution only exists, and the network is adequate, for the single case in which the heat to be removed from stream 1 is exactly balanced by that to be added to

stream 2. This network has no flexibility as it cannot handle changes in flowrates or inlet temperatures while maintaining target temperatures.

The describing equations can be demonstrated to be inconsistent.

The equations describing the system are:

$$H1 = (T1-T2).F1.Cp1.R1 \quad (3.18)$$

$$H1 = (T5-T4).F2.Cp2.R2 \quad (3.19)$$

$$H1 = U1.A1.\ t(T1,T2,T3,T4) \quad (3.20)$$

$$T3 = R1.T2 + (1.R1)T1 \quad (3.21)$$

$$T6 = R2.T5 + (1-R2).T4 \quad (3.22)$$

Rearranging (3.21) and (3.22)

$$R1 = (T1-T3) \,/\, (T1-T2) \quad (3.23)$$

$$R2 = (T6-T4) \,/\, (T5-T4) \quad (3.24)$$

Substituting for R1 and R2 in (3.18), (3.19), (3.20)

$$H1 = (T1-T3)\ F1.Cp1 \quad (3.25)$$

$$H1 = (T6-T4)\ F2.Cp2 \quad (3.26)$$

$$H1 = U1.A1.\ t(T1,T2,T3,T4) \quad (3.27)$$

Predetermined values are the inlet temperatures T1, T4, the target temperatures T3, T6, the flowrates F1, F2, heat capacities Cp1, Cp2 and the exchanger variables U1, A1.

Rewriting the equations (3.25), (3.26), (3.27) as functions of the unknown variables.

$$f1\ (H1) = 0 \quad (3.28)$$

$$f2\ (H1) = 0 \quad (3.29)$$

$$f3\ (T2,T3,H1) = 0 \quad (3.30)$$

Therefore overall there are three equations in three unknowns but equations (3.28) and (3.29) have only one unknown, H1, and therefore, in general cannot be solved.

It can be seen that any particular subset of streams not connected to a utility exchanger will require an exact heat balance of the above form and therefore should be avoided if flexibility is required.

In the work of Linnhoff et al in the I.Chem.E. User Guide (1982) this case, for an unpinched problem, is called a subset.

As a solution to literature problem 6SP1 Linnhoff (1979) gives the network shown in Figure 3.3.

It can be seen that the network has -4 degrees of freedom and that no bypasses are included. As the methods of Linnhoff and co-workers design networks for single sets of operating conditions the exchanger areas are effectively state variables which can be calculated to satisfy that particular set of conditions. However in an operating network which must bring all streams to their target temperatures the areas are fixed and the network will be unable to satisfy a range of conditions.

Even if bypasses were assigned to ensure that the degrees of freedom were zero Figure 3.4 shows that streams 4 and 1 form a subset not connected to a utility and the network will only operate if the heat loads of streams 1 and 4 are exactly balanced. This will clearly be very unlikely to occur in practice. Although Linnhoff (1979) claims that "this solution would probably be considered to be the most promising design in a practical environment" if a practical environment included inlet variations and the need to meet target temperatures on streams 1 and 4 the network would be a poor candidate.

If target temperatures were not rigidly fixed this may not be the case, see Linnhoff and Kotjabasakis (1984).

In summary, if a subset as defined by Linnhoff is present in a network it means that one set of streams may not be connected to a utility exchanger. If this is the case the network has no flexibility

as it is defined in the present work. (All streams must be able to be brought to preset target temperatures for all inlet conditions).

For an unpinched problem using only cooling or heating utilities (not both) a subset will always produce a set of streams isolated from a utility.

For a pinched problem two subsets (obviously a single subset cannot exist) may or may not produce a set of streams isolated from a utility exchanger as one subset may be connected to the cooling utility and one to the heating. If more than two subsets are present a set of streams will be isolated from a utility and this must be avoided.

3.5.2 Networks featuring an overspecified exchanger

Consider the network given in Figure 3.5.

Applying equation (3.16) to this network it can be seen that the degrees of freedom are zero, however the describing equations cannot be solved.

The equations for exchanger 2 are:

$$H2 = (T5-T6).F2.Cp2 \qquad (3.31)$$

$$H2 = (T8-T7).F3.Cp3 \qquad (3.32)$$

$$H2 = U2.A2.\ t(T5,T6,T7,T8) \qquad (3.33)$$

Variables which are fixed are the inlet temperatures T5 and T7, target temperature T6 and the values F2, F3, Cp2, Cp3, U2, A2.

∴ Writing (3.31), (3.32) and (3.33) in terms of the unknown variables

$$f1\ (H2) = 0 \qquad (3.34)$$

$$f2\ (H2, T8) = 0 \qquad (3.35)$$

$$f3\ (H2, T8) = 0 \qquad (3.36)$$

Here there are three equations in two unknowns and therefore in the general case no solution exists.

Even though the equations describing the entire network have the same number of unknown variables as equations this particular subset of equations has not.

Networks which include an exchanger which has five or more associated temperatures of flowrates preset should be avoided as the describing equations for that exchanger will in general have no solution and in an operating network this would result in target temperatures not being met.

In this case bypassing the exchanger would create two new unknown variables (a temperature and split fraction) and one extra mixing equation, a net gain of one variable thereby enabling the describing equations for the exchanger/bypass system to be solved.

A more subtle variant of the same problem occurs in the network shown in Figure (3.6)

For this network the degrees of freedom analysis gives DF = 1 therefore before the describing equations can be solved one of the unknown variables must be preset.

The variables in the network which can (and must) be directly controlled in practice are the utility exchanger heat loads C1 and W1 and the three split fractions. The degrees of freedom analysis

indicates that the two utility loads and two split fractions must be used for regulatory control, the meeting of target temperatures, and this is logical as there are four target temperatures to be controlled and there must be one control variable per controlled target variable. The remaining split fraction is the single degree of freedom indicated and this may be used for optimizing control, for example minimizing total utility use. Each value selected for the optimizing split fraction will produce a different solution for the regulatory control values and therefore different operating conditions in the network.

There are three possible choices for the optimizing control variable, split fractions 1, 2 or 3, but only one of these is acceptable.

For example if split fraction 2 is specified the degrees of freedom analysis indicates that, as now DF = 0, a single solution for the unknown variables should exist. In fact by presetting split fraction 2 the describing equations have been made inconsistent.

Consider the equations for exchanger 3 and its associated bypass, referring to Figure 3.7

$$H3 = (T6-T7).F2.Cp2.R3 \qquad (3.37)$$

$$H3 = (T10-T9).F3.Cp3 \qquad (3.38)$$

$$H3 = U3.A3.\ t(T6,T7,T9,T10) \qquad (3.39)$$

$$T8 = R3.T7 + (1-R3).T6 \qquad (3.40)$$

By referring to Figure 3.6 it can be seen that the variables which are preset are T6, T8, T9, F2, F3, Cp2, Cp3 and R3

Writing (3.37), (3.38), (3.39) and (3.40) in terms of the unknown variables:

$$f1(H3,T7) = 0 \qquad (3.41)$$

$$f2(H3,T10) = 0 \qquad (3.42)$$

$$f3(H3,T7,T10) = 0 \qquad (3.43)$$

$$f4(T7) = 0 \qquad (3.44)$$

These four equations have only three unknown variables and therefore in general have no solution. Therefore presetting split fraction 2, that is using it as an optimizing control, will lead to inadequate operation

The mixing equation (3.44) fixes a unique value for T7 and therefore associated with the exchanger are two fixed temperatures T6, T9, one fixed flowrate F3 and two variables which are effectively fixed if a value is specified for R3: the flowrate (R3.F2) and the temperature T7. Five flowrates and temperatures are effectively fixed and therefore the exchanger is overspecified.

By similar arguments it can be shown that if split fraction 1 is used for optimizing control exchanger 1 is overspecified. This is not immediately obvious and depends on the fact that T10 is uniquely specified by the describing equations for exchanger 3 and, therefore, exchanger 1 has effectively five fixed temperatures and flowrates, T1, T10, T11, (R1.F1) and F3.

If split fraction 3 is fixed the network's describing equations can be solved (the equations are consistent).

These observations have implications for control purposes. For this particular network:

1) Split fractions 1 and 2 must be used for regulatory control i.e. the meeting of target temperatures.
2) Split fraction 3 is the only one which can be used for optimizing control, the object of which is likely to be the minimization of utility use.

This is due to the fact that, as explained by Marselle (1980), after each move of the optimizing controller the regulatory system should be able to be brought to unique steady state values, which do not exist when the equations are inconsistent.

If split fractions 1 or 2 are used for optimizing control the split fraction values required for the regulatory system cannot be found, either theoretically by solving the describing equations or practically by attempting to control an actual network to meet target temperatures.

If a much larger network with many degrees of freedom were involved it might not be an easy task to determine which split fractions should be used for optimizing control. A computer based method of determining which to use and of checking for consistent equations in general is described in Chapter 5 and is demonstrated using this example in Appendix C.

The two cases above should be avoided when designing a network.

Using this degrees of freedom analysis a number of problems may be studied.

3.6 Applications of Degrees of Freedom Analysis

3.6.1 Unpinched problems with minimum number of units

For an unpinched problem using only heating or cooling utilities (not both) the I.Chem. E. User Guide (1982) gives the minimum number of units (excluding subsets, which have been shown to produce non-flexible networks) as:

$$(E + V)_{MIN} = N \qquad (3.45)$$

As has been shown in equation (3.16) the degrees of freedom of a network is given by:

$$DF = V + S - N$$

∴ Substituting from (3.45)

$$DF = V + S - (E + V)$$

$$DF = S - E \qquad (3.46)$$

As has been discussed the degrees of freedom for a network must be at least zero so, from (3.46), for a unpinched network with the minimum number of units there must be at least as many splitters as exchangers.

As an example the network in Figure 3.8(a) is unpinched (as only cooling utilities are needed) and has the minimum number of units as defined by (3.45). As it stands this network has -3 degrees of freedom. To obtain DF = 0 three stream-splits or bypasses must be introduced.

To comply with the rules developed in the badly posed problem analysis it is found that each exchanger must be involved in at least one split stream. Two possible configurations with DF = 0 are given in Figure 3.8(b) which has three bypasses, and Figure 3.8(c) which has two bypasses and one stream-split.

An unacceptable configuration with DF = 0 is shown in Figure 3.8(d). It can be shown that in this network the unbypassed exchanger 3 has effectively three preset temperatures and two preset flowrates and is therefore overspecified.

For unpinched problems with the minimum number of units a rule for asigning splitters is: introduce one splitter per exchanger, with each exchanger involved in at least one split stream.

For an unpinched network with M more than the minimum number of units then up to M exchangers can be left uninvolved in split streams. However some or all of these exchangers may be bypassed to provide additional degrees of freedom for optimizing and flexibility purposes.

For DF = 0 the networks have as many splits and utility exchangers as streams thus providing one manipulated variable for each target temperature enabling a regulatory control scheme to be devised. The choice between the many possible placements of splitters in a network cannot be made on the basis of a degrees of freedom analysis; dynamic, economic and flexibility considerations must be applied.

3.6.2 Pinched problems

In the work of Linnhoff et al in the I.Chem.E. User Guide (1982) a pinched problem is regarded as two unpinched problems, one above and one below the pinch. A typical example is shown in Figure 3.9(a). Using equation (3.16) this network has three degrees of freedom however in Linnhoff's analysis the three intermediate temperatures at the pinch take specific values dependent on the chosen value of the minimum temperature approach ΔT_{min}. These can then be regarded as stream starting or finishing temperatures and the problem split into two unpinched cases for which all starting and finishing temperatures are known, as shown in Figure 3.9(b). In this case DF = 0 for each unpinched subproblem above and below the pinch. Marselle, Morari and Rudd (1982) in their definition of resiliency require that this chosen ΔT_{min} value must be maintained for all operating conditions. A control scheme would require the calculation of the intermediate temperatures at the pinch reducing the problem to two regulatory control tasks one either side of the pinch. An alternative control scheme would be to treat the network as a whole and to use three splitters for optimizing control taking up the three degrees of freedom, and to use these to minimize utility use for each set of operating conditions encountered. If a pinched problem is partitioned into two unpinched subproblems the minimum number of units required must be calculated separately for each subproblem and therefore the total number of units used is likely to be greater than the minimum number of units calculated for the network taken as a whole, which for pinched problems the I.Chem.E. User Guide (1982) gives as:

$$(E+V)_{MIN} = N + 2 - (\text{No. separate sets of streams}) \qquad (3.47)$$

remembering that for a pinched problem two subsets can sometimes be present without restricting flexibility. From (3.47) the minimum acceptable number of units is N, as was the case in (3.45) for unpinched problems. If more than N units are present and all the exchangers are bypassed, as they must be if each unpinched subproblem has the minimum number of units, the degrees of freedom of the entire system will be greater than one.

For the example being considered, if the pinch is ignored and a bypass assigned so that DF=0 for the network taken as a whole as shown in Figure 2.9(c) then the control task is purely regulatory in that the only control objective is the meeting of the four network target temperatures. This network is incapable of meeting any specified intermediate temperatures and therefore a fixed ΔT_{min} cannot be maintained for all operating conditions. This type of approach can, however, prove very attractive as is demonstrated in Chapter 7.

As shown above the minimum acceptable number of units for a pinched problem is N (although this does not guarantee that a subset of streams isolated from a utility is not present). From (3.45) this is also the minimum acceptable number of units for an unpinched problem and the analysis and conclusions of Section 3.6.1 are equally applicable, therefore, to pinched problems when dealt with as a whole rather than partitioned into unpinched subproblems. In general a pinched problem will have M more than the minimum number of units and therefore M exchangers need not <u>necessarily</u> be bypassed.

3.7 Conclusions from Degrees of Freedom Analysis

1) For any network the degrees of freedom (DF) are given by:

$$\text{Degrees of freedom} = \text{No. utility exchangers} + \text{No. splitters} - \text{No. process streams}$$

2) For any network to be controllable (able to meet all target temperatures) the degrees of freedom must be greater than or equal to zero.

3) To avoid inconsistent describing equations for a network (indicating that generally the network would be impossible to control properly and that a number of target temperatures could not be met) the following features should be avoided:

   (a) Subsets of streams not connected via heat exchangers to a utility exchanger.
   (b) Networks featuring an overspecified exchanger, that is an exchanger which is required to operate with five or more preset flowrates and temperatures.

4) For the networks with degrees of freedom greater than zero care must be taken in ensuring that the split fractions chosen to be used as optimizing control variables do not cause the network's describing equations to become inconsistent.

5) For unpinched problems, to ensure that the degrees of freedom are greater than zero and that the network's describing equations are consistent the following must apply:

   (a) No. heat exchanger (E) + No. utility exchangers (V) > No. process streams (N)
   (b) No. splitters (S) > No. process streams (N) - No. utility exchangers (V)
   (c) At least (N-V) exchangers must be involved in split streams

6) For pinched problems where a fixed ΔT_{min} value must be maintained the minimum number of units and splitter assignment must be determined by partitioning the problem into two unpinched subproblems and treating these separately. This will usually mean that the degrees of freedom of the entire system will be greater than zero and a form of optimizing control will be needed: either controlling to intermediate temperatures or minimizing utility use.

By ignoring the fixed ΔT_{min} objective the control task may be reduced to a purely regulatory one and the rules in (5) can be applied to the network as a whole.

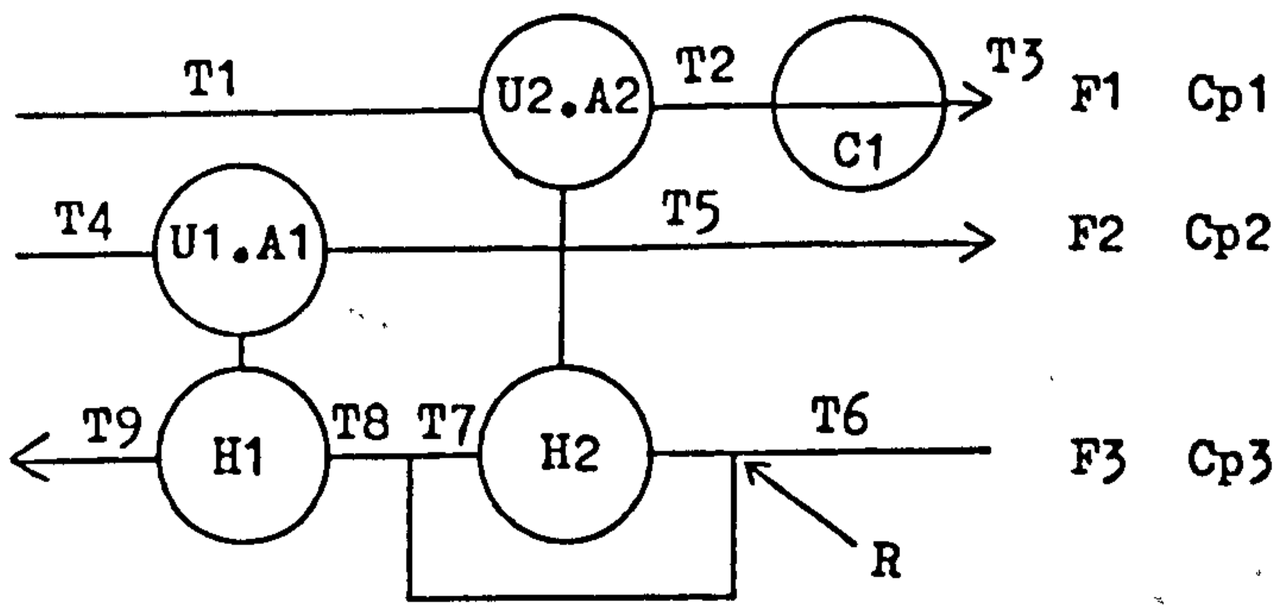


Figure 3.1 A network used as an example in the degrees of freedom analysis.

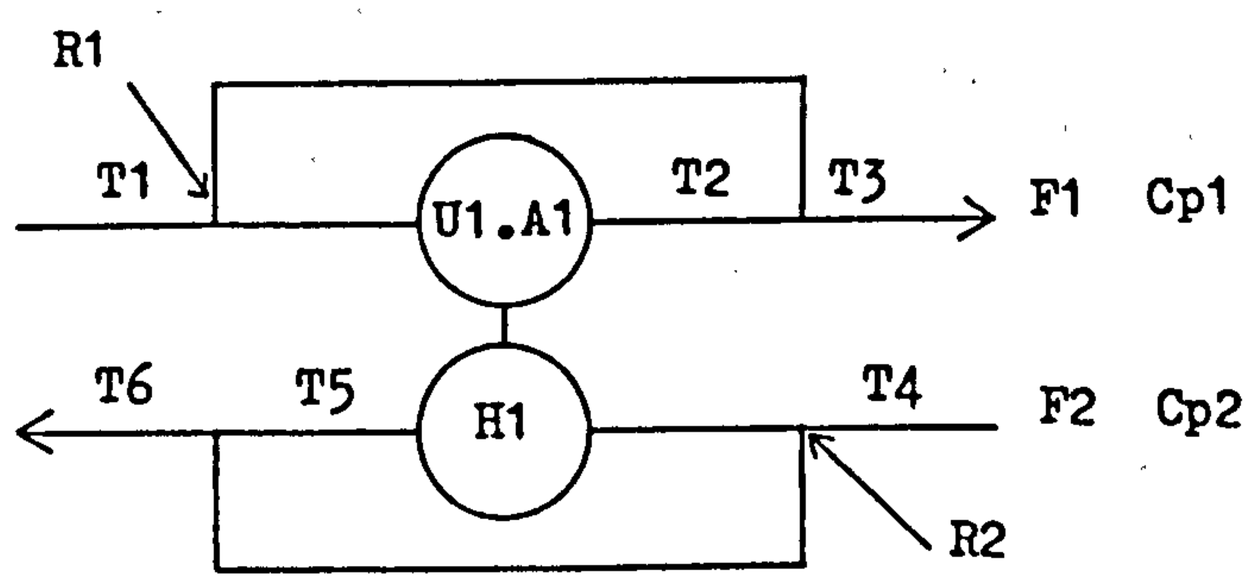


Figure 3.2 A network used to demonstrate a badly posed problem.

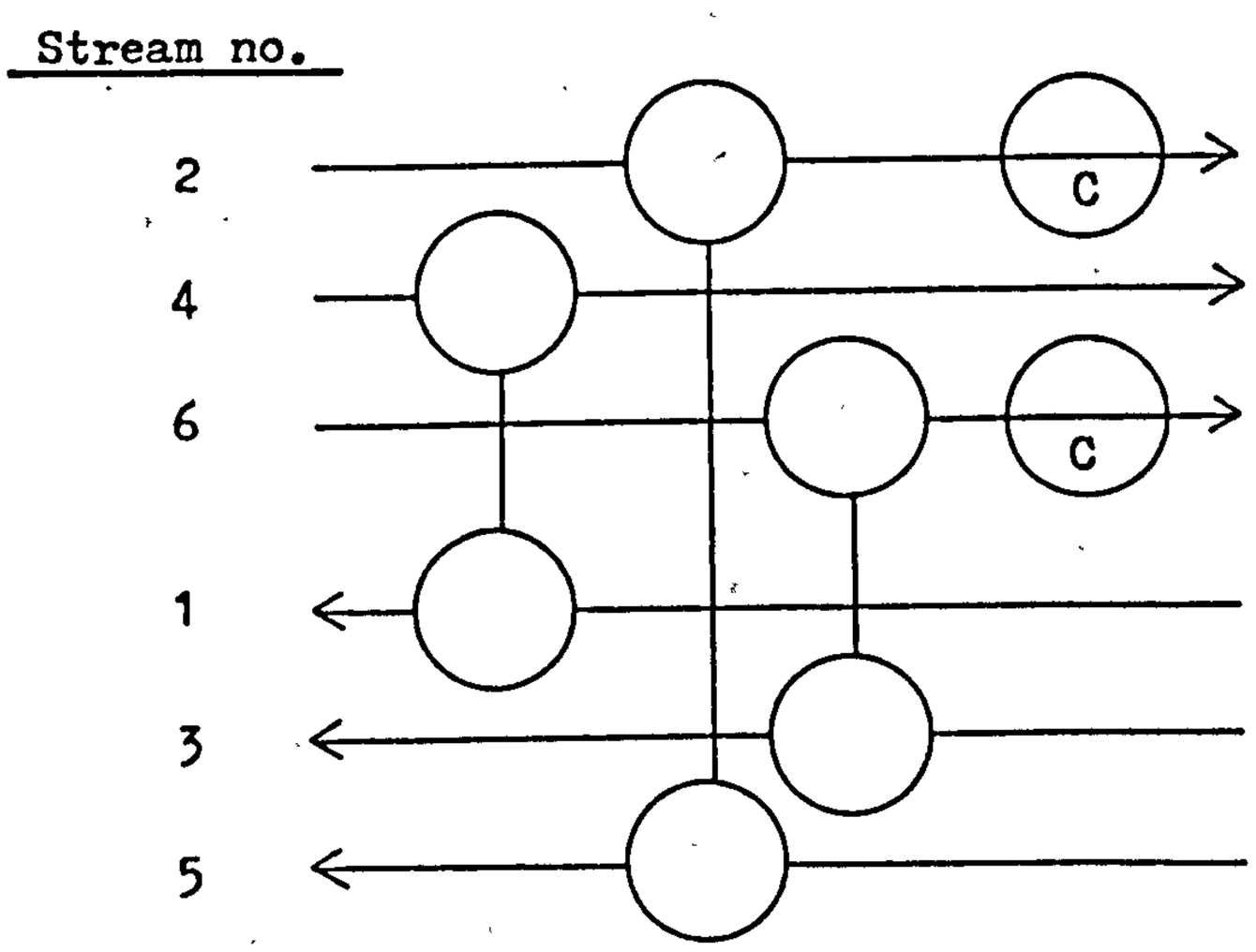


Figure 3.3 A solution given by Linnhoff to problem 6SP1.

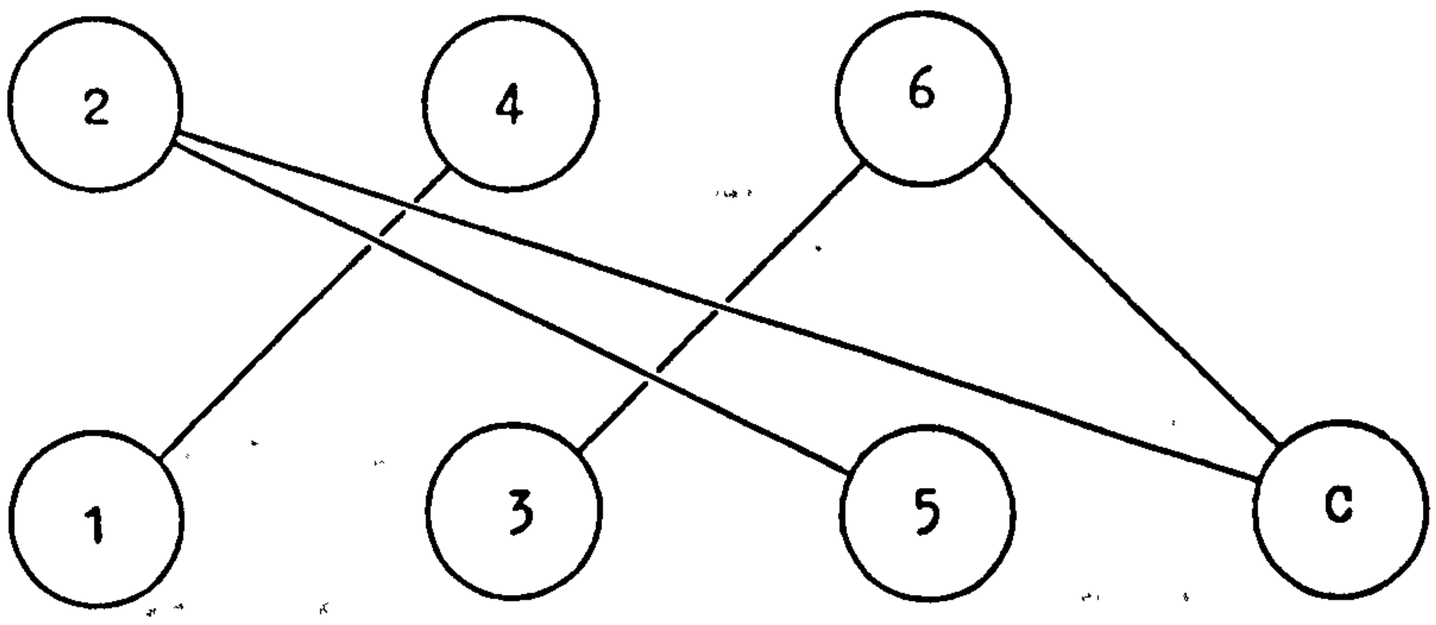


Figure 3.4 Network topology of Figure 3.3.

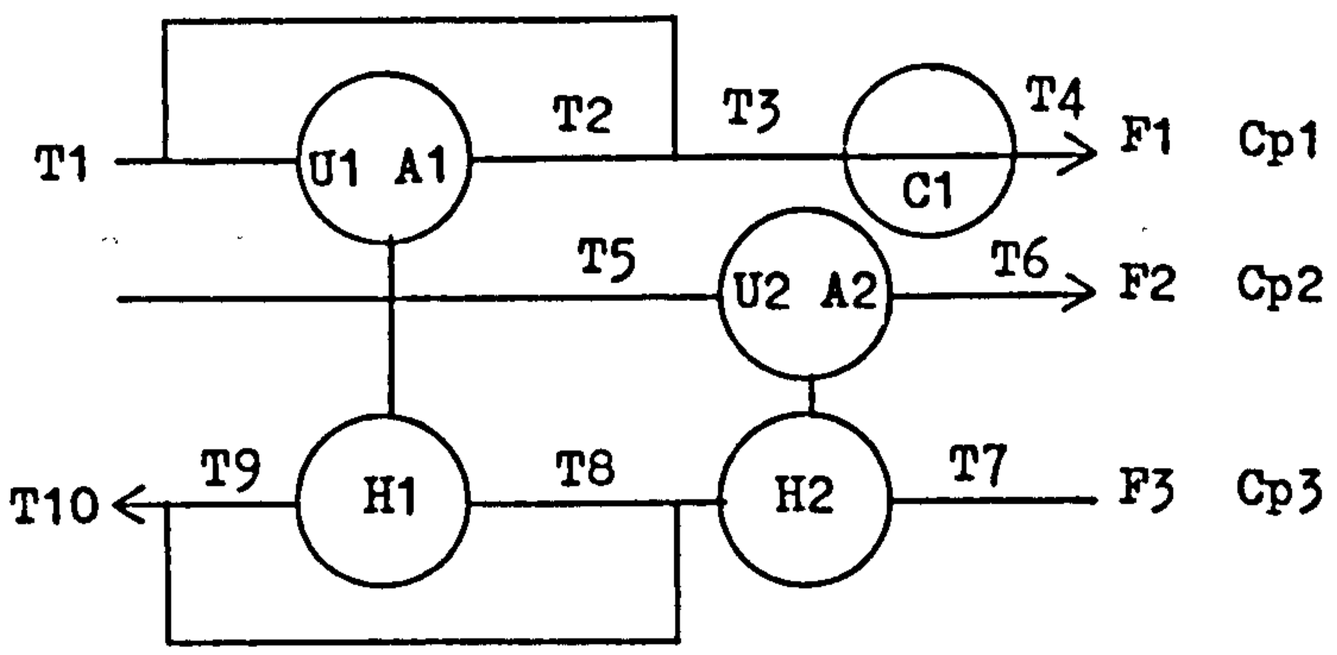


Figure 3.5 A network featuring an overspecified exchanger.

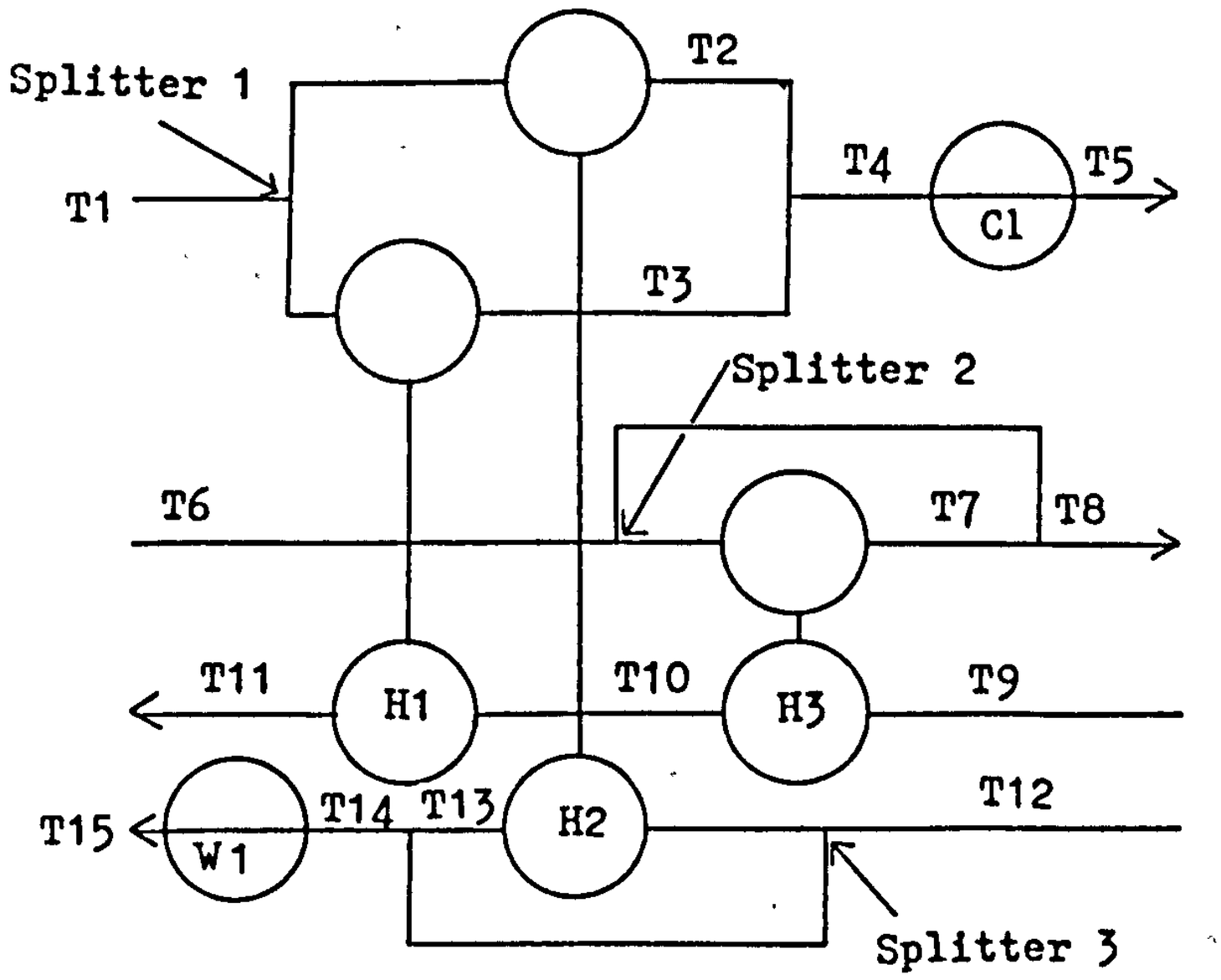


Figure 3.6 A network which requires optimizing control.

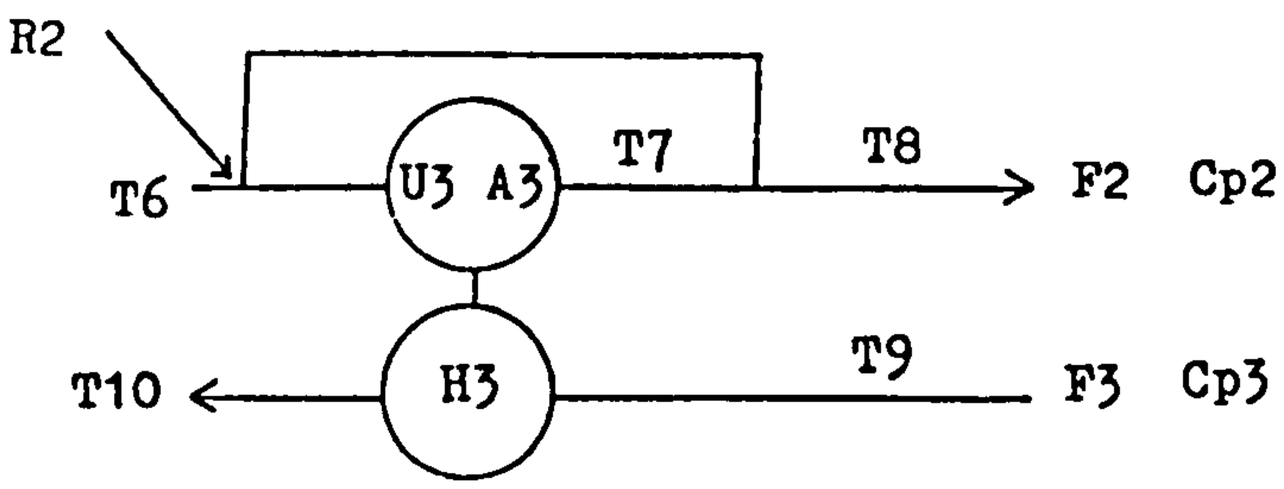


Figure 3.7 Exchanger 3 from the network given in Figure 3.6.

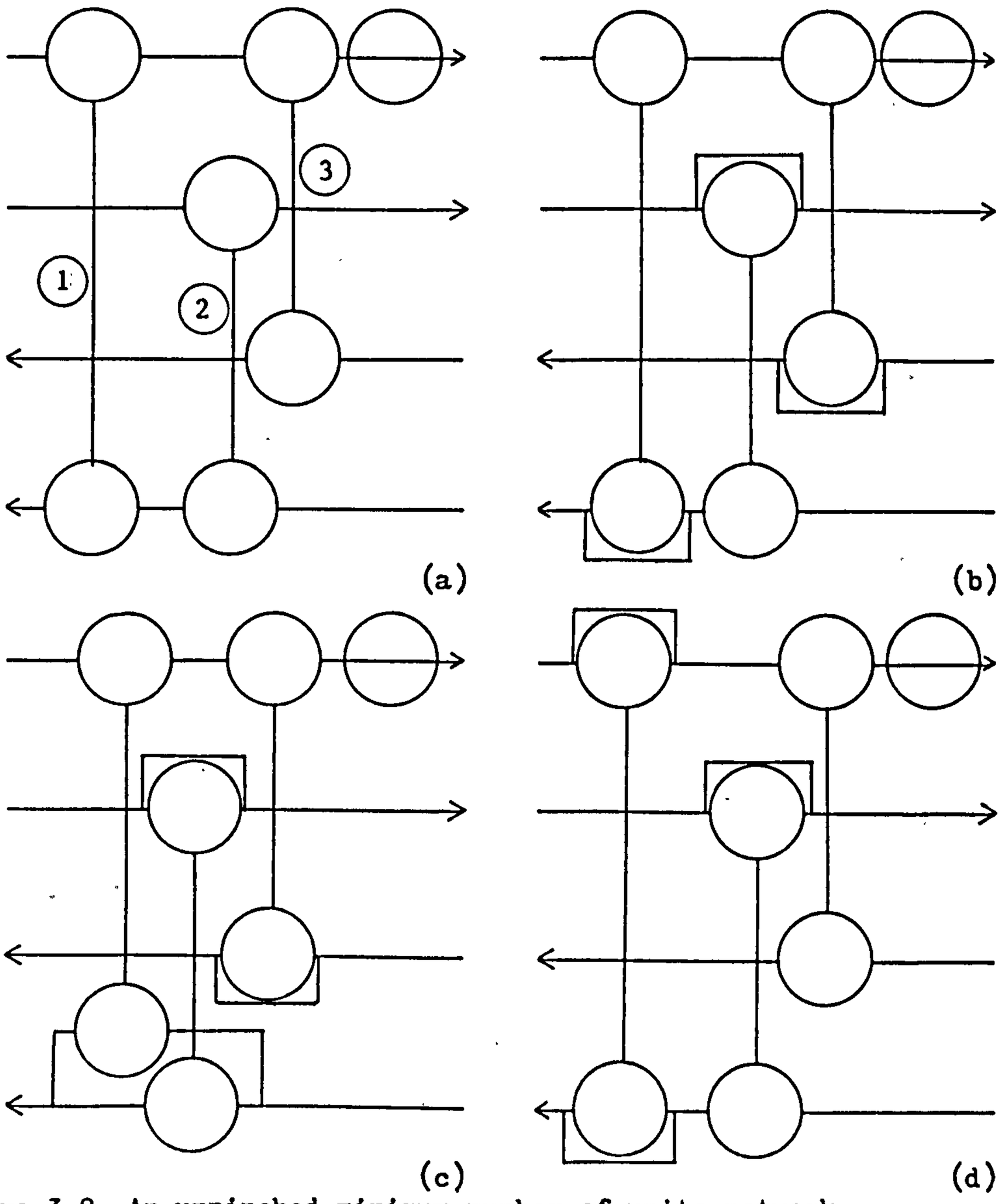
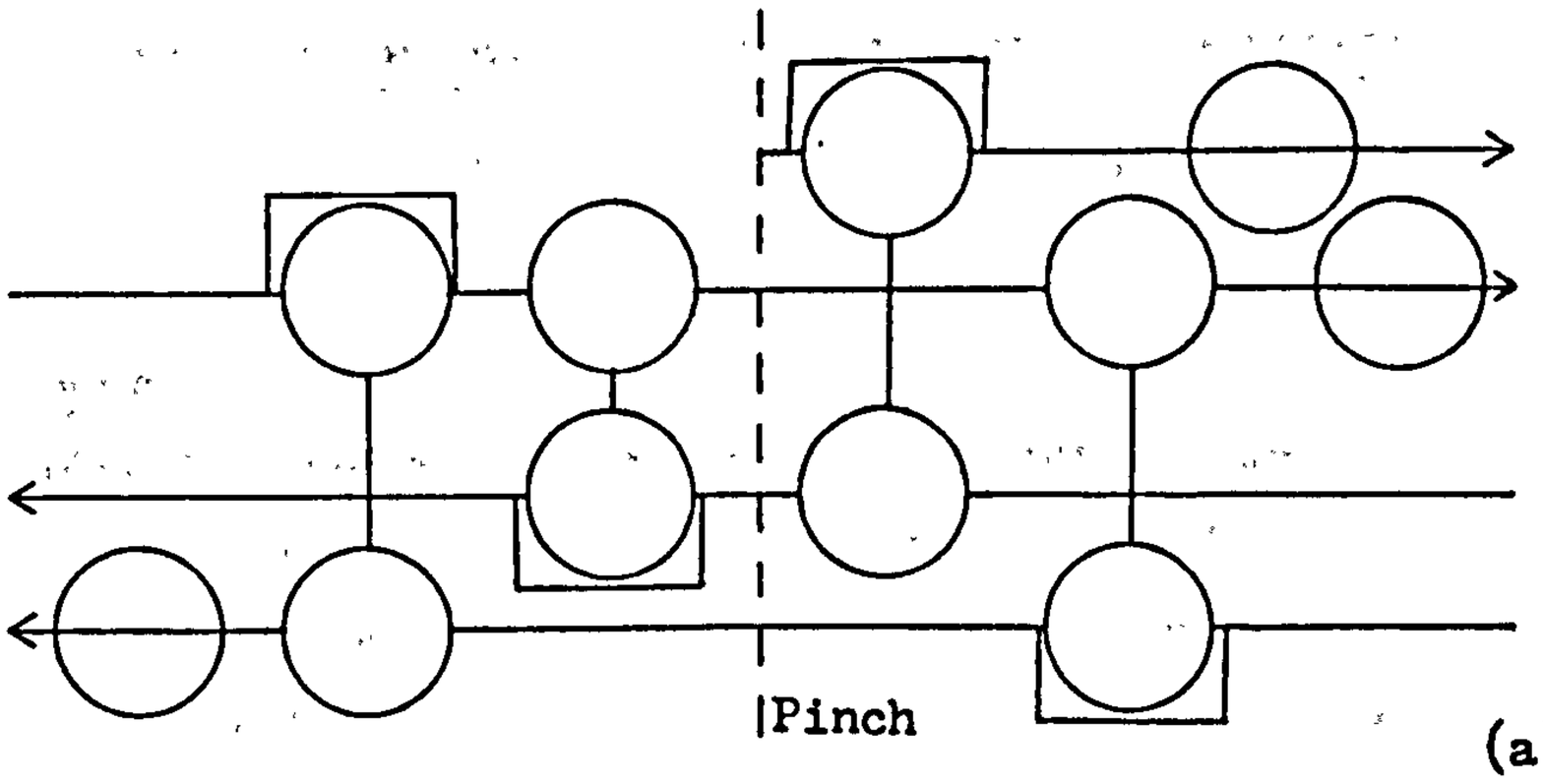
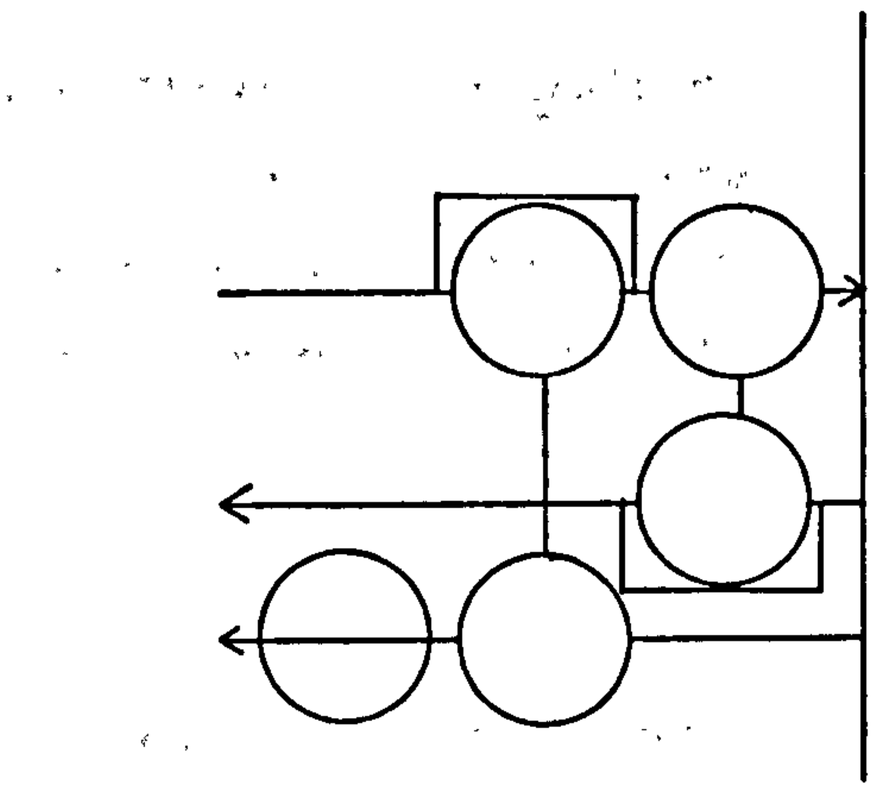
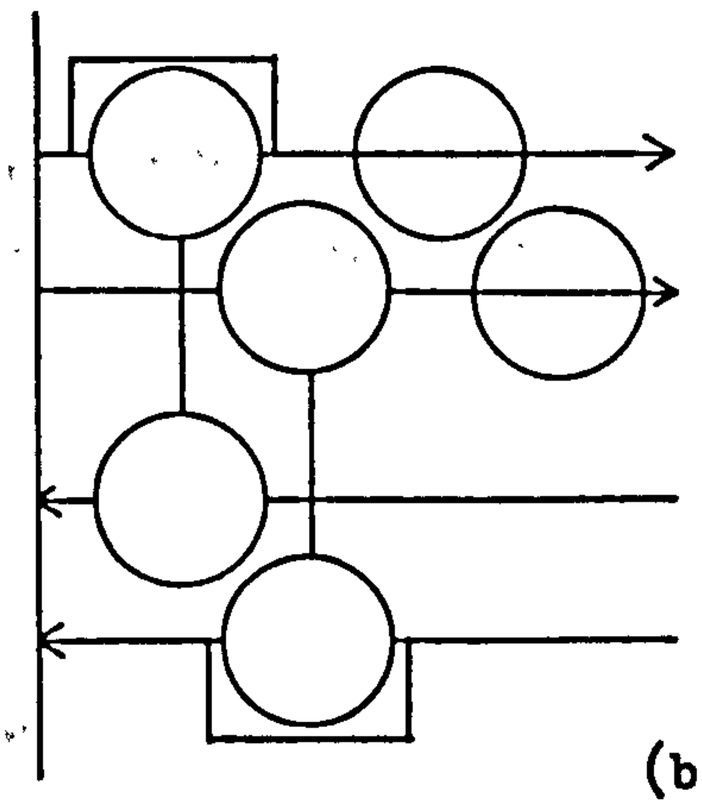


Figure 3.8 An unpinched minimum number of units network.

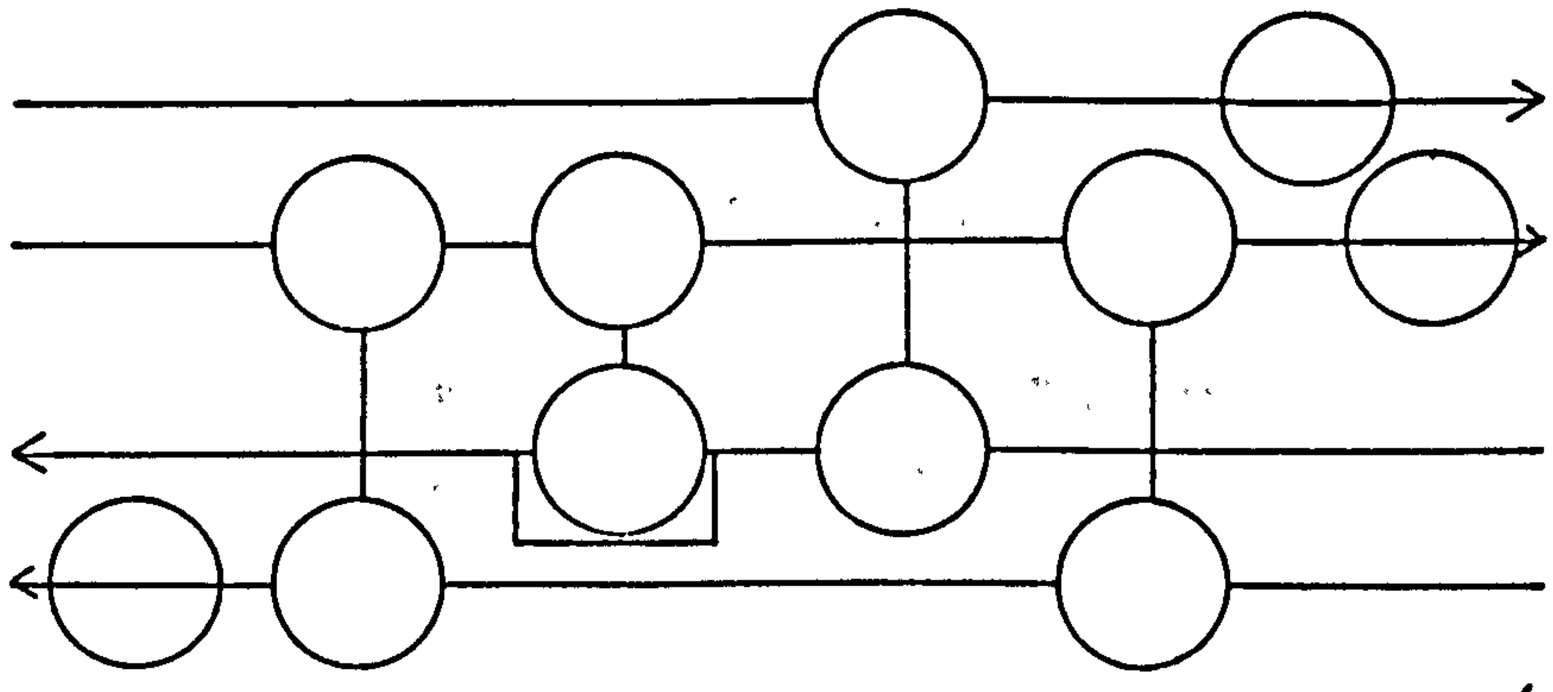


(a)





(b)



(c)

Figure 3.9 A pinched problem.

CHAPTER 4

DEVELOPMENT OF A NEW DESIGN METHOD FOR FLEXIBLE HEAT EXCHANGER NETWORKS

4.1 Introduction

In addition to having a consistent set of describing equations which must have at least one solution, as described in Chapter 3, a heat exchanger network must also satisfy certain operational constraints. These may be divided into two types: physical and imposed.

4.2 Physical Constraints

The network in Figure 3.1 was found to have -1 degrees of freedom. Assigning an extra bypass as shown in Figure 4.1 produces a network which has zero degrees of freedom.

The describing equations for this network are:

$$H1 = (T4-T5).R1.F2.Cp2 \quad (4.1)$$

$$H1 = (T10-T9).F3.Cp3 \quad (4.2)$$

$$H1 = U1.A1.\ t(T4,T5,T9,T10) \quad (4.3)$$

$$H2 = (T1-T2).F1.Cp1 \quad (4.4)$$

$$H2 = (T8-T7).R2.F3.Cp3 \quad (4.5)$$

$$H2 = U2.A2.\ t(T1,T2,T7,T8) \quad (4.6)$$

$$C1 = (T2-T3).F1.Cp1 \quad (4.7)$$

$$T6 = R1.T5 + (1-R1).T4 \quad (4.8)$$

$$T9 = R2.T8 + (1-R2).T7 \quad (4.9)$$

The degrees of freedom analysis indicates that this set of equations has a unique solution. As this is a purely mathematical argument, however, there is no guarantee that this solution represents a physically possible operating state. To ensure that the solution does represent physically possible operation all the following inequality constraints must be satisfied.

1) All exchanger temperature approaches must be greater than the minimum physically possible temperature approach, DT.
In this case:

$$T4 - T10 > DT \qquad (4.10)$$

$$T5 - T9 > DT \qquad (4.11)$$

$$T1 - T8 > DT \qquad (4.12)$$

$$T2 - T7 > DT \qquad (4.13)$$

Note that this temperature approach is <u>not</u> equivalent to Linnhoff's (1979) definition of a minimum allowable temperature approach. In the work of Linnhoff the minimum allowable temperature approach is set to provide a trade-off between capital and operating costs. The minimum physically possible temperature approach, DT, will be lower than this and represents the absolute minimum temperature approach which could occur in a given sized exchanger. A good approximation to DT could, quite reasonably, be zero.

2) All heat exchanger heat loads must be positive, implying that hot streams are reduced in temperature when passing through an exchanger and cold streams are increased in temperature (given feasible split fraction values).
In this case:

$$H1 > 0 \qquad (4.14)$$

$$H2 > 0 \qquad (4.15)$$

3) All split fraction values must be greater than 0.0 and less than 1.0.

In this case:

$$0.0 < R1 < 1.0 \qquad (4.16)$$

$$0.0 < R2 < 1.0 \qquad (4.17)$$

4.3 Imposed Constraints

Imposed constraints are those set by the designer in order to achieve a 'good' design according to some criteria. It is in this way that a new economic target can be set.

As has been discussed in Chapter 2 the design of a cost optimum network which is required to handle uncertain operating conditions is not only very difficult to achieve using current computational methods but also would probably produce a needlessly overexact solution given the highly uncertain nature of the operating conditions and economic data. In addition, the nature of an overall heat exchanger design scheme requires the production of a number of different flexible networks which can then be assessed on the basis of their dynamic behaviour as well as on direct economic differences.

The work of Linnhoff and co-workers as reported in the I.Chem.E. User Guide (1982) provides a useful method for setting economic targets. Linnhoff and co-workers conclude that because of the relatively high cost of utilities, especially steam, compared with that of equipment a network which recovers a high proportion of the recoverable heat will be very close to the 'theoretical' minimum cost network. In their work this target is used in a successful synthesis method for non-flexible problems which does not guarantee an optimal solution but does produce a good solution.

As energy prices are likely to remain high in the future this goal of 'high heat recovery' is a good one to adapt for a flexible design scheme. To implement this an extra constraint can be imposed on the

network to ensure high heat recovery throughout the operating range of uncertain inlet conditions.

For an unpinched problem which requires only heating or cooling utilities (not both) this extra constraint may be the requirement that all possible heat must be recovered for all points in the operating range. The application of this target to a network which requires only heating for some sets of inlet conditions and only cooling for others is described in Appendix A.

This heat recovery target is the same as that used in the work of Morari and co-workers discussed in Chapter 2. For pinched problems the heat recovery target is different. As discussed in Chapter 2 the requirement of Morari and co-workers that a network must maintain a particular value of the minimum allowable temperature approach is too strict. A better target could take the form that the amount of utilities used at any point in the operating range should be less than some specified multiple of the minimum theoretical utility use calculated at the same point in the operating range for the minimum physically possible temperature approach.

The heat recovery (or utility use) constraint could be, for example,

$$\frac{\text{Actual Utility Use}}{\text{Theoretical Minimum Utility Use}} < 1.2 \quad \text{applied through the range.}$$

There are many possible ways of formulating a heat recovery constraint and the choice of which to use is discussed further in Chapter 6. At this stage it is sufficient to note that a heat recovery constraint can be formed and included with the physical constraints.

Ensuring that all the constraints, physical and imposed, are satisfied for the entire operating range of inlet conditions, therefore, will ensure that the network can operate for all possible set of inlet conditions and that high heat recovery is ensured throughout the range.

In this way it will be seen that the solution of equation (2.6) to find an optimum cost solution is not necessary. By ensuring that a network satisfies the feasibility constraint (2.5) it is assured not only that the network will operate for all conditions but also that high heat recovery is achieved.

4.4 Incorporating the Feasibility Constraint into a Design Procedure

In order to apply the feasibility constraint (2.5) the first stage of a design procedure is to use a non-flexible synthesis method, such as that of Linnhoff et al presented in the I.Chem.E. User Guide (1982), to design a base network for any single set of operating conditions in the operating range of inlet conditions. Both the network structure (placement of exchangers and splitters) and size (exchanger areas) should be determined. It is preferable that the synthesis method used should generate a number of alternative networks rather than a single 'optimum' design as it is impossible to determine at this stage the best structure for a flexible network. Bypasses should be assigned according to the rules developed in Chapter 3 in order to ensure that the degrees of freedom of the system are greater than or equal to zero.

The first stage of the design procedure is based on computer modelling of the designed base network. The computer model can be used to solve the network's describing equations and hence for any particular set of uncertain inlet conditions it will calculate the regulatory control split fractions that are required in order to bring the stream target temperatures to their specified values. As all the other state variables are also calculated the network constraint function values, both physical and imposed, can be calculated and used to determine whether the solution represents feasible operation.

If one (or more) of the constraints is violated it is clear that the solution does not represent feasible operation. If optimizing control variables are present it may be possible to manipulate these to produce feasible network operation. If the constraints cannot be satisfied or the degrees of freedom are zero then the conclusion is

that the network cannot handle the particular set of inlet conditions under consideration and must be altered, either in size or structure, to accommodate them.

It is preferable that the final flexible network design should retain zero degrees of freedom as then the control task is purely regulatory, provided that optimizing control is not needed to achieve higher heat recovery. For a network with zero degrees of freedom the feasiblity constraint (2.5) can be adapted in order to produce an automated method for searching for a point in the operating range which produces infeasible operation, that is searching for a set of possible inlet conditions which the network under consideration cannot handle. If no such set of conditions can be found using the procedure then the network is feasible for the entire operating range.

As was shown in equation (2.8), and interpreting the nomenclature as in Table 4.1, a system is fully feasible (for zero degrees of freedom) if the following applies:

$$\max_{\theta^J \in D} \; g_j(d,\theta^j) < 0 \quad \text{for each } j \in J \tag{4.18}$$

where g_j; $j=1,J$ are the constraint functions developed from the physical and imposed constraints discussed above.

Writing (4.18) in terms of a search procedure:

For each constraint in turn ($j=1,J$)

a) Check that the "worst" possible value in the operating range is feasible.

$$\text{Is } \max_{\theta^J \in D} \; g_j(d,\theta^j) < 0 \; ? \tag{4.19}$$

b) As a secondary, but not essential, check ensure that all other constraints at this point are feasible.

$$\text{Are } g_i(d,\theta^j) \leq 0 \text{ for each } i=1,J \text{ except } i=j \text{ ?} \qquad (4.20)$$

As this procedure involves only a maximization it can be implemented effectively and linked to a network modelling program. Note, however, that global maximization is required.

If the degrees of freedom are greater than zero then the equivalent form of equation (4.18) is:

$$\underset{\theta^j \in D}{\text{Max}} \quad \underset{z^j}{\text{Min}} \quad g_j(d, z^j, \theta^j) \leq 0 \quad \text{for each } j \in J \qquad (4.21)$$

This formulation (4.21) does not guarantee full feasibility but it is highly probable that such a procedure will detect any conditions producing infeasible operation.

The method has the immediate advantage of searching for infeasible operation throughout the operating range rather than just at corner-points.

An outline design scheme based on this search procedure is given in Figure 4.1. Note that a range of initial networks can be considered and this may lead to a number of acceptable fully feasible designs for the same problem. The final design can then be chosen on the basis of capital cost, control and dynamic behaviour and other considerations.

It can be seen that this outline design scheme includes a number of procedures which have not been discussed so far: distinguishing between sizing and structural infeasibilities, determining which exchangers to oversize and how to modify a network to remove structural infeasibilities. To clarify these procedures it is necessary to implement a form of the design scheme in a computer program and investigate a number of example problems.

d = equipment design parameters which are fixed values: exchanger areas.

x = state variables which can be determined by solving the set of equations representing the process system: intermediate stream temperatures, F.Cp values, exchanger heat loads, utility heat loads and regulatory split fraction values.

z = Control variables which may be adjusted during operation to meet utility and operating targets: optimizing split fraction values

θ = uncertain variables: inlet temperatures and F.Cp values

In addition to these variables it is assumed that the following are fixed by the problem data: stream target temperatures and overall heat transfer coefficients.

Table 4.1 Interpretation of the general nomenclature for the specific treatment of heat exchanger networks used in the present work.

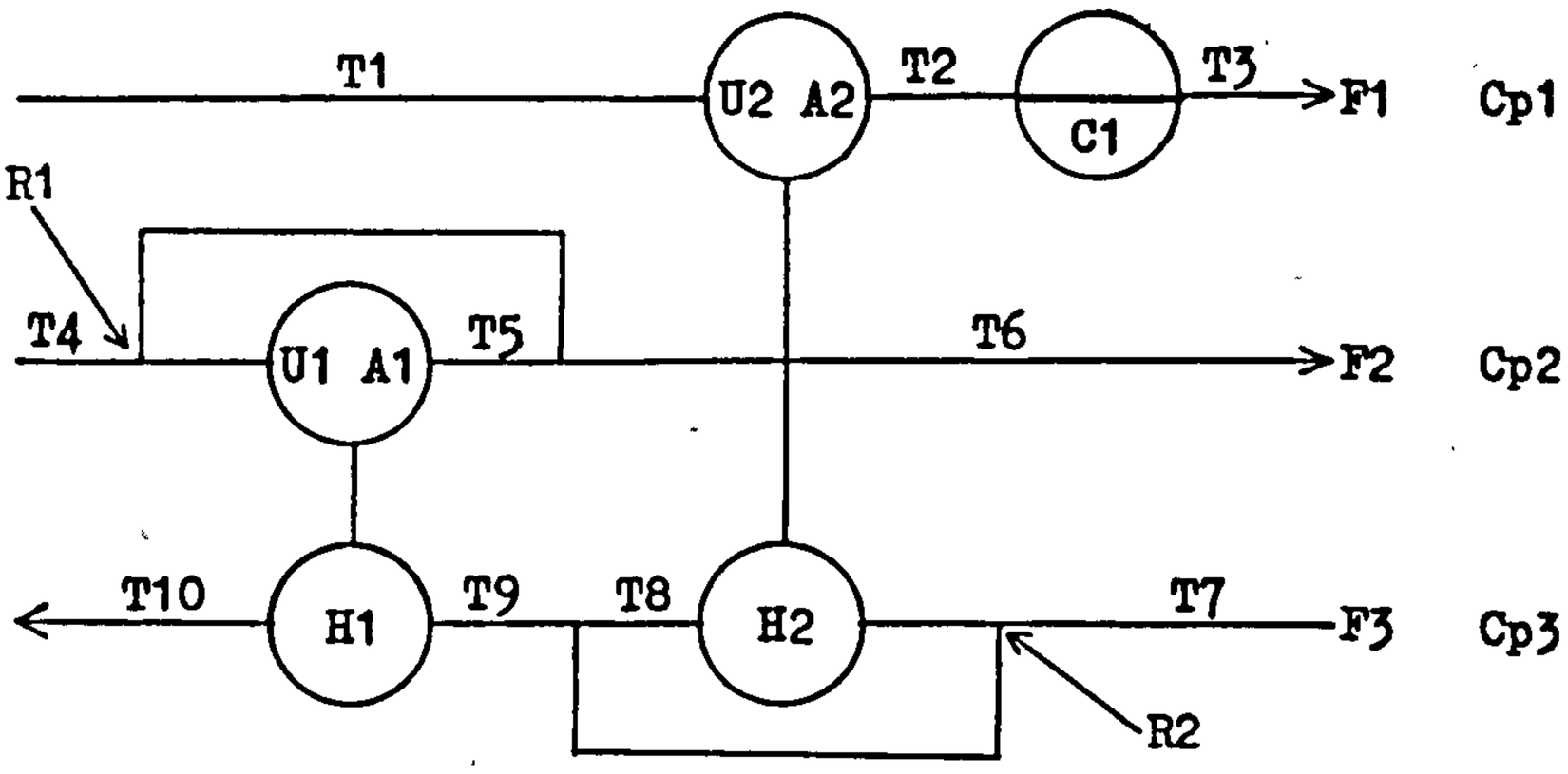


Figure 4.1 A network used to identify physical constraints.

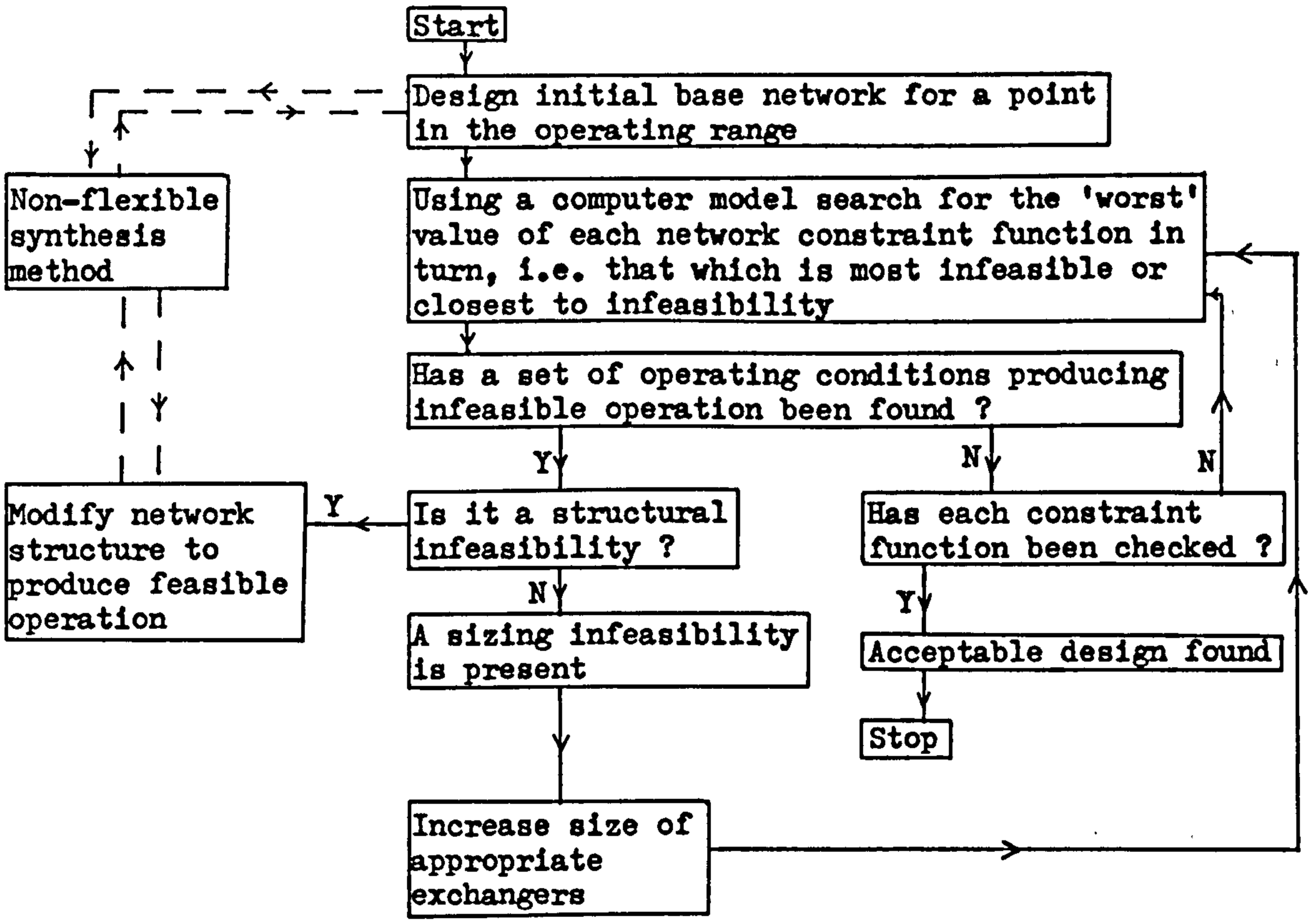


Figure 4.2 An outline design scheme for the design of flexible networks.

C H A P T E R 5

COMPUTER IMPLEMENTATION OF THE DESIGN SCHEME

5.1 Introduction

A computer program has been developed which implements part of the design procedure described in Chapter 4. The object of the program is to model any particular designed network and search for a particular set of inlet conditions, from within a given operating range, which the network cannot handle.

The general form and specific features of the program are described in the following chapter.

5.2 Network Modelling

5.2.1 Network representation

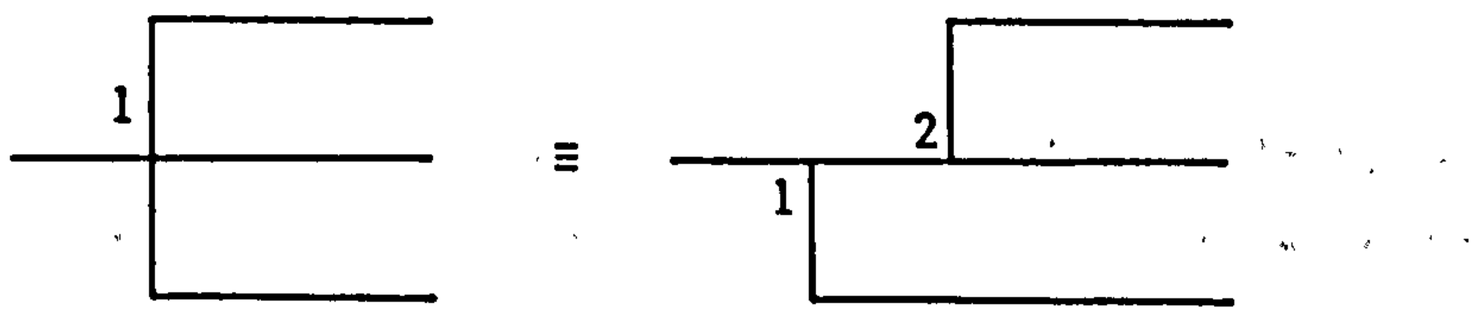
In the work of Marselle (1980), the checking of a network at each cornerpoint of an operating range requires the writing of a new computer program for each different network studied. This is time consuming and inefficient and a better approach is to write a completely general program which will model any specified arrangement of heat exchangers, utility exchangers, splitters and mixers. In the program developed in the present work the network configuration is specified in the input data supplied to the program as a process matrix. The form of the process matrix used is similar to that used in unpublished programs used by Grossmann (1977).

For the network under consideration each substream is numbered and the input and output streams specified for each unit. The process matrix for a simple example is shown in Figure 5.1.

Advantage is taken of the limited number of types of units which are required and the fact that no unit has more than two inputs and two outputs. Units such as stream-splits which do not have two inputs and two outputs are specified by indicating each 'missing' stream with a zero. By detecting the position of the zeros the program determines

whether a unit is an exchanger, a splitter, a mixer or a utility exchanger.

Any arrangement of units is possible except for the restriction that all utility exchangers must be placed at the end of process streams. This is not always appropriate but reflects the fact that the methods of Linnhoff and co-workers often tend to place the utilities there. (In some circumstances if steam is being raised or if multiple levels of utilities are available it is better to distribute a number of utility exchangers along a process stream). Only two-way splitters (and mixers) may be specified but a splitter (or mixer) of more than two ways may be represented as two or more two-way splitters in series.

e.g. 

Data Input

In addition to the process matrix the following data are also required by the program:

1) Upper and lower bounds on the heat capacity flowrate, F.Cp, value of each process stream.
2) Upper and lower bounds on the inlet temperature of each process stream.
3) The target temperature of each process stream (which is invariant).
4) The minimum possible temperature approach for all exchangers.
5) The U and A value of each exchanger.
6) Data relevant to a heat recovery target constraint.

5.2.2 First stage of analysis

A network modelling program can only consider one set of inlet conditions at a time. A number of options exist within the program for setting particular values for the inlet F.Cp values and temperatures

from within the operating range of inlet conditions. A particular combination of upper and lower bounds on the inlet F.Cp and temperature values can be specified simply by entering the number of a cornerpoint to be studied (a consistent method of numbering cornerpoints is used) and this facility enables automatic checking of each cornerpoint in turn without having to change the input data. Alternatively a particular choice of inlet F.Cp values and temperatures may be supplied directly to the program or a particular set of values may be generated automatically as part of an optimization procedure; these need not be cornerpoints.

Once a set of inlet variables has been specified a number of quantities may be calculated.

1) F.Cp values, which are assumed to be temperature independent, may be calculated for substreams not involved in split streams. Referring to Figure 5.1, for example, the F.Cp value of substream 2 may be set equal to the F.Cp value of substream 1 and these values will be maintained at the solution of the describing equations.

2) Using only the inlet and outlet temperatures and F.Cp values a subroutine based on Linnhoff's (1979) work may be used to calculate the theoretical minimum required hot and cold utility use and to determine whether the problem is pinched or unpinched for any particular specified minimum temperature approach. These values are of use in setting a heat recovery target constraint.

3) An initial guess of the temperatures and split fractions which cannot be immediately calculated can be made. This can be a direct guess for each value supplied directly to the program or a 'sensible' guess can be generated automatically based on the inlet and outlet temperatures of each process stream which ensures that process stream temperatures decrease on passing through exchangers for hot process streams and increase for cold process streams.

4) Equations which can be solved directly to give particular substream states may be applied:

   (a) The two outlet temperatures from a splitter must equal the inlet temperature.

   (b) The two outlet F.Cp values from a splitter must equal the appropriate split fraction value times the inlet F.Cp value.

   (c) The F.Cp values of all the substreams in the same branch of a split stream must be equal.

5) The degrees of freedom of the network may be calculated using the procedures developed in Chapter 3. If the calculated degrees of freedom are less than zero then the describing equations cannot be solved and the program stops. If the degrees of freedom are greater than zero then a number of split fractions must be used for optimizing control. As has been demonstrated in Chapter 3 the choice of which to use as optimizing control variables may not be obvious. The program contains a routine which will generate successive choices of which split fractions to use until either all combinations have been tried or a suitable choice is found.

5.2.3 Equation solving

5.2.3.1 Choice of equations to solve by iterative methods

The network modelling program required for the new design scheme is different in form to most flowsheet simulators. In general a flowsheet simulator takes a designed system and for specified inputs to the system calculates the resulting output conditions. This is the basis of general sequential-modular flowsheeting packages and of many programs written for particular applications such as the crude train modelling program described by Al-Zakri and Bell (1981a), (1981b).

In the present work both the inputs and outputs of a heat exchanger network are fully specified and the program must calculate the state variables (including regulatory split fractions) which are required to meet the specified output values.

To solve this problem an equation-oriented approach must be used, that is the program must set up all the network's describing equations and solve them simultaneously. The equations governing a heat exchanger network cannot be solved in a single step if the exchanger models include a log-mean temperature difference . An iterative calculation scheme must be used and a number of library routines exist for the solution of this type of problem.

The basis of all such solution procedures is that they produce successive estimates of the vector $\bar{x}_s$ which is a solution to the set of equations $f(x) = 0$.

i.e. x_{sj}, $j=1,J$ is a solution to $f_i(x_1, x_2 \ldots x_J) = 0$, $i=1,J$

The number of variables (J) must equal the number of equations.

There are a number of choices of which describing equations to solve by iterative methods. A number of possibilities can be illustrated by considering as a simple example the network given in Figure 5.1. This network is shown in its usual representation in Figure 5.2.

The equations describing this network, rearranged into a form suitable for an equation solving routine, are: (following the format used in Chapter 3)

$$H1 - (T4-T5).R1.F2.Cp2 = 0 \quad (5.1)$$

$$H1 - (T10-T9).F3.Cp3 = 0 \quad (5.2)$$

$$H1 - U1.A1.\ t(T4,T5,T9,T10) = 0 \quad (5.3)$$

$$H2 - (T1-T2).F1.Cp1 = 0 \quad (5.4)$$

$$H2 - (T8-T7).R2.F3.Cp3 = 0 \quad (5.5)$$

$$H2 - U2.A2.\ t(T1,T2,T7,T8) = 0 \quad (5.6)$$

$$C1 - (T2-T3).F1.Cp1 = 0 \quad (5.7)$$

$$T6 - R1.T5 + (1-R1).T4 = 0 \quad (5.8)$$

$$T9 - R2.T8 + (1-R2).T7 = 0 \quad (5.9)$$

Rewriting (5.1) to (5.9) in terms of the unknown variables:

$$f_1(H1,T5,R1) = 0 \quad (5.10)$$

$$f_2(H1,T9) = 0 \quad (5.11)$$

$$f_3(H1,T5,T9) = 0 \quad (5.12)$$

$$f_4(H2,T2) = 0 \quad (5.13)$$

$$f_5(H2,T8,R2) = 0 \quad (5.14)$$

$$f_6(H2,T2,T8) = 0 \quad (5.15)$$

$$f_7(C1,T2) = 0 \quad (5.16)$$

$$f_8(T5,R1) = 0 \quad (5.17)$$

$$f_9(T8,T9,R2) = 0 \quad (5.18)$$

Note that it has been assumed that the equations described in Section 5.2.2 (1) and (4) have been applied e.g. the three temperatures associated with a splitter have been assumed to be equal.

The set of equations (5.10) to (5.18) may now be supplied directly to an equation solving routine which will find a solution for the vector of unknown variables:

$$\bar{x} = (H1,H2,C1,T2,T5,T8,T9,R1,R2)$$

On the first iteration $\bar{x}_o$ will contain the guessed values of heat loads, temperatures and split fractions.

The routine then produces successive sets of values for $\bar{x}$ until the set of equations (5.10) to (5.18) are satisfied to within a specified accuracy.

For each successive set of values in $\bar{x}$ the equations described in Section 5.2.2 (4) must be reapplied as the new values in $\bar{x}$ may include different split fraction values and substream temperatures.

The problem may be partitioned in an attempt to reduce the computer time used by reducing the dimensions of the problem.

Partition 1

It is possible to reduce the number of equations to be solved by iteration to the following seven:

$$f_1(H1,T5,R1) = 0 \qquad (5.19)$$
$$f_2(H1,T9) = 0 \qquad (5.20)$$
$$f_3(H1,T5,T9) = 0 \qquad (5.21)$$
$$f_4(H2,T2) = 0 \qquad (5.22)$$
$$f_5(H2,T8,R2) = 0 \qquad (5.23)$$
$$f_6(H2,T2,T8) = 0 \qquad (5.24)$$
$$f_8(T5,R1) = 0 \qquad (5.25)$$

with the vector of unknown variables $\bar{x}' = (H1,H2,T2,T5,T8,R1,R2)$.

For each new vector $\bar{x}'$ produced by the routine not only must the equations in Section 5.2.2 (4) be applied before the new function values given by the left-hand sides of (5.19) to (5.25) can be calculated, but the values T9 and C1 must also be directly calculated from (5.16) and (5.18) as these variables have now been left out of $\bar{x}'$.

$$C1 = f_7'(T2) \qquad (5.26)$$
$$T9 = f_9'(T8,R2) \qquad (5.27)$$

Thus two variables and hence two equations have been removed from the iteration.

Partition 2

Another partition can be formed by including in the iteration only one equation per exchanger:

$$f_3(H1,T5,T9) = 0 \qquad (5.28)$$

$$f_6(H2,T2,T8) = 0 \qquad (5.29)$$

$$f_8(T5,R1) = 0 \qquad (5.30)$$

$$\bar{x}'' = (T8,R2,R1)$$

In this case for each new value of $\bar{x}''$ the following variables must be directly calculated:

$$H2 = f_5''(T8,R2) \qquad (5.31)$$

$$T2 = f_4''(H2) \qquad (5.32)$$

$$T9 = f_9''(T8,R2) \qquad (5.33)$$

$$H1 = f_2''(T9) \qquad (5.34)$$

$$T5 = f_1''(H1,R1) \qquad (5.35)$$

$$C1 = f_7''(T2) \qquad (5.36)$$

Note that this solution scheme depends on the fact that the equations (5.31) to (5.36) can be expressed directly in the form of the explicit variable. This is not the case when a log-mean temperature difference relationship is present, for example (5.3) cannot be manipulated to give an explicit expression for T5.

Most solution procedures attempt to minimize the number of torn variables, that is the number of variables used in the iterative procedure. Of the examples given above Partition 2 with only three torn variables would be preferred. However it may be better to solve a large number of almost linear equations than a small number of more nonlinear ones. In the present work, in trials based on a limited number of examples it was found that it is generally better, in terms of number of iterations and computer time used to find a solution, to use three equations and variables per exchanger rather than two or one

in the iteration. The scheme used is as that in Partition 1. This observation is unlikely to be generally true as the best partition is likely to be dependent on the numerical methods used, the initial guess of the solution vector and other factors.

5.2.3.2 Solution procedure for the general case

In the general case the equations supplied to the iterative solution procedure are:

1) Three equations per exchanger
2) One equation for each mixer of which the output is also an output from the network. (The temperature outputs from other mixers need not meet any target values).

The variables used in the iteration are:

1) Every split fraction not used for optimizing control i.e. the regulatory controls.
2) One heat load per heat exchanger.
3) Every temperature which is an output from a heat exchanger but not an output from the network.

Using the same nomenclature as Chapter 3:

From equation (3.16) the number of splits required for regulatory control (that is required to make the degrees of freedom of the network zero) is (N-V).

If V utility exchangers are present (and the restriction used in the present work that a utility exchanger can only complete a stream is valid) the number of streams which are completed by units other than utility exchangers is (N-V).

The other units which can complete streams (bring them to their target temperatures) are mixers and heat exchangers. Therefore, if the number of streams completed by mixers is m the number of streams completed by heat exchangers is (N-V) - m.

Each exchanger has two outputs therefore the total number of heat exchanger outputs is 2E.

The total number of outputs from heat exchangers which are not outputs from the network is given by the total number of heat exchanger outputs minus the number of streams completed by heat exchangers: $2E - [N-V-m] = 2E - N + V + m$.

Therefore using the above results:

The number of equations used in the iterative solution procedure is:

$$3E + m$$

The number of variables used in the procedure is:

$$(N-V) + E + (2E-N+V+m) = 3E + m$$

Therefore the choice of variables and equations is consistent as for <u>any</u> network the number of variables is equal to the number of equations.

All the other network variables may be calculated directly from the variables used in the iterative procedure.

All these procedures, the identification and formulation of the variables and equations to pass to the solution procedure and the formulation of the equations to calculate the other network variables, are carried out automatically by the program for any network specified by the process matrix.

5.2.3.3 <u>Equation solving routine used</u>

The equations and variables requiring an iterative solution are passed to a library routine for solution.

The routine used is an unpublished implementation of a modified form of Broyden's method as described by Perkins (1979). In this method the initial approximation to the Jacobian matrix, the inverse of which is required in the solution procedure, is found by finite

difference techniques. This enables badly posed problems to be identified as they give rise to singular initial Jacobian matrices. Singular Jacobians may also arise because of unlucky initial guesses of the variables but this is extremely unlikely if 'sensible' guesses are used.

If the Jacobian is singular it means the specifications cannot be met and this will mean in practice:

1) The problem is badly posed because of the network structure chosen and the rules developed in Chapter 3 to prevent this should be applied.

or

2) The choice of optimizing split fractions has made the problem badly posed, therefore another choice should be tried.

The equation solving routine should find a solution to the describing equations if they are consistent. As has been discussed previously this solution may or may not represent a physically possible operating state. In the design procedure infeasible solutions are equally as important as feasible solutions and care must be taken in formulating the unit models used by the modelling program to ensure that infeasible solutions can be found.

5.2.4 Modelling of infeasible operation

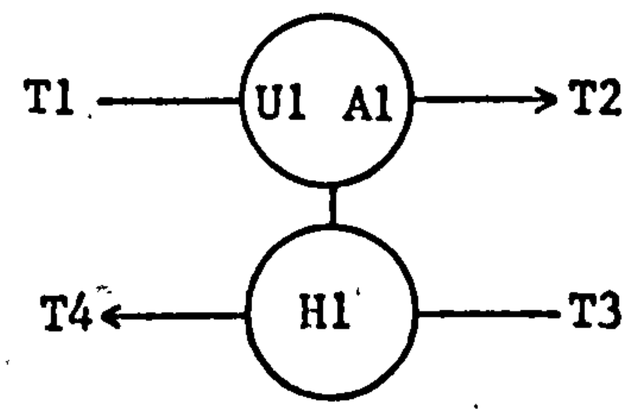
The splitter and mixer equations used can deal with split fractions which are negative or greater than one and still produce mathematically meaningful results.

The main problem arises in the heat exchanger modelling.

Exchanger Models

In the present work all heat exchangers are assumed to be countercurrent, single shell pass and single tube pass, although the design method will allow for any exchanger model to be used.

In addition to a heat balance on the hot and cold stream of each exchanger the following exchanger modelling equation is used:



$$\frac{H1}{U1.A1} = \Delta T_{LM} = \frac{(T1-T4) - (T2-T3)}{\ln\left(\frac{T1-T4}{T2-T3}\right)} \qquad (5.37)$$

This is the standard log-mean temperature difference relationship.

In the solution procedure, however, there is no restriction on the values which the temperatures T1, T2, T3 and T4 can take. For some cases ΔT_{LM} cannot be calculated from the above equation and a modified formulation must be used to give a numerical solution. Cases in which equation (5.37) is inappropriate and the way in which it is modified in the present work are given in Appendix B.

The condition is also introduced that if any heat exchanger heat load reaches any value sufficiently close to zero, i.e. $|\text{Heat Load}| < \varepsilon$; ε small, then the hot and cold stream heat balances must be satisfied but the overall (log-mean) heat balance need not. This is simply to speed up the solution procedure which can be slow if very small heat loads must be satisfied exactly. A small heat load, in practice and theory, is negligible. In the present work ε is set so that exchanger heat loads causing hot and cold process streams to change in temperature by $<1^{o}$ are treated as being negligible.

Because of these modifications the program for equation solving and network modelling is robust and for any particular network will find a solution for almost all relevant choices of inlet variables. But, obviously, not all solutions found will be physically possible, or, in the case of utility use, acceptable. For example if a temperature crossover is present (in the above network T4>T1) a

solution based on the modified ΔT_{LM} will be returned but the solution is physically impossible. Checks need to be made to determine whether a solution represents feasible operation.

5.3 Constraints

To determine whether a solution is feasible values of the constraint functions (physical and imposed) governing the network, as set out in Chapter 4, are calculated.

If one (or more) of the constraint function values is infeasible and the degrees of freedom are greater than zero then it is possible that by manipulating the optimizing control split fractions a feasible solution will be found.

5.3.1 Modelling optimizing control

A set of split fractions suitable for optimizing control is determined during the first calls to the equation solving routine. If, for any particular set of inlet conditions, one (or more) of the constraint function values is infeasible then the optimizing split fractions must be manipulated, keeping the inlet conditions constant, in an attempt to find a feasible operating point.

The program uses the NAG Library routine EØ4UAF, a quasi-Newton method which does not require analytic derivatives, to manipulate the optimizing split fractions in order to minimize total utility use while attempting to satisfy all the network constraints. The optimizing split fraction values are bounded to be between 0.0 and 1.0. The minimization can be controlled to make the finding of a feasible point more important than minimization of the total utility use. This is acceptable because one of the constraints which must be satisfied is a heat recovery target constraint.

It would have been possible to combine the equation solving and optimization using routine EØ4UAF by making the minimization subject not only to the network inequality constraints but also to the network's describing equations formulated as equality constraints. By

separating the equation solving and optimization steps a number of advantages result:

1) Advantage is taken of the fact that the equation solving routine is highly reliable whereas the optimization routine EØ4UAF is less reliable and often needs 'fine tuning' to find a solution.
2) There is better monitoring of the progress of the procedures as the describing equations are satisfied at every step of the optimization.
3) In cases where there are zero degrees of freedom the requirement is solely for an equation solving routine and it is better to use a simple equation solver than an optimizing routine.

5.4 Use of the Program for Cornerpoint Checking

The program, as it has been described so far, can now be used, for any given network, to automatically check every cornerpoint of a specified operating range for feasible operation.

A flowchart of the program used in this way is given in Figure 5.3. It will be seen that a number of refinements have been included:

1) Direct guesses of the unknown variables need only be made for the first cornerpoint checked. For subsequent cornerpoints the previous solution is used as the first estimate as it should be close to the new solution if only a few inlet variables are changed at the new cornerpoint.
2) Checks to ensure that the problem is not badly posed need only be made for the first cornerpoint. The choice of which split fractions to use as optimizing controllers also needs to be made only once.

The operation of the program for an example problem is demonstrated in Appendix C.

A listing of the program HERON used for cornerpoint checking is included on microfiche as Appendix F.

5.5 Maximization of Constraint Functions

The program as described can be used to model the behaviour of a network for any particular set of inlet conditions. Checking every cornerpoint is time consuming and, as has been discussed in Chapter 2, does not necessarily guarantee feasible operation at intermediate points. The mathematical formulation presented in Chapter 4, for networks with zero degrees of freedom, can be used to search for infeasible operating points throughout the operating range. This procedure is implemented in the computer program for the case when a network has zero degrees of freedom. Consideration only of networks with zero degrees of freedom is not as restrictive as might be thought and does bring some positive advantages as will be demonstrated in Chapter 7.

The procedure used to maximize individual constraint function values is NAG routine EØ4JAF, a quasi-Newton procedure for finding the maximum of a function of several variables subject only to fixed upper and lower bounds on the variables. In the program the function to be maximized is any selected network constraint function and the variables are the process stream inlet temperatures and heat capacity flowrates which are bounded by their upper and lower limits specified in the problem data. The maximization procedure, as it is subject only to bounds on the variables, is very reliable. For each successive set of inlet conditions generated by the optimizing routine the network modelling procedures described in the previous sections are applied and new values of all the constraint functions calculated. If at any stage any constraint function value, not just the one being maximized, is found to be infeasible then the set of inlet conditions giving rise to infeasible operation is reported. Also the inlet conditions which give rise to the highest value found during the search of the constraint function being maximized are reported, covering the rare case in which they are not the final values returned by the maximization routine, for

example due to multiple local maxima or failure in the maximization routine.

A starting point for the optimizing routine, that is an initial set of inlet conditions, can be provided in a number of ways. The heat capacity flowrates and inlet temperatures may all be set to their upper limits, to their lower limits or to values mid-way between their upper and lower limits. Particular starting values may be supplied as input data or, if a number of constraints are being maximized in turn, the starting values for the maximization may be the values which maximized the previous constraint function considered. As the global maximum constraint values are required for the entire specified operating range each constraint function should be maximized a number of times from different starting points if it is suspected that local maxima are present.

The program can be used to maximize each network constraint function in turn. If no infeasible constraint values are found then the network under consideration can operate over the full operating range. A flowchart of the program's operation in this way is given in Figure 5.4.

A number of features are not included in this flowchart but are included in the program.

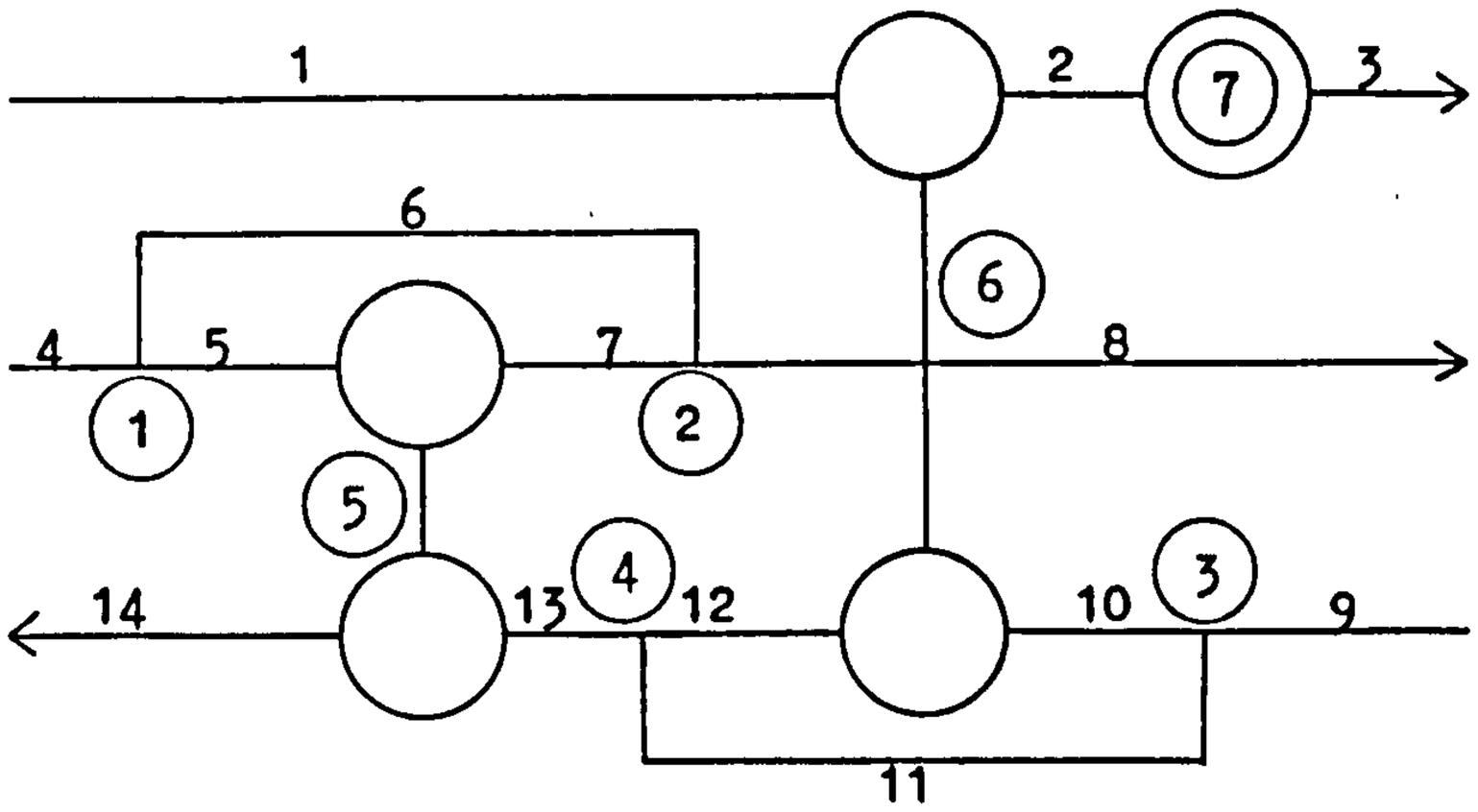
1) Badly posed problems are detected as in the cornerpoint checking procedure given in Figure 5.3.
2) Not all heat capacity flowrates and inlet temperatures need be variable. If the lower and upper limits specified for an inlet variable are equal then the program detects this and does not pass that variable to the optimizing routine.
3) The physical constraint applied to split fraction values is that they must be between 0.0 and 1.0. Therefore it is possible to both maximize or minimize each split fraction value by an appropriate selection of the form of the constraint function relating to split fractions.

4) In both the cornerpoint checking and constraint maximization procedures redundant utility exchangers as described in Appendix A may be automatically removed from a network or may be retained according to user choice.
5) All variable, equation and constraint values used in the equation solving and optimizing routines are scaled either by appropriate choice of units used for the input data (e.g.F.Cp values should be ~ 1.0) or by automatic scaling within the program. Scaling makes the numerical methods more reliable.
6) Various levels of output are available within the program depending on its mode of operation and the amount of detail required.

An example of the use of the program for constraint function maximization is given in Appendix D. A listing of the program HERON1 used for constraint maximization is included on microfiche as Appendix G.

Referring back to Figure 4.1 it will be seen that the program automatically carries out the stages of the search for the worst value of each constraint function in turn and detects sets of inlet conditions which produce infeasible operation.

Using the modelling program a number of case studies can be investigated which will clarify the other procedures identified in Chapter 4 and in Figure 4.1.



A simple network with the substreams numbered in order from input to output and units numbered (n)

Process matrix required by program

7,14	Number of units , Number of substreams
1,4,0,5,6 2,6,7,8,0 3,9,0,10,11 4,11,12,13,0 5,5,13,7,14 6,1,10,2,12 7,2,0,3,0	One line for each unit of the form : (Unit number , Number of the substream which is the first input stream to the unit , No. of second input to unit , No. first output from unit , No. second output from unit) "Missing" streams are indicated by "0"

Figure 5.1 The process matrix representation of a simple network.

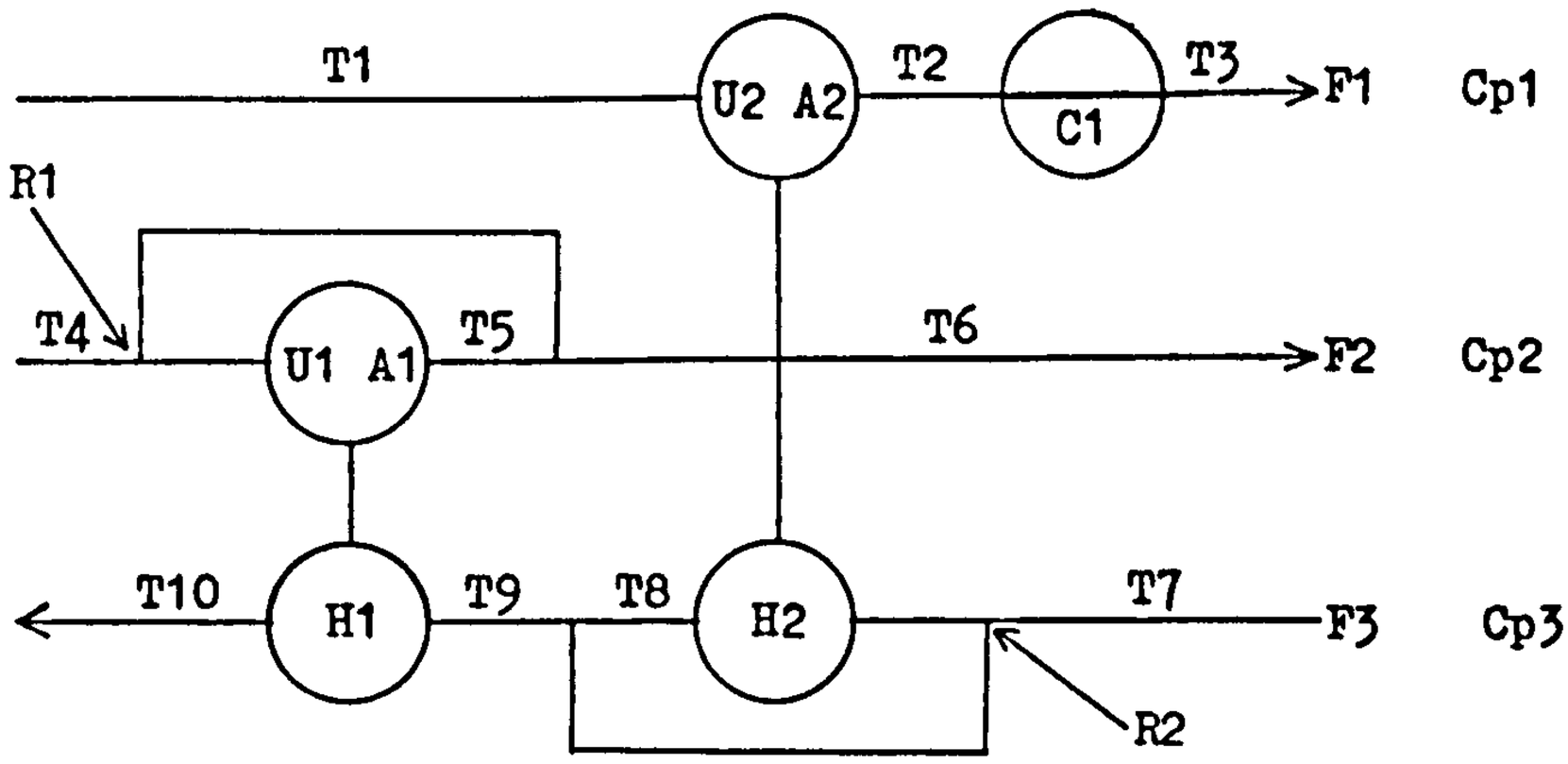


Figure 5.2 The usual representation of the network given in Figure 5.1.

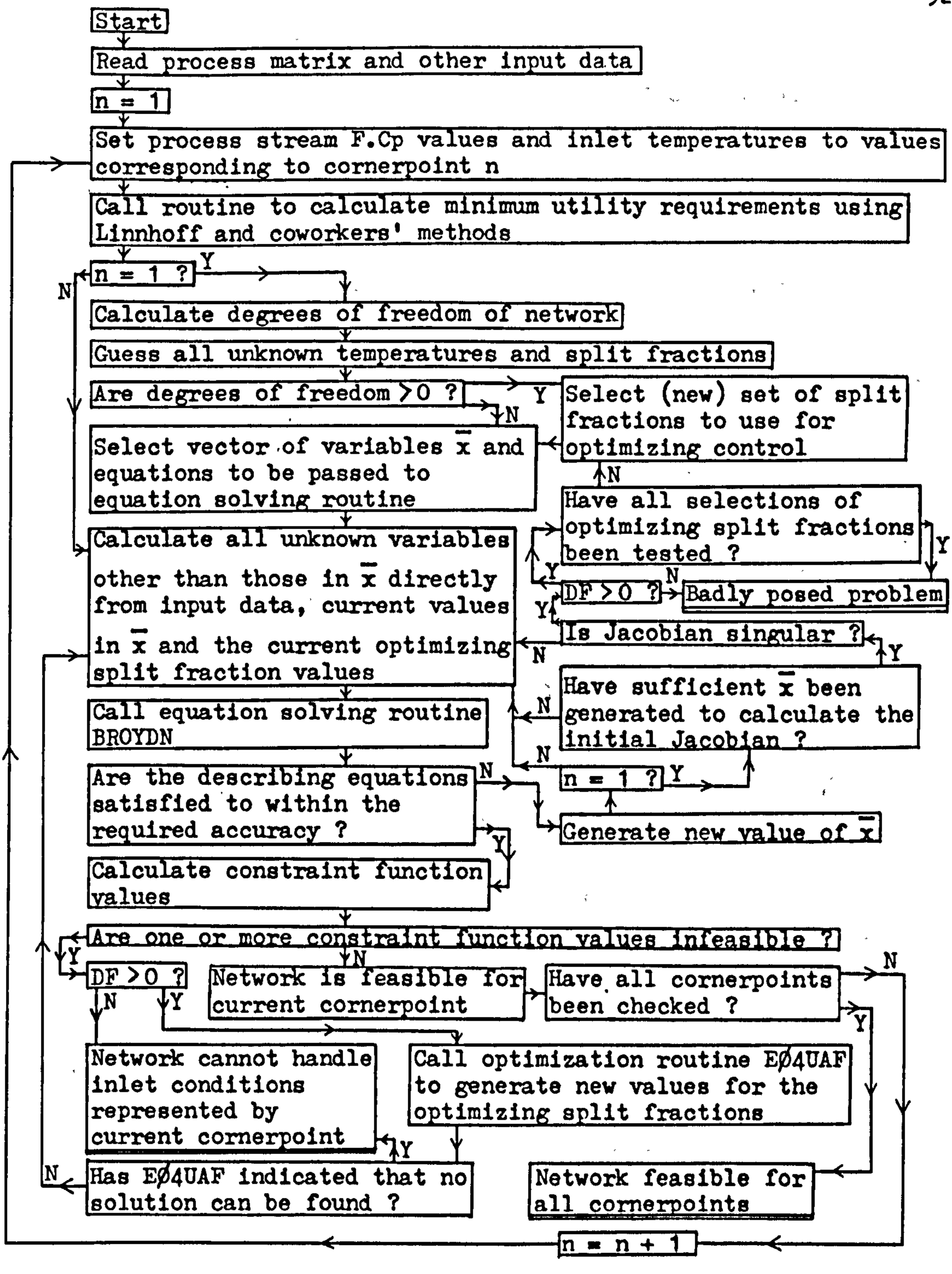


Figure 5.3 Use of the program for cornerpoint checking.

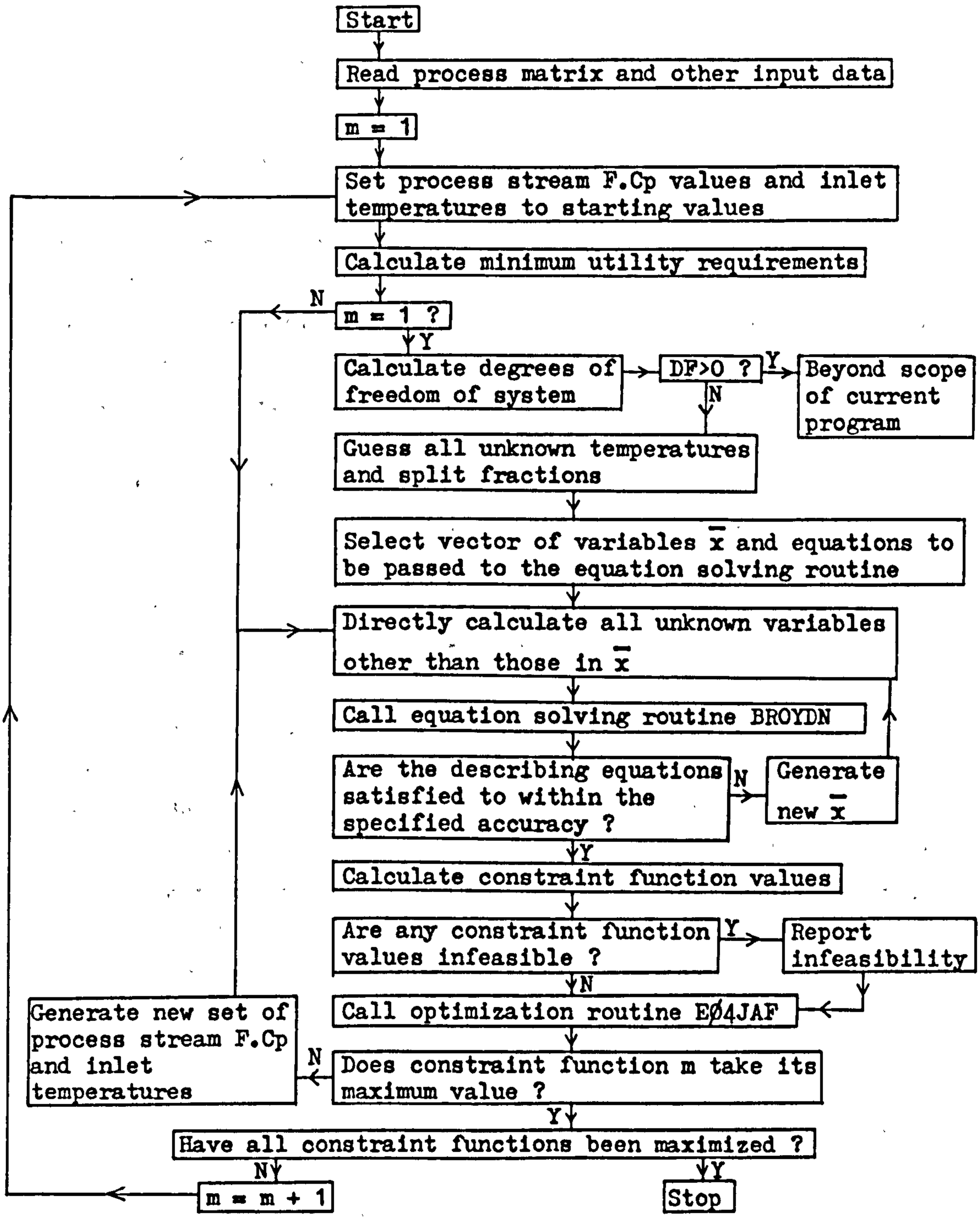


Figure 5.4 Use of the program for maximizing each network constraint function value.

C H A P T E R 6

INVESTIGATION OF THE DESIGN PROBLEM

6.1 Introduction

The computer program for network modelling described in Chapter 5 can now be used to investigate a number of example problems. Of particular importance to the design method is the form of the constraint functions for a network and how they can be used to distinguish between feasible and infeasible operation. These factors are considered in the following chapter.

6.2 Infeasible Operation

When a set of operating conditions which produces infeasible operation has been found it is important to know, in order to modify the network, whether the infeasibility arises from sizing deficiencies in the network or from structural deficiencies.

For the case of an exchanger fitted with a bypass this problem may be resolved as follows. Figure 6.1 shows the general form of such an exchanger/bypass system which may form part of a larger network in which case the inlet and target temperatures indicted may be intermediate temperatures in the larger network. The describing equations (6.1) to (6.5) which must be satisfied for feasible operation are given and it is these equations which are solved by the computer program.

6.2.1 Sizing infeasibilities

Figure 6.2(a) shows an exchanger/bypass system which is correctly sized for the particular stream conditions shown. It is important that an exchanger should not be needlessly oversized and so the exchanger area in this case, included in the U.A value of 8.80, has been calculated to just satisfy the stream conditions. This is indicated by the fact that the required exchanger split fraction is 1.0, that is all the flow is passing through the exchanger.

Figure 6.2(b) shows the same set of stream inlet and target temperatures and F.Cp values but with the exchanger oversized for the required duty; the area chosen gives U.A = 9.80 rather than the minimum required value of 8.80. To compensate for the fact that the exchanger is oversized it can be seen that a fraction of the total hot stream flowrate, 0.095, has been bypassed round the exchanger. The hot stream outlet temperature from the exchanger of 426.32 is less than that in Figure 6.2(a) of 433.33 where the exchanger was perfectly sized.

In general for a bypassed exchanger if the exchanger is oversized then the exchanger split fraction will be less than 1.0. For an oversized exchanger the exchanger split fraction will also be greater than 0.0. In the case under consideration the exchanger split fraction is always greater than 0.606 if the exchanger is oversized, Figure 6.2(c) shows this limiting case.

If the exchanger is undersized, U.A = 7.80 rather than 8.80, then the situation is as shown in Figure 6.2(d). The split fraction which is required is greater than 1.0. This is obviously not a physically possible solution but mathematically the equations governing the system, (6.1) to (6.5) in Figure 6.1, are satisfied.

The hot stream outlet temperature from the exchanger of 442.54 is greater than that for the perfectly sized exchanger in Figure 6.2(a). As the exchanger area is further reduced this outlet temperature increases. A discontinuity occurs when the required outlet temperature is equal to the inlet temperature as shown in Figure 6.2(e). If this case were encountered during computer modelling a solution could not be found as the required exchanger split fraction is infinite. Investigation would, however, show rapidly increasing values for the split fraction value at each iteration before failure.

If the exchanger is further undersized with U.A = 3.80 the performance is as shown in Figure 6.2(f). Here the hot stream temperature is required to increase on passing through the exchanger, thus

the exchanger split fraction required is less than 0.0 to give a solution to the equation (6.1) given in Figure 6.1.

From this example it can be seen that the required exchanger split fraction for an exchanger/bypass system required to meet a fixed duty varies with the U.A value as shown in Figure 6.3. If the bypass is placed on the cold side of exchanger rather than the hot side (as in Figure 6.2) then this graph has the same form with asymptotes at the same value of U.A but may have different required exchanger split fraction values for other U.A values. All the arguments presented apply equally to both cases as the fact that an exchanger is undersized or oversized is independent of the bypass placement and is dependent only on the temperatures external to any bypass (e.g. T1, T3, T5, T6 in Figure 6.1). Thus from the point of view of flexibility the choice of whether to place a bypass on the hot or cold side of an exchanger is irrelevant.

Note that for all the cases presented the heat load is greater than 0.0 and the hot and cold end temperature approaches of the exchanger are greater than 0.0. The only constraint value which is infeasible when the exchanger is undersized is the split fraction of the bypass.

A procedure for correctly sizing a bypassed exchanger in a network is:

1) Size the exchanger for the initial set of operating conditions assuming that the exchanger split fraction is 1.0.
2) Use the computer program to maximize the required exchanger split fraction with respect to all inlet variables.
3) Check at each iteration whether the exchanger split fraction takes a negative value.
4) If a maximum split fraction is found which is greater than 1.0 and no negative value has been found then set the exchanger split fraction to 1.0 and resize the exchanger for the inlet conditions producing the maximum value.

5) If a negative value has been found maximize the required exchanger split fraction, starting from the set of inlet conditions producing a negative value, to find the inlet conditions producing the maximum negative split fraction, set the value to 1.0 and resize the exchanger for these conditions.

6) If at any iteration the exchanger heat load is found to be less than 0.0 or any temperature approach (directly adjacent to the exchanger or the mixer) is less than 0.0 then a structural infeasibility exists which cannot be removed by appropriate resizing of the exchanger.

In summary two cases of infeasible split fraction values may occur. For a variable inlet F to a network, varying between Fl and Fu, the required exchanger split fraction for an exchanger/bypass system may vary as shown in Figure 6.4.

In each case (a) and (b) the most infeasible point I is indicated as that which corresponds to the inlet conditions for which the exchanger is most undersized. It is for this set of conditions that the exchanger should be resized by setting the exchanger split fraction to 1.0. In case (a) I occurs at the maximum positive split fraction, in case (b) I occurs at the maximum negative split fraction.

In general there will be more than one variable inlet condition and this will produce more complex functions but any exchanger/bypass subsystem in a network must be sized for the set of inlet conditions which produces the maximum negative required exchanger split fraction (if one exists) or the maximum positive value (if not). These maxima may not occur at cornerpoints of the operating range.

Resizing an exchanger by setting the exchanger split fraction to 1.0 ensures that this is the maximum required value for any point in the operating range, thus the exchanger is oversized just sufficiently to deal with the required operating range.

It is conceivable that synthesis methods will generate networks in which bypasses are used to balance F.Cp values, possibly to allow previously disallowed matches to occur. In this case sizing for exchanger bypass fractions of 1.0 will be inapplicable and the highest physically possible value maintaining an F.Cp balance must be used.

6.2.2 Structural infeasibilities

For an exchanger/bypass system if any constraint value other than a split fraction is infeasible then the exchanger/bypass system is structurally infeasible, that is the infeasibility cannot be removed by appropriate oversizing of that exchanger. Figure 6.5 shows four cases of structural infeasibility for an exchanger/bypass system. In none of these cases can the exchanger be sized correctly to handle the required duty. In each case setting the exchanger split fraction to 1.0 would produce a negative temperature approach at one or both ends of the exchanger thus making calculation of the exchanger area using the standard log-mean temperature difference, ΔT_{LM}, impossible. The modelling program has, however, found a mathematical solution for all these cases using the modified ΔT_{LM} formulation described in Appendix B. In Figure 6.5(b) it can be seen that the temperature approaches immediately adjacent to the exchanger are positive. The split fraction is greater than 1.0 and it may be thought that this represents a sizing infeasibility. However, the required temperature approach beyond the mixer is negative and this prevents correct sizing of the exchanger to handle the duty as the exchanger split fraction may not be set to less than or equal to 1.0 as this produces a negative temperature approach adjacent to the exchanger.

In summary, for an exchanger/bypass system in a network, if a set of inlet conditions requires an infeasible split fraction with all other constraint values feasible then the exchanger may be resized to handle the required duty. If a set of inlet conditions require negative heat loads or temperature approaches adjacent to the exchanger or mixer then a structural infeasibility exists and the design of the entire network must be reconsidered for this set of inlet conditions.

In many unpinched problems the networks feature the minimum number of units and, as has been shown in Chapter 3, if this is the case then all exchangers must be bypassed. Maximizing each exchanger split fraction in turn for a network of sized exchangers directly detects the sets of inlet conditions for which the exchangers must take their maximum areas. In a method based on checking cornerpoints of the operating range finding the maximum required exchanger areas for such a network would require calculation of all exchanger areas for each cornerpoint. It can be seen that the method developed in the present work is much more efficient and, as will be demonstrated, detects cases for which the inlet conditions requiring maximum exchanger areas do not correspond to a cornerpoint of the operating range. These advantages arise because the required exchanger area varies in a predictable way with the required exchanger split fraction so that the inlet conditions for which an exchanger is most undersized are also the conditions for which the required exchanger split fraction can be found to be most infeasible.

6.2.3 Unbypassed exchangers and stream-splits

Networks which have more than the minimum number of units (and which, therefore, will contain unbypassed exchangers if the degrees of freedom of the network are zero) or include stream-splits have design flexibility as described by Linnhoff and Flower (1978). The concept of design flexibility is completely different to the concept of flexibility as used in the present work. Design flexibility means that a network may satisfy the required duty for a range of exchanger areas. The designer is free to choose exchanger areas according to some criteria in addition to the requirement that the network should satisfy the required duty. Usually the additional criteria would be to minimize the total exchanger cost. Different choices of areas give rise to different intermediate conditions (temperatures, split fractions) and therefore it may be possible to remove temperature crossovers and negative heat loads by making a different choice of areas. Similarly exchanger/bypass systems which are shown to be size infeasible may be made feasible by resizing exchangers other than the one directly involved in the exchanger/bypass although to produce a

good rather than optimum design resizing only the exchanger in the exchanger/bypass system may be adequate. In terms of Figures 6.2 and 6.5 the infeasibility of an exchanger/bypass system may be removed by manipulating the required intermediate temperatures and hence the required exchanger duty by altering exchanger areas elsewhere in the network.

A minimum number of units network with no stream-splits has no design flexibility, intermediate conditions are fixed for any particular set of inlet conditions and only one choice of areas is possible.

A particular case of design flexibility is a network containing a stream-split, that is a split process stream having one (or more) heat exchangers on each branch as opposed to a bypass which has an exchanger on one branch only. The exchangers involved in a stream-split will usually have design flexiblity. The problem of determining whether a set of areas producing feasible operation can be found for this case is discussed in Chapter 8.

6.2.4 Summary

A flowsheet of the method used for distinguishing between sizing and structural infeasibilities is given in Figure 6.6.

If a structural infeasibility is found then the network structure under consideration is inadequate and must be changed by rearrangement or addition of exchanger or splitters. It may be possible to determine by inspection a change which might remove the infeasibility or it may be necessary to use a non-flexible synthesis method to redesign the network for the set of conditions which produce infeasible operation, either by providing an entirely new network to consider or by forming a combined network as described in Chapter 7.

If a sizing infeasibility of type (1) is found then this may be removed by oversizing the appropriate bypassed exchanger. If a sizing infeasibility of type (2) is found then this may be removed by simultaneously resizing all the exchangers in the network. This can occur

when design flexibility is present in a network and the resizing problem is discussed in Chapter 8.

6.3 Form of Constraint Functions

It has been shown that exchanger bypass fractions are important in determining whether a network can operate feasibly over a given operating range of uncertain inlet conditions. The work of Grossmann and co-workers, discussed in Chapter 2, has demonstrated that if constraint functions are convex or monotonic then feasible operation at the cornerpoints of an operating range guarantees feasible operation over the full range.

A number of examples may be studied to clarify this conclusion.

6.3.1 Example Problem 6.1

A network solving literature problem 6SP1 designed by Linnhoff (1979) is given in Figure 6.7. Bypasses have been assigned to ensure that the degrees of freedom of the network are zero.

To investigate the form of the constraint functions the stream 1 inlet conditions T1 and F.Cp1 are allowed to vary as follows:

$$2.6 < F.Cp1 < 2.9 \quad ; \quad 420 < T1 < 460$$

In fact as only one process stream has variable conditions Splitter 3 is not strictly needed as the inlet conditions of process streams 3 and 5 are invariant. In this type of situation the techniques of Linnhoff and Kotjabasakis (1984) are useful in determining which bypasses are needed. As the problem is unpinched (and, in fact, remains unpinched over the full operating range) only cooling utilities are needed. As this is the case then meeting the target temperatures ensures that maximum possible heat recovery is achieved.

The modelling program described in Chapter 5 was used to investigate the variation in required exchanger split fractions with the inlet variables F.Cp1 and T1. Figure 6.8 shows the variation of exchanger

split fraction 4 with F.Cpl and Tl over the given operating range and Figure 6.9 shows the variation of exchanger split fraction 2 over the same range. It can be seen that neither exchanger split fraction 4 nor 2 takes feasible values for the entire operating range. Referring to the previous section it can be seen that the operating conditions producing the most infeasible operation in both cases are when Tl and F.Cpl take their lower bounds i.e. 420 and 2.6 respectively.

Using the procedures discussed in the previous section the exchangers may be resized to handle these operating conditions; the area of exchanger 2 must be increased to give a U.A value of 28.31 instead of 12.30 and the U.A value of exchanger 4 must be increased to 0.04 from 0.03.

When this is done the variation of required exchanger split fractions for exchangers 2 and 4 both follow the general form given in Figure 6.10. It can be seen that the exchanger split fractions take feasible values for any combination of Tl and F.Cpl. As the exchanger split fractions are monotonic with F.Cpl and Tl the fact that the four cornerpoints of the operating range, marked ■, produce feasible required split fractions ensures that the entire operating range produces feasible values.

For this example all of the other network constraint functions, based on heat loads and temperature approaches, remain feasible over the full operating range and are monotonic.

Two literature problems have been identified in which a cornerpoint feasible design is not fully feasible and these are now investigated in terms of the form of their constraint functions.

6.3.2 Example Problem 6.2

The problem is based on a network given by Marselle (1980) to which a bypass has been added to ensure that the degrees of freedom are zero. The network is shown in Figure 6.11. The operating range over

which the network is required to operate is defined by variations in T1:

$$260 < T1 < 340$$

When T1 = 260 the problem is unpinched with only heating utilities required. For T1 = 340 the problem is unpinched with only cooling utilities required. Therefore the problem includes redundant utility exchangers as described in Appendix A. The target set for utility use is that the maximum amount of heat should be recovered with only the cooling utility or the heating utility exchanger (not both) used for any set of conditions.

With the exchanger sized to give U.A = 3.98 the value of the required exchanger split fraction varies with T1 as shown in Figure 6.12.

It can be seen that the required exchanger split fraction takes infeasible values for much of the range but that at the two corner-points of the operating range the required values are feasible, thus simply checking the cornerpoints is insufficient in this case.

Applying the rules developed in Section 6.2 it is clear that the value of T1 producing the most infeasible split fraction value is T1 = 303.3. At this point the network requires neither heating nor cooling utilities. The streams are in perfect heat balance. It is interesting to note that Parkinson, Liebman, Pedersen and Templeman (1982) also give an example in which setting streams in perfect heat balance provides the most severe test of heat recovery.

Resizing the exchanger for T1 = 303.3 produces a required U.A value of 8.04. The variation of the required exchanger split fraction with T1 for the resized exchanger is shown in Figure 6.13.

In this case the exchanger split fraction constraint is clearly non-convex and feasible operation for the cornerpoints does not

guarantee full feasibility. The maximization of the exchanger split fraction value, as discussed in Chapter 4, would detect the most infeasible point corresponding to T1 = 303.3 and, therefore, in this case would produce a different design to that produced by cornerpoint checking.

In this case the critical point for design purposes corresponds to a recognisable physical state in the network: the streams in perfect heat balance. This is not the case in the next problem.

6.3.3 Example Problem 6.3

The problem is given by Morari (1982b), (1983) and, using slightly modified data, by Saboo and Morari (1984). The network and stream data are given in Figure 6.14(a). The only variable inlet is the F.Cp value of stream 1 which varies between 0.190 and 0.341.

6.3.3.1 Morari and co-workers' treatment of the problem

The procedure used by Morari and co-workers is to test the network at the two extreme points of the operating range. These are shown in Figures 6.14(b) and 6.14(c). It can be seen that for each case the network is unpinched with only cooling utilities required.

Saboo and Morari (1984) note that:

1) Different exchanger areas are required for each case
2) For economic reasons the minimum allowable temperature approach for each exchanger should be 18.

For an F.Cp value of 0.254, between the limits, the conditions are as shown in Figure 6.14(d).

A negative temperature approach exists at the cold end of exchanger 1 and therefore this exchanger cannot be sized so as to produce feasible operation at this set of operating conditions.

Morari and co-workers conclude that the network has a structural deficiency which is difficult to detect.

6.3.3.2 A new approach

Morari and co-workers' requirement that the temperature approaches, ΔT, should exceed 18 is a purely economic objective. If ΔT_{min} is less than 18 it is assumed that the required exchanger areas are larger and that extra area costs are more significant than any extra heat recovered.

To investigate the problem using the techniques developed in the present work exchanger areas need to be determined. If the critical points for network design are the cornerpoints of the operating range then the maximum required exchanger areas will occur at one or other of the extreme operating points. Bypasses must also be assigned according to the rules developed in Chapter 3. Figure 6.14(e) shows this network with each exchanger sized for the cornerpoint requiring the maximum U.A value. As in Example Problem 6.1, one inlet is variable and the techniques of Linnhoff and Kotjabasakis (1984) can be used to eliminate bypasses which are not needed.

Morari and co-workers' detection of a case in which the cold end ΔT of exchanger 1 is negative still applies and so the constraint functions relating to exchanger 1 are studied.

Figure 6.15 shows the variation of the cold end temperature approach, ΔT, of exchanger 1 with the F.Cp of stream 1. Figure 6.16 shows the variation of the hot end ΔT of exchanger 1 with F.Cp1.

The graphs demonstrate that for the extreme points of the operating range, F.Cp1 = 0.190 and 0.341, the hot and cold end ΔT values are feasible. Note, however, that for F.Cp = 0.190 the hot end ΔT is close to zero indicating that the exchanger is greatly oversized for this particular duty. The actual temperature approach is much smaller than the minimum allowable temperature approach as used by Morari and

co-workers. The implications of this feature of oversized exchangers are discussed in Chapter 7.

For some intermediate values of F.Cpl both the hot and cold end ΔT values are negative, indicating that the network is structually infeasible for those conditions. This is not revealed by simple cornerpoint checking. The procedures developed in the present work would detect this infeasibility. Referring to Figure 6.15, minimizing the cold end ΔT (or, for consistency with the mathematical development of Chapter 4, maximizing - ΔT) with respect to F.Cpl would, as the constraint function is continuous, easily detect negative ΔT values and give the F.Cpl value producing most infeasible operation, in fact this is the value F.Cpl = 0.254 used by Morari and co-workers to demonstrateinfeasible operation. This procedure is demonstrated in Appendix D. The network can now be redesigned for this particular set of conditions; a feasible network structure solving the problem is given by Saboo and Morari (1984).

A feature of this problem is that the operating conditions producing the most infeasible operation do not have any particular significance in terms of the problem table calculations, unlike Example Problem 6.2. The method implemented in the present work is the only one which can find automatically the operating point which produces the most infeasible operation.

Example Problems 6.1, 6.2 and 6.3 all consider the form of physical constraint functions. It is also necessary to consider appropriate formulations for the utility use constraint.

6.4 Utility Use Constraint

As has been noted, for networks designed for unpinched problems which require only heating utilities or only cooling utilities the meeting of target temperatures automatically ensures that maximum possible heat recovery is achieved.

For pinched problems(or unpinched problem designed not to recover the maximum amount of heat, possibly to keep the network structure simple) both heating and cooling utilities must be provided. For these cases an appropriate heat recovery constraint must be formulated to ensure that a flexible design maintains high heat recovery.

Alternative formulations of such a constraint are demonstrated in the following Example Problem.

6.4.1 Example Problem 6.4

The problem data and network used are given in Figure 6.17. To recover the maximum amount of heat possible with this network structure the exchangers must have infinite areas. This is not possible and an economic trade-off must be made between capital and operating costs. To study the utility use characteristics of the network the areas are arbitrarily set to give the U.A values shown in Figure 6.17.

For a given network three values of minimum (total) utility use may be determined for any particular set of inlet conditions:

1) Actual utility use.
   This is the utility use actually required by an operating network and can be determined directly from a network simulation. (If a network has degrees of freedom greater than zero then these must be manipulated to give minimum actual utility use while maintaining feasible operation).

2) Theoretical minimum utility use.
   This is the minimum utility requirment for the particular set of stream data and is independent of the network used. It is based on purely thermodynamic arguments and can be found using the Linnhoff problem table approach, setting the minimum allowable temperature approach to zero. For pinched problems networks cannot be designed for zero temperature approaches and therefore this quantity represents the lower bound on utility use. (The problem with calculating 'minimum' utility use for a minimum allowable temperature approach of

greater than zero is that in a flexible network exchangers will often be oversized for a particular duty and lower temperature approaches than the 'minimum' allowable may be possible, thus enabling less than the 'minimum' utility use).

3) Minimum utility use possible for a given network structure. This is calculated assuming that the exchangers in a network are sized to achieve minimum utility use. Infinite area exchangers may be used.

It follows from the definitions of these quantities that for any given network:

Theoretical minimum utility use	<	Minimum possible utility use for the given network structure	<	Actual utility use

These three quantities are plotted in Figure 6.18 for the network shown in Figure 6.17 over the given operating range.

As described in Chapter 4 an appropriate utility use target is to ensure 'low' utility use for the entire opeating range.

If the actual utility use figure were used directly to set a utility use target, by specifying that the actual utility use should be less than a particular value over the entire operating range, there would be no guarantee that the figure set would be achievable for all sets of inlet conditions. It is better to ensure that the actual utility use is within a specified tolerance of a minimum utility use calculated for each set of inlet conditions.

The value of the minimum possible utility use for the given network structure is likely to be of limited use in providing a reference level for the actual utility use as it is network dependent and therefore not suitable when designing alternative networks for the same problem. Referring actual utility use to the theoretical minimum use,

which provides a lower bound based only on the particular operating conditions (not the network), overcomes this problem.

One possible measure is to calculate the actual utility use as a percentage greater than the theoretical minimum use. For Example Problem 6.4 the variation of this figure with F.Cpl is shown in Figure 6.19.

In this case it can be seen that at the cornerpoint F.Cpl = 1.1 the actual utility use is ~ 33.3% greater than theoretical minimum use. If the utility use for this set of conditions has been set by an economic trade-off between exchanger area and utility cost in a non-flexible design scheme then as this is an economical utility use level for one set of operating conditions it is likely to be economical (but not optimum) for the full range of conditions and it can be applied to the entire operating range.

The maximum value of utility use occurs for F.Cpl = 1.02 and is 35% greater than theoretical minimum and this is the set of conditions for which the utility use constraint, if set to be less than or equal to 33.3%, is most strongly violated. When a constraint is violated the designer must judge the importance of the violation. For physical constraints the penalties in terms of unpredictable behaviour, control and process stream target temperatures if a constraint is indicated to be violated for a particular set of operating conditions will generally mean that the infeasibility must be removed by changing the network's size or structure. For a utility use constraint the designer may judge the violation to be negligible and redefine the constraint. In this case, for example, it is clear that if the utility use target is set to 33.3% greater than theoretical minimum or less and the actual maximum use is 35% greater than theoretical minimum then this difference is negligible and the network is acceptable. If the violation is greater the designer may want to examine part of the operating range around the most infeasible point to estimate the extent of the infeasible region and the frequency with which sets of operating conditions producing infeasible operations are likely to occur. The extra capital cost

involved in making the network able to handle operating conditions which occur very infrequently may be excessive. This problem is not within the scope of the present work in which designs must be fully feasible over the operating range.

Some problems have very low theoretical minimum utility use requirements for operating conditions which produce a close heat balance between hot and cold streams. In this case expressing actual utility use as a percentage greater than theoretical minimum use is not appropriate as very large values result. For this case it is better to express utility use as an absolute difference between actual use and theoretical minimum use.

Utility use may be split into hot and cold utility use or further into different levels of hot and cold utility. This would enable more complex utility targets to be set taking into account the relative costs of each level used. Actual, possible and theoretical minimum utility use can be calculated for each particular level and, as above, used in target setting.

6.5 Summary

In this section a number of useful results have been demonstrated. They may be summarized as follows:

1) For an exchanger/bypass system in a network it is possible to determine whether the exchanger is undersized for the given operating range by maximizing the required exchanger split fraction. The set of inlet conditions which produce the maximum required exchanger area can be found automatically by maximizing the required exchanger split fraction.

2) Structural infeasibilities may be detected by searching for infeasible values of the heat load and temperature approach constraint functions.

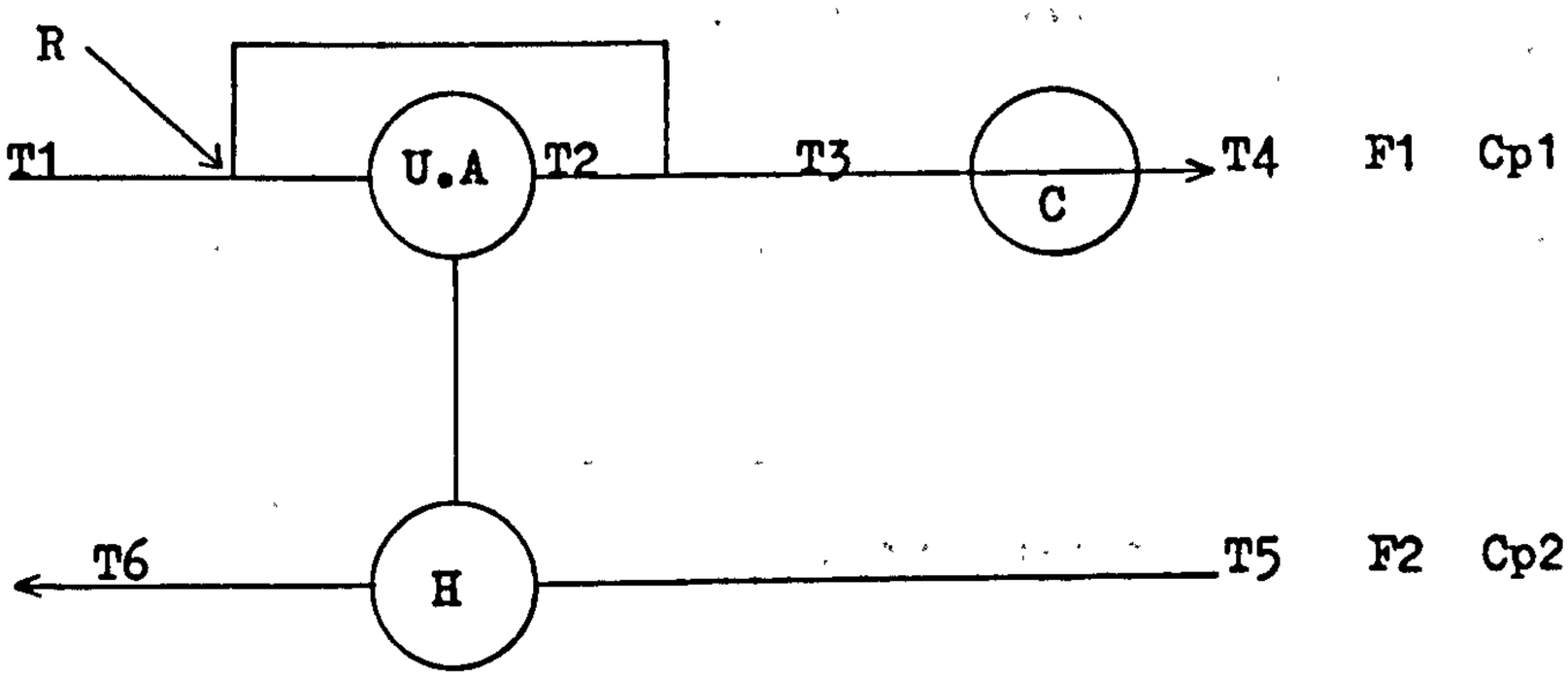
3) Case studies demonstrate that constraint functions need not be convex or monotonic therefore, from the work of Grossmann and co-workers, ensuring that a network can operate for all the cornerpoints of an operating range does not guarantee operation over the full range.

   The methods developed in the present work will automatically detect such cases, find the operating conditions producing the most infeasible constraint function values and ensure that a network is tested over the full range.

4) To ensure that a network has 'high' heat recovery over the full operating range a good utility use constraint function can be formed by relating actual utility use to the theoretical minimum utility use.

   For a network using both heating and cooling utilities the suggested formulation is to specify that the total utility required as a percentage greater than theoretical minimum use should not exceed a value which has been found to be economical for a single set of operating conditions.

Using the procedures developed it is now possible to consider a number of flexible design examples.



$$H = (T1-T2).R.F1.Cp1 \qquad (6.1)$$

$$H = (T6-T5).F2.Cp2 \qquad (6.2)$$

$$H = U.A.\frac{(T1-T6)-(T2-T5)}{\ln((T1-T6)/(T2-T5))} \qquad (6.3)$$

$$T3 = R.T2 + (1-R).T1 \qquad (6.4)$$

$$C = (T3-T4).F1.Cp1 \qquad (6.5)$$

Figure 6.1 The describing equations for an exchanger/bypass system

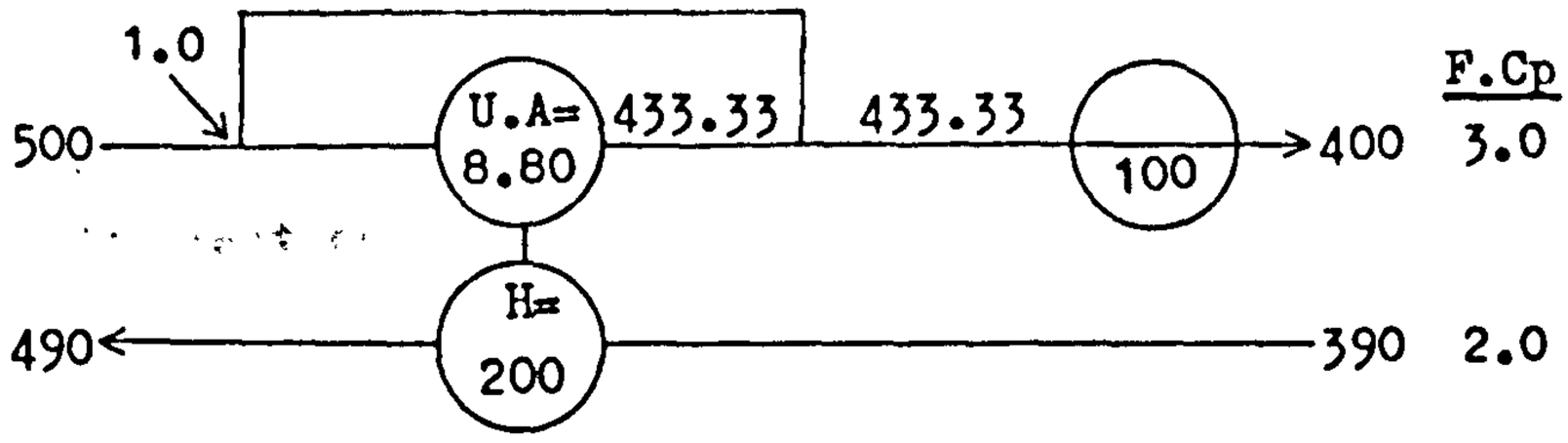


Figure 6.2 The result of using different exchanger areas in an exchanger/bypass system required to meet a fixed duty.

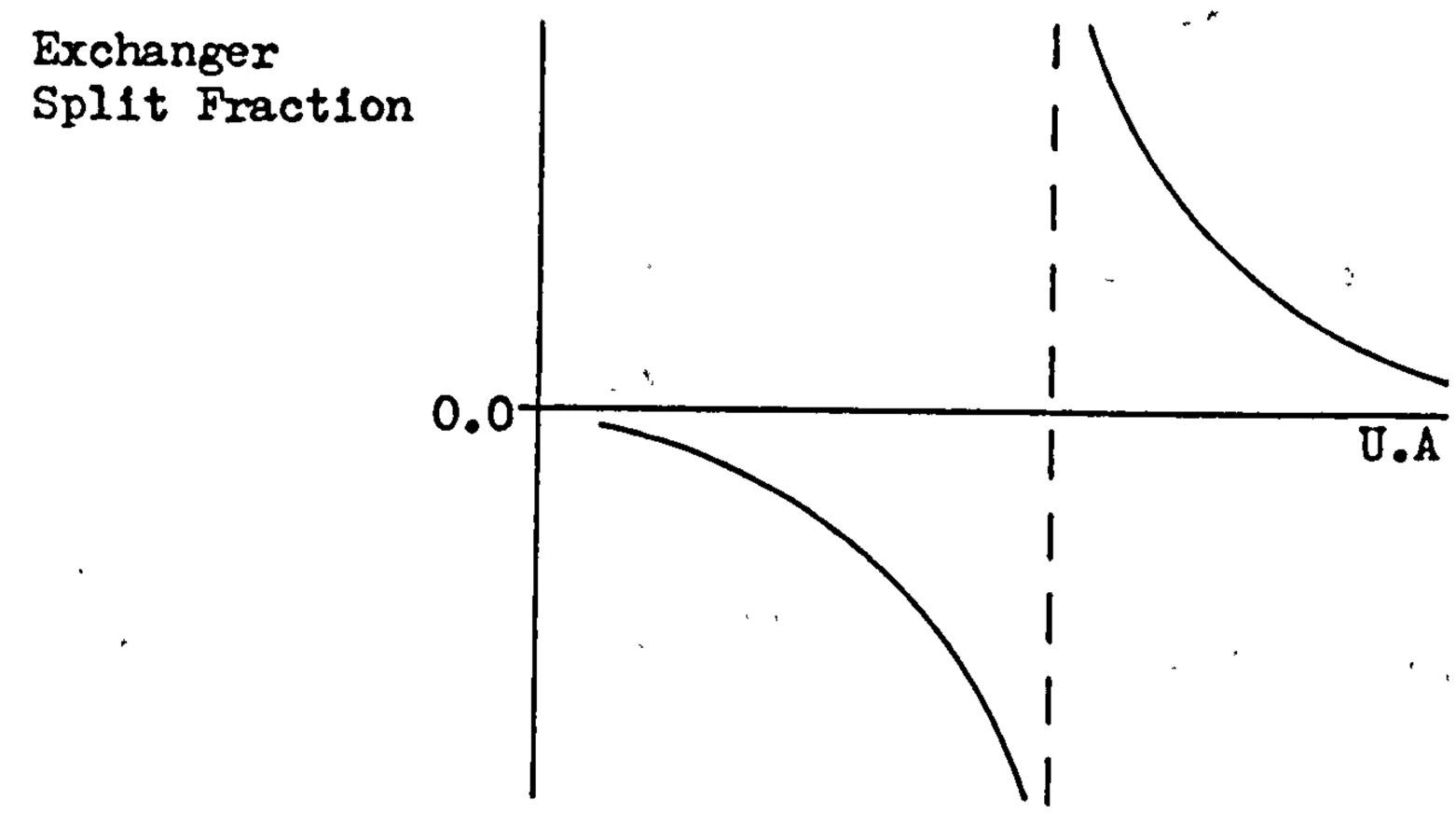


Figure 6.3 Variation of required exchanger split fraction with U.A for an exchanger/bypass system required to meet a fixed duty.

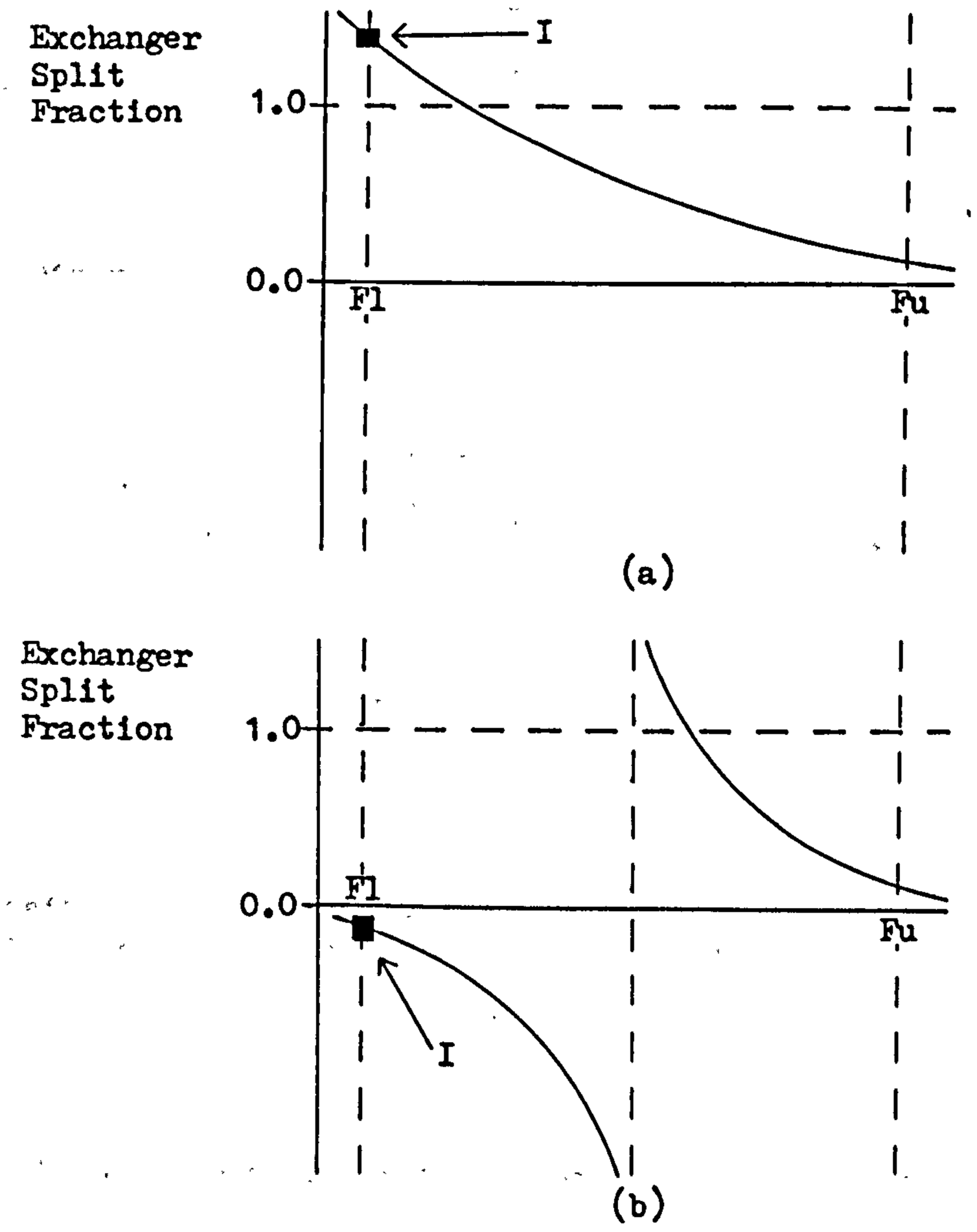
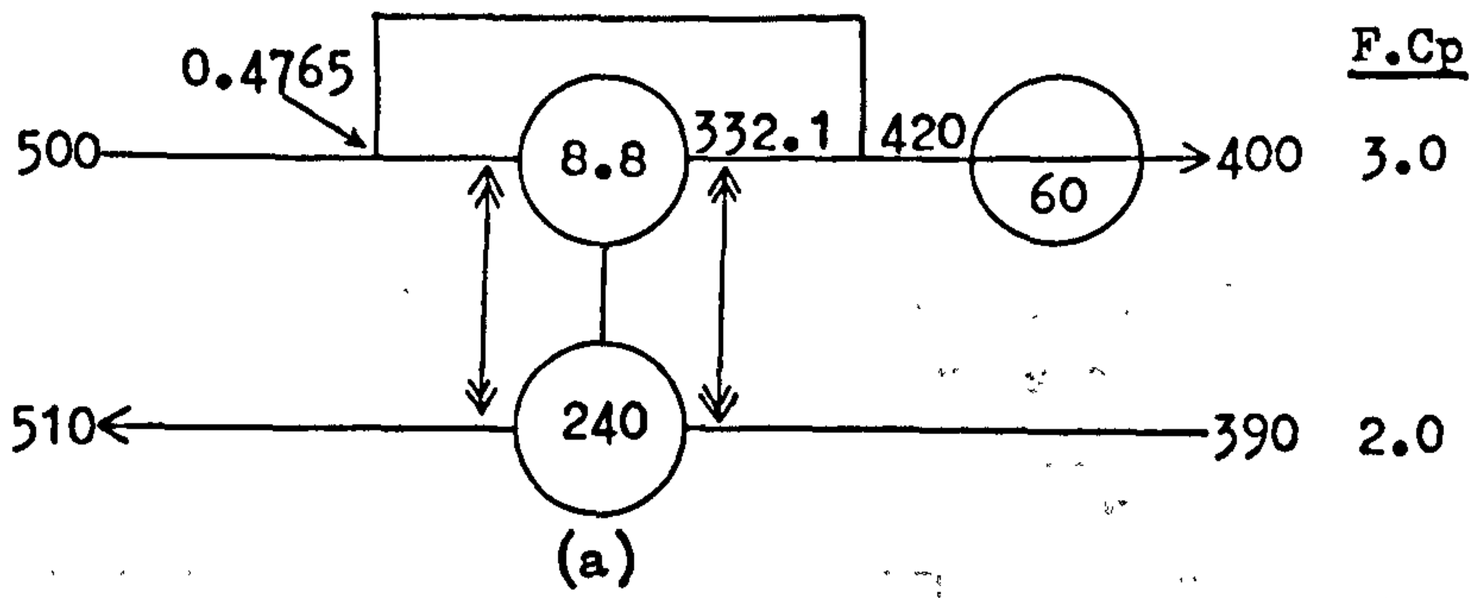


Figure 6.4 Variation of required exchanger split fraction with an inlet variable for an exchanger/bypass system in a network.



Infeasible values are marked *
Temperature difference infeasibilities are arrowed

Figure 6.5 An exchanger/bypass system showing structural infeasibilities.

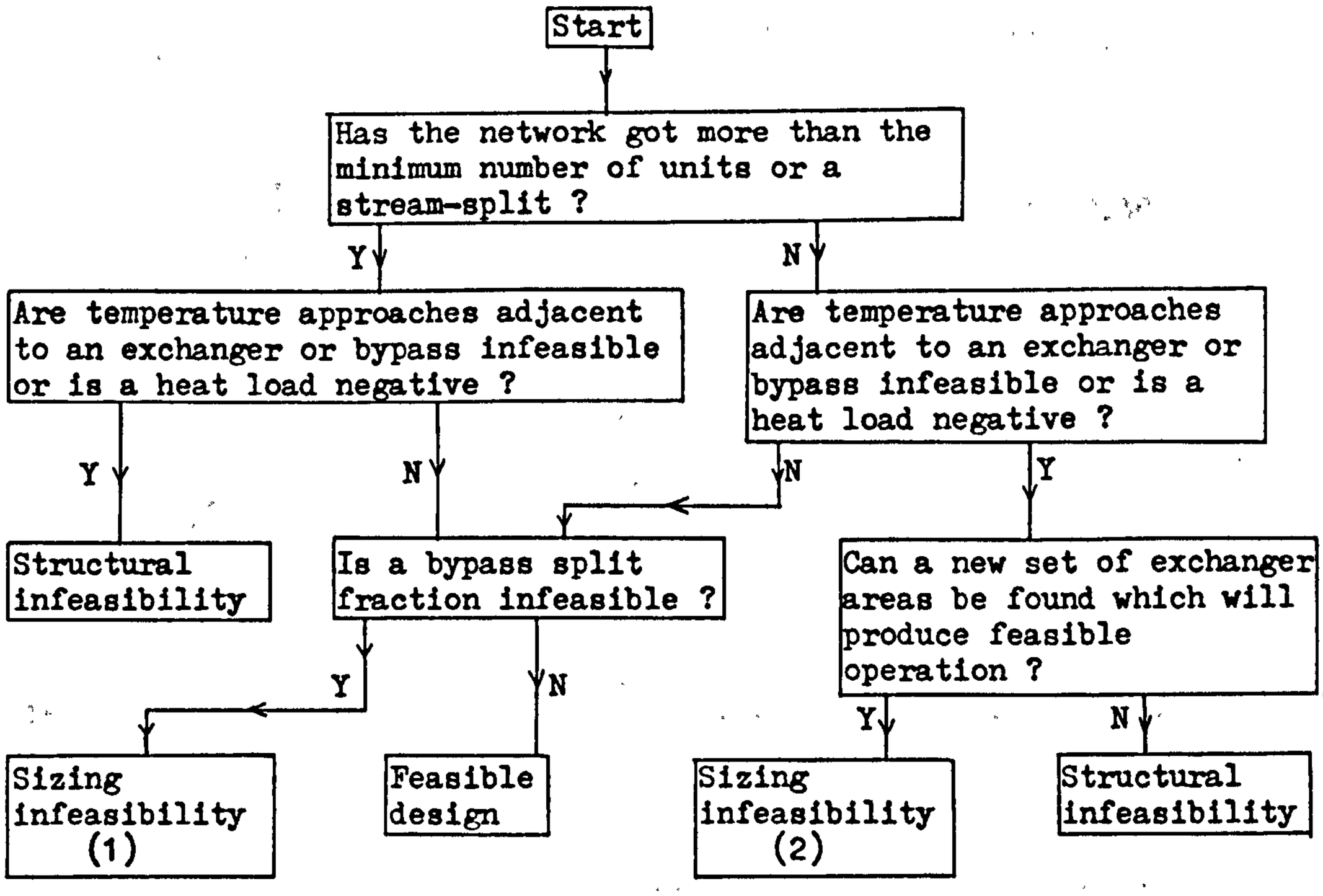


Figure 6.6 Distinguishing between sizing and structural infeasibilities.

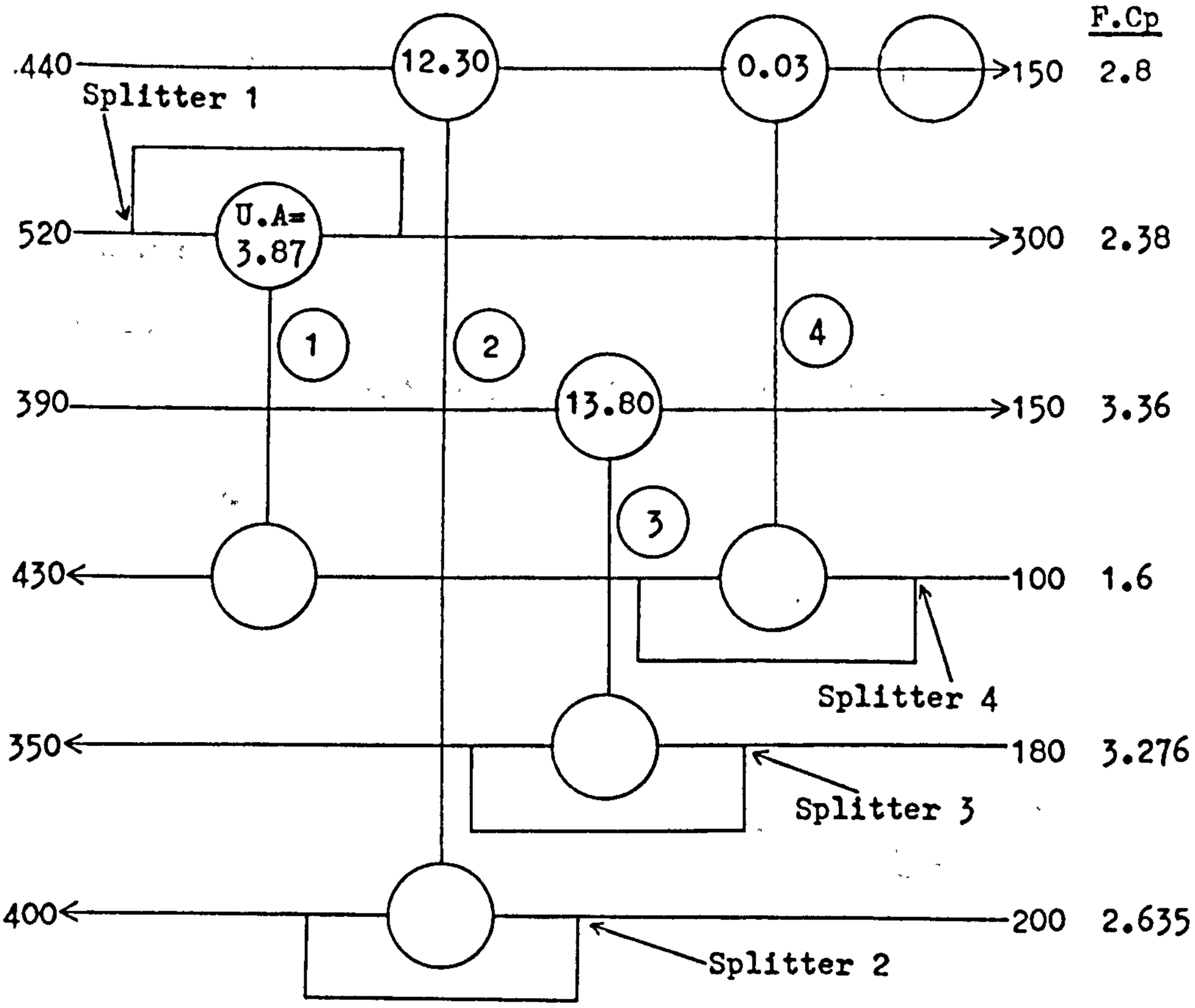


Figure 6.7 A network solving problem 6SP1 used in Example Problem 6.1.

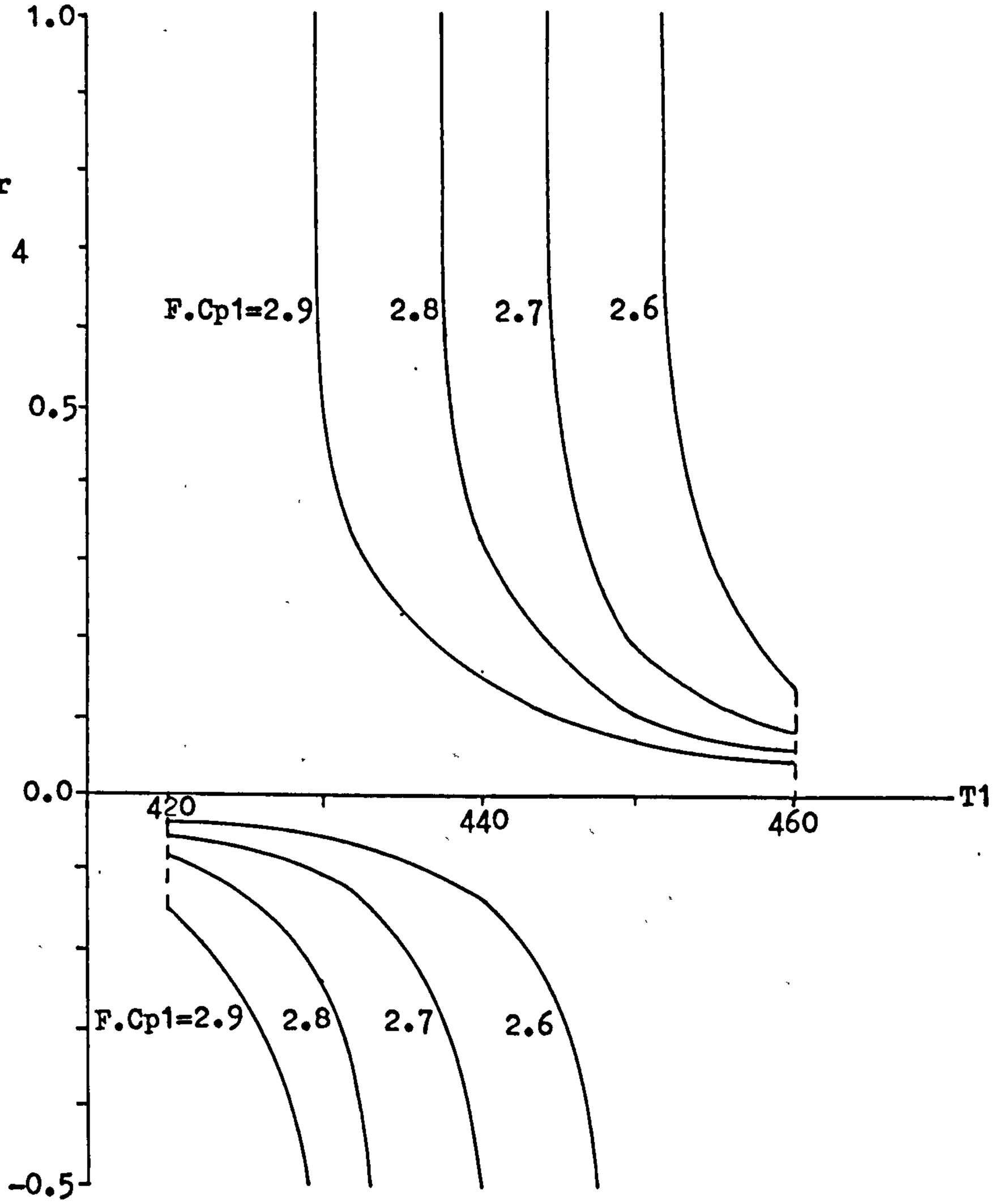


Figure 6.8 The variation of required exchanger split fraction 4 with F.Cp1 and T1 in Example Problem 6.1.

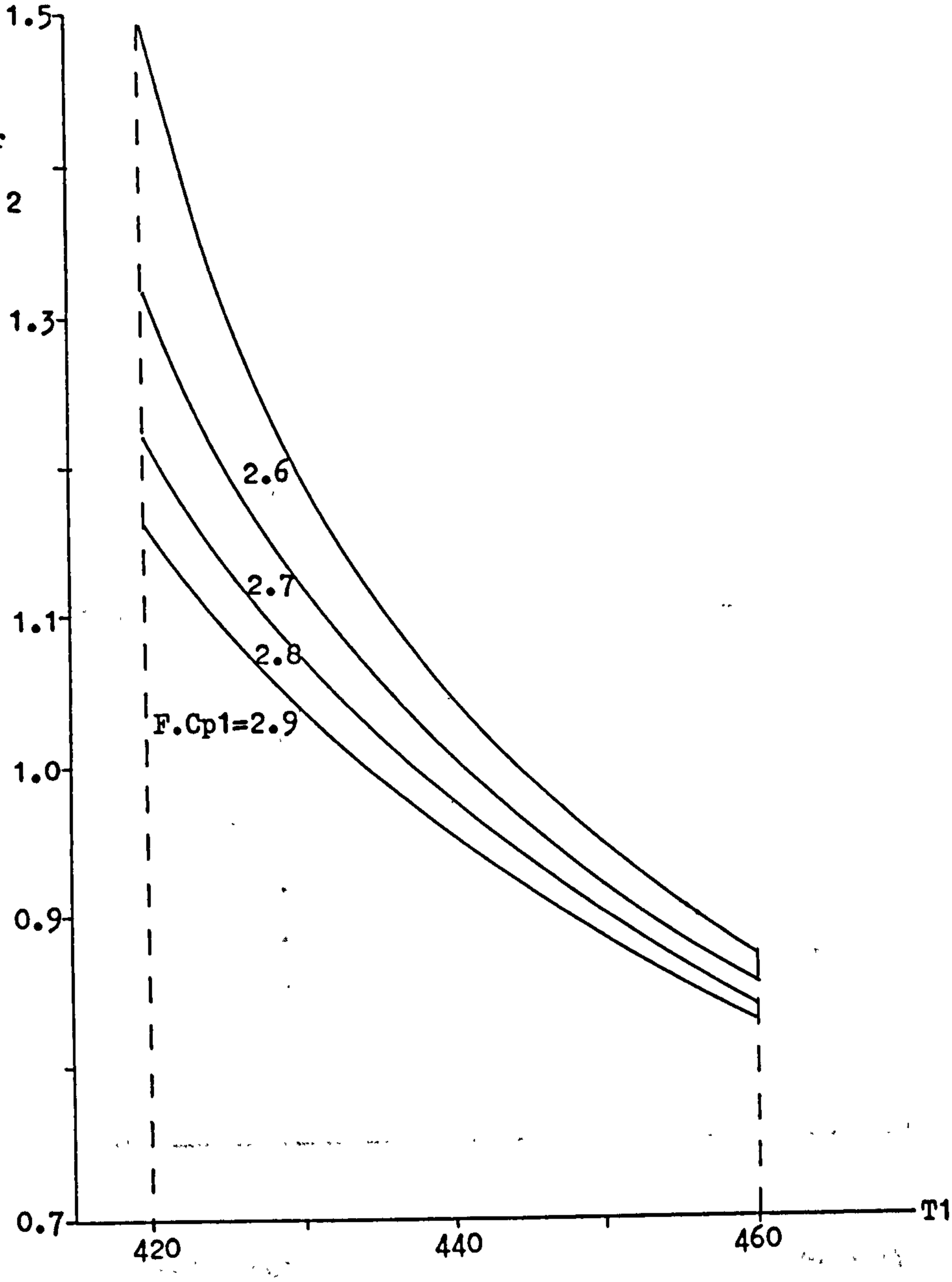


Figure 6.9 The variation of required exchanger split fraction 2 with F.Cp1 and T1 in Example Problem 6.1.

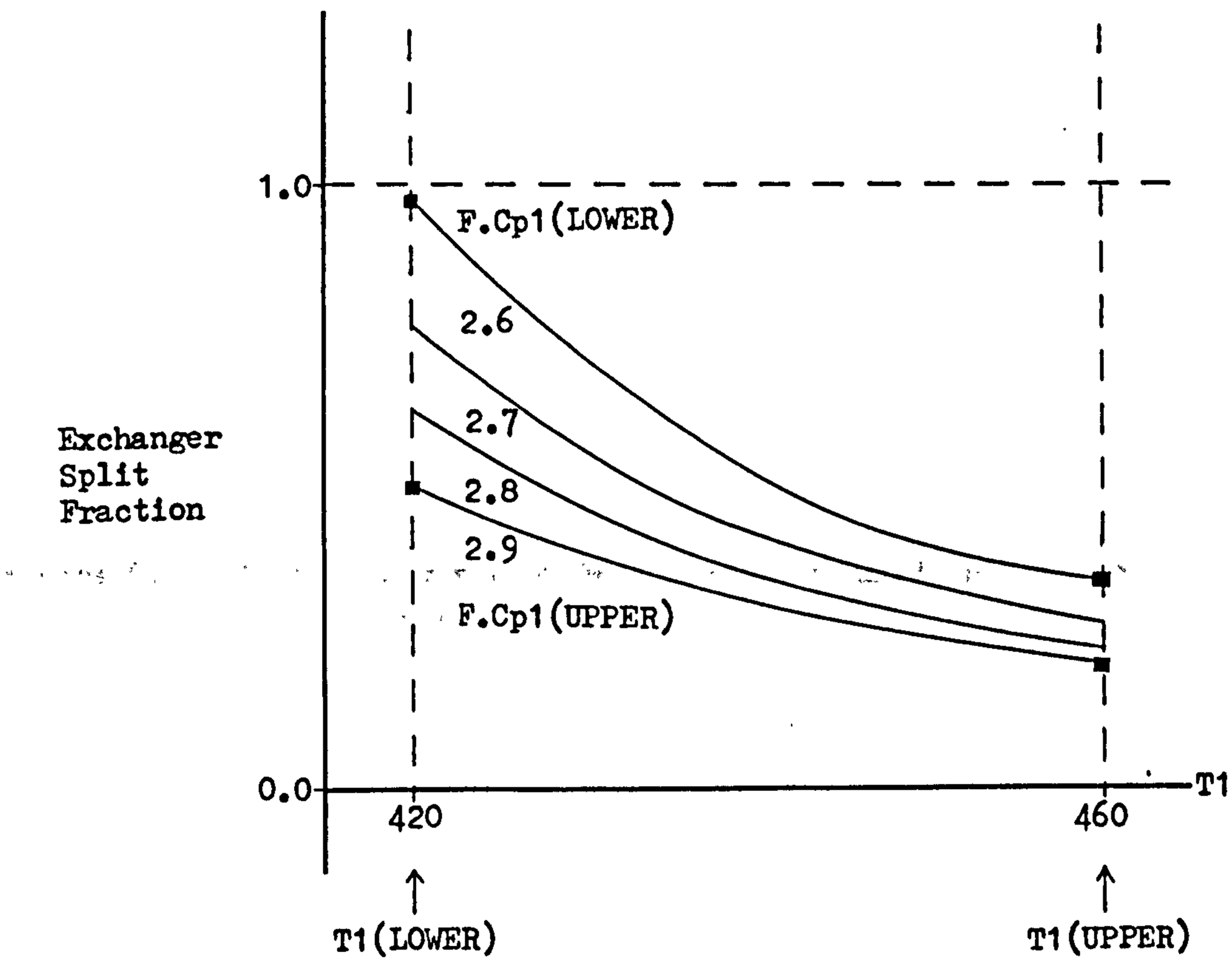


Figure 6.10 The variation of required exchanger split fraction 2 or 4 with F.Cp1 and T1 in Example Problem 6.1.

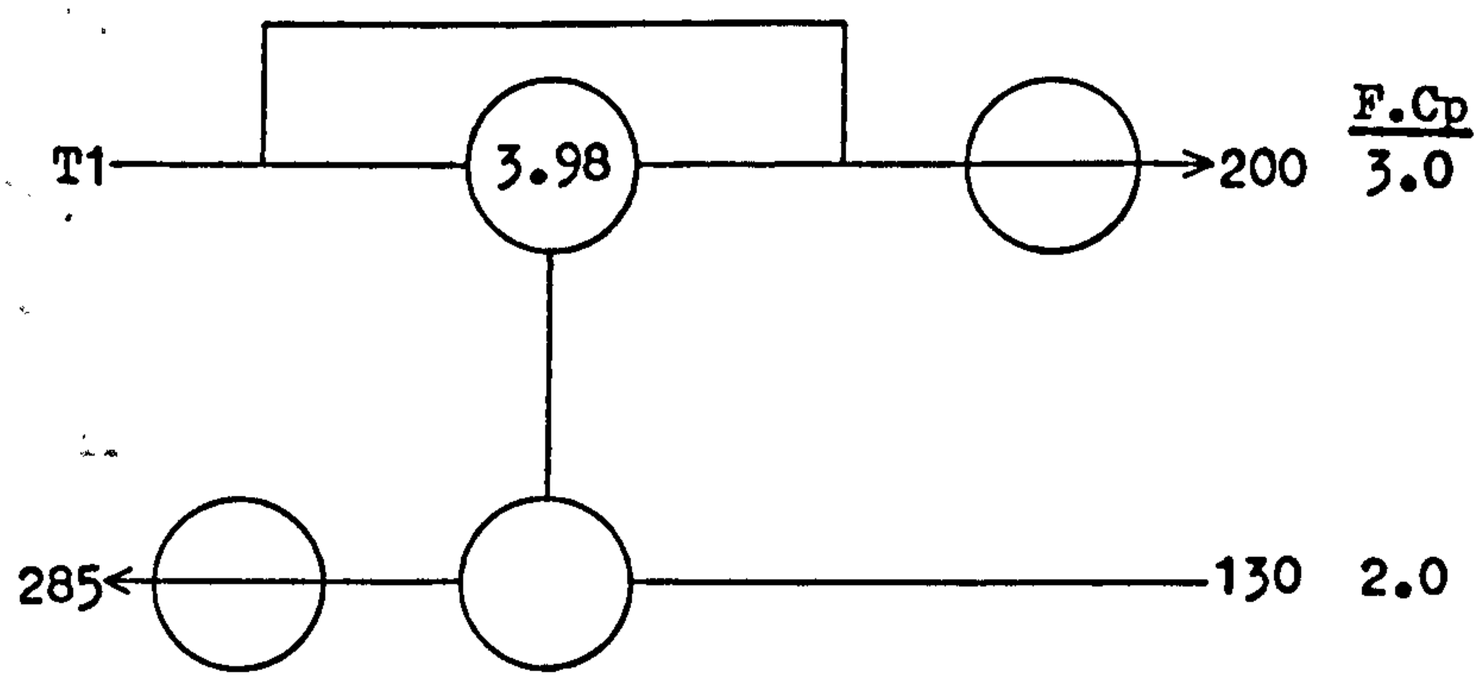


Figure 6.11 A network given by Marselle(1980) used in Example Problem 6.2.

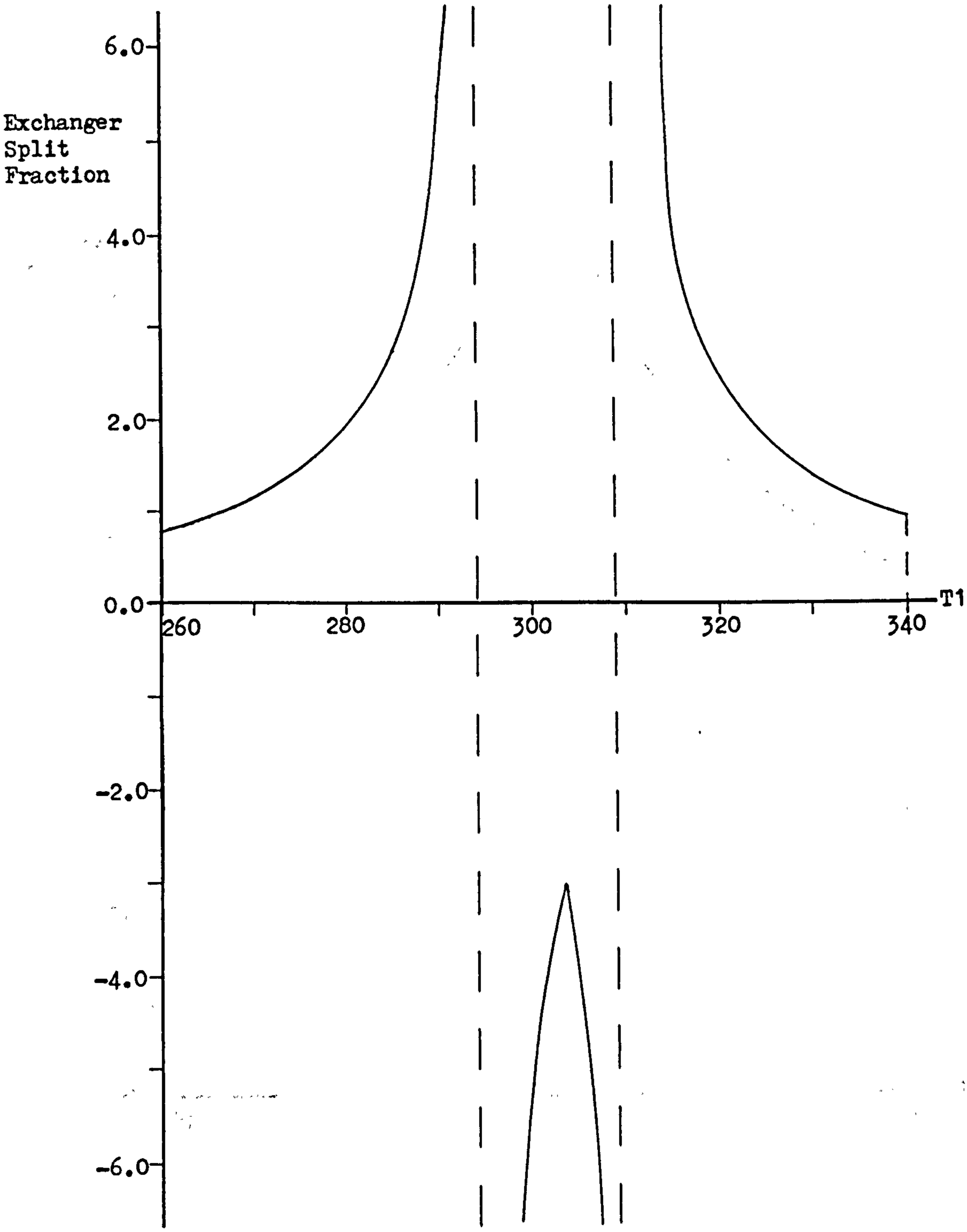


Figure 6.12 Variation of required exchanger split fraction with T1 for U.A = 3.98 in Example Problem 6.2.

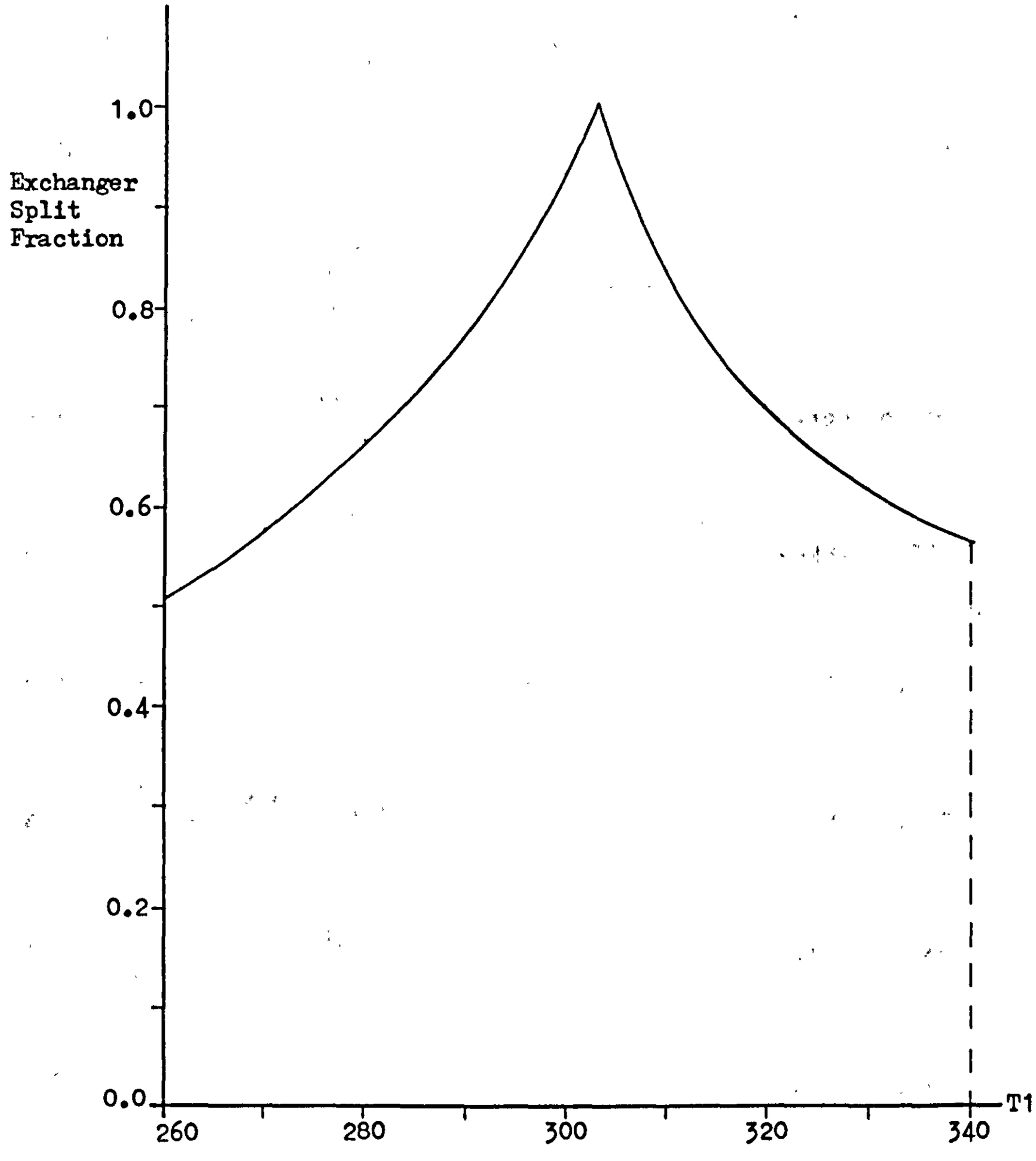


Figure 6.13 Variation of required exchanger split fraction with T1 for U.A = 8.04 in Example Problem 6.2.

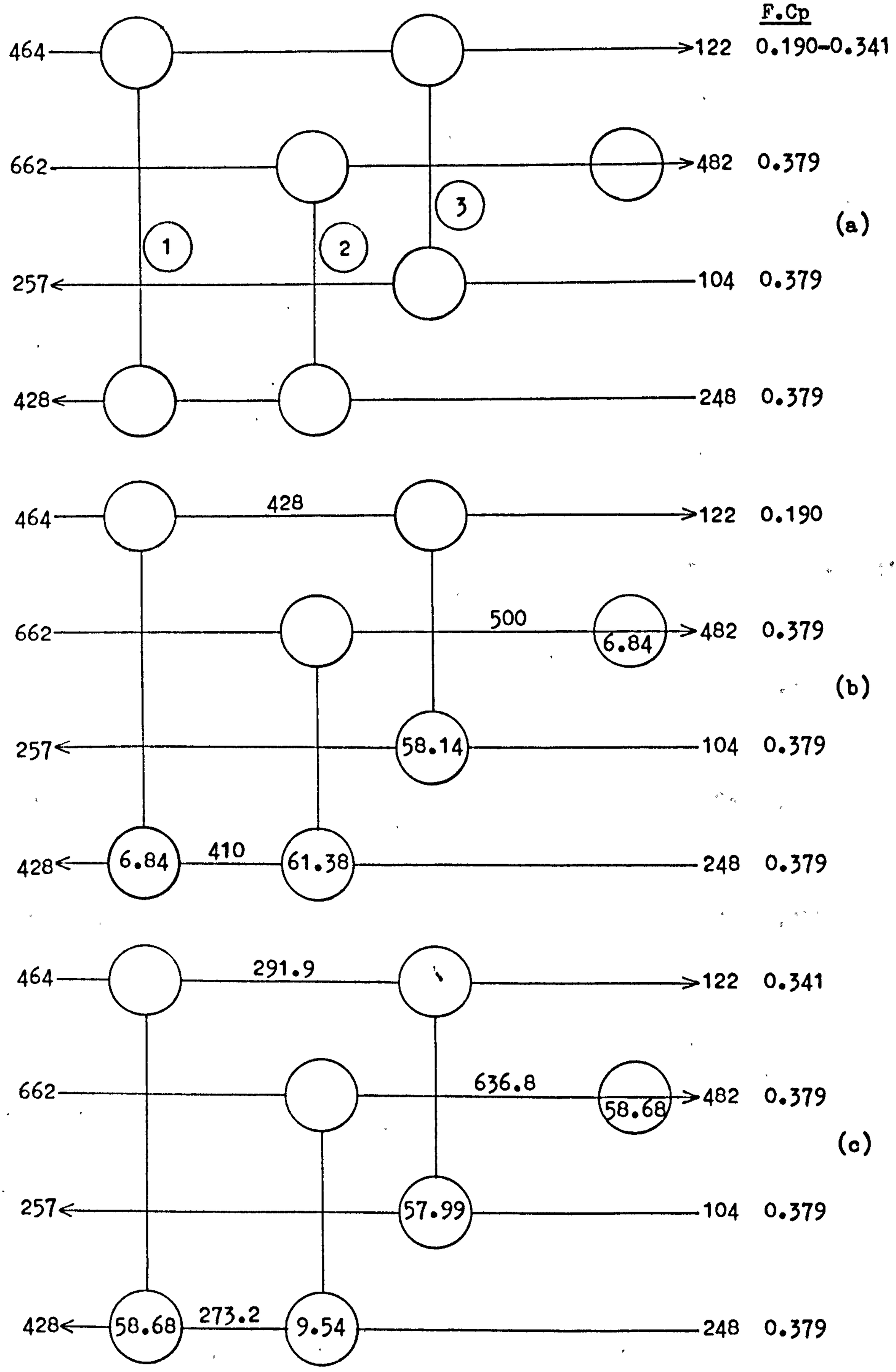


Figure 6.14 A network posing flexibility problems used in Example Problem 6.3.

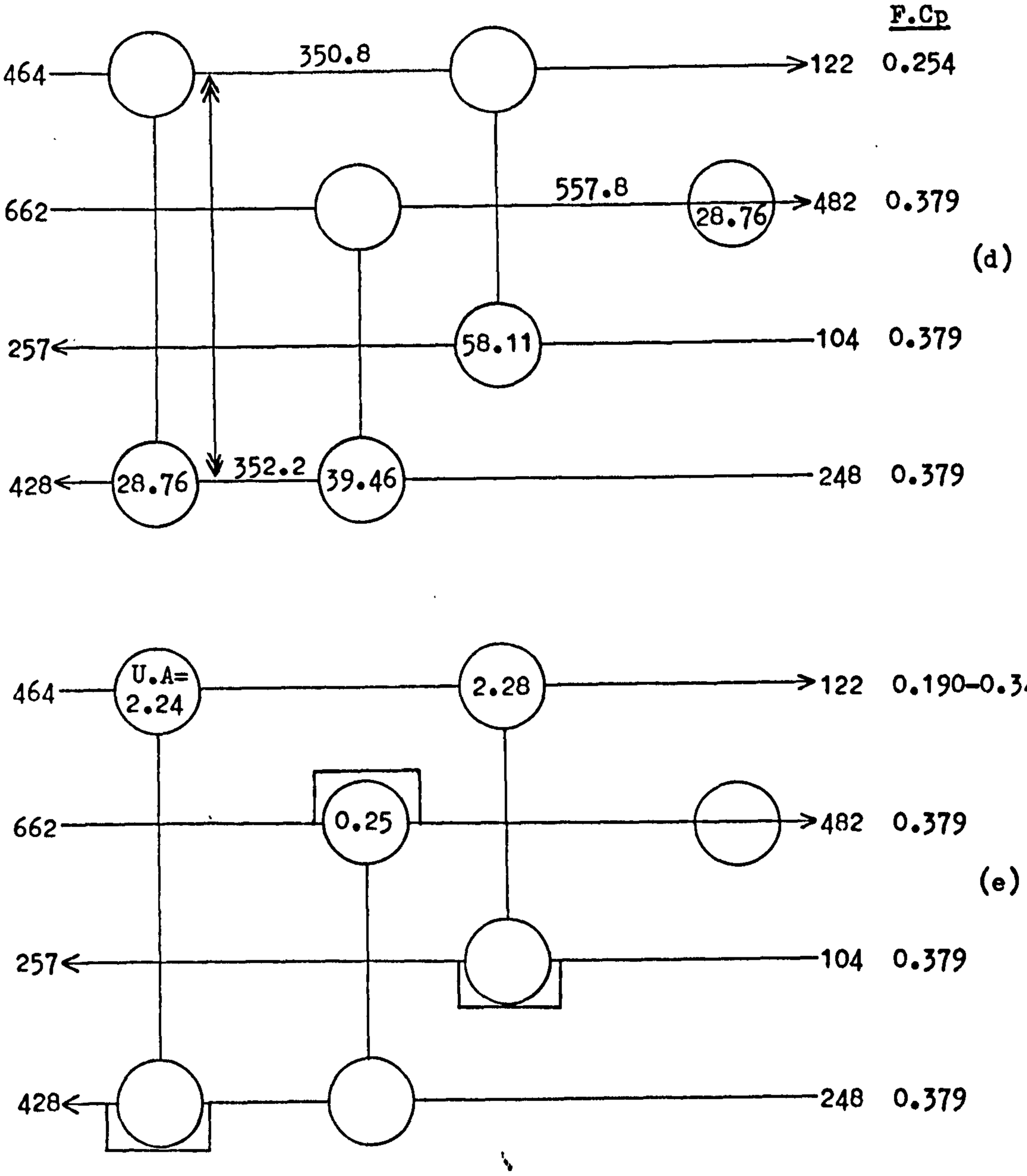


Figure 6.14 continued

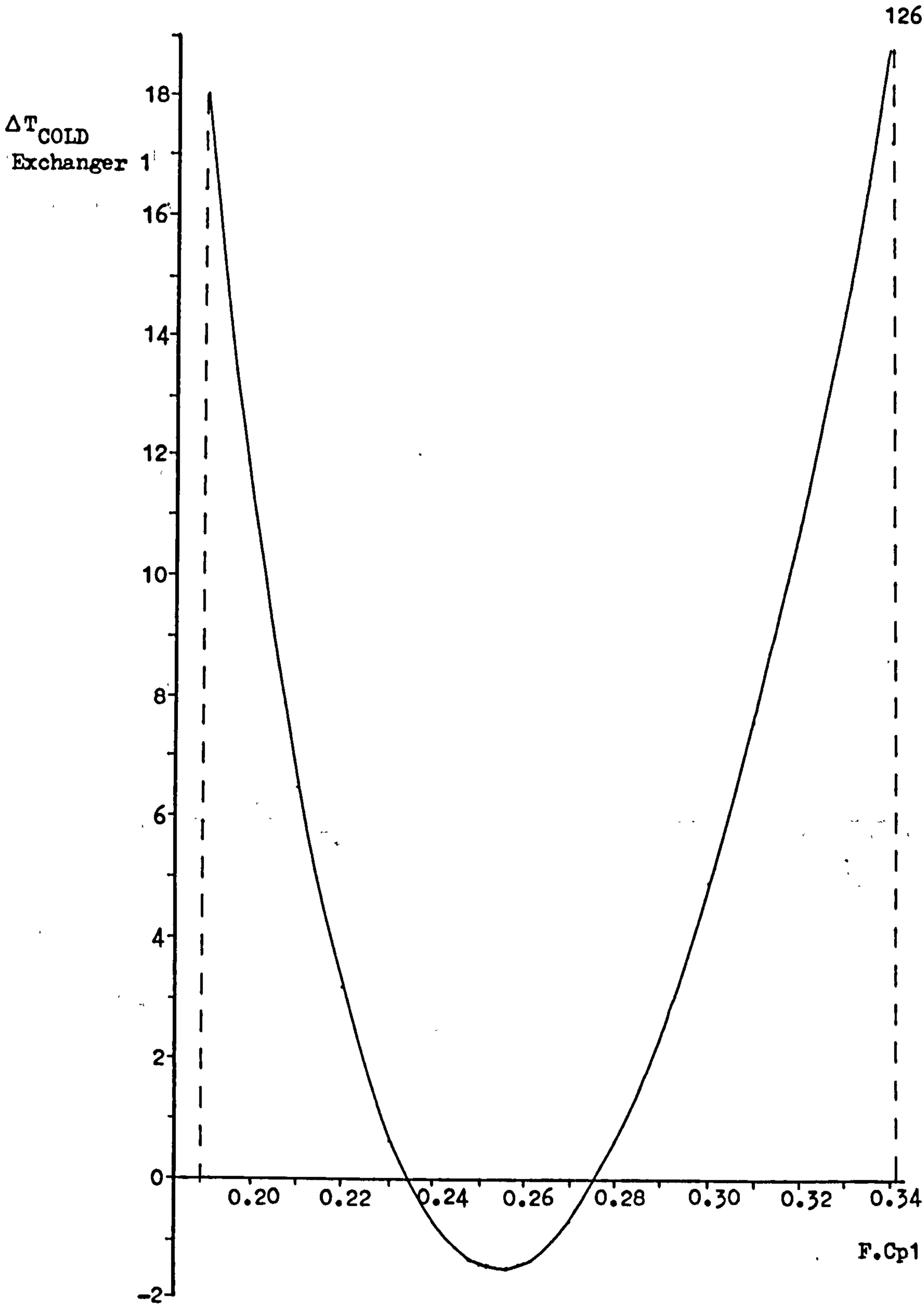


Figure 6.15 The variation of the cold end ΔT of exchanger 1 with F.Cp1 in Example Problem 6.3.

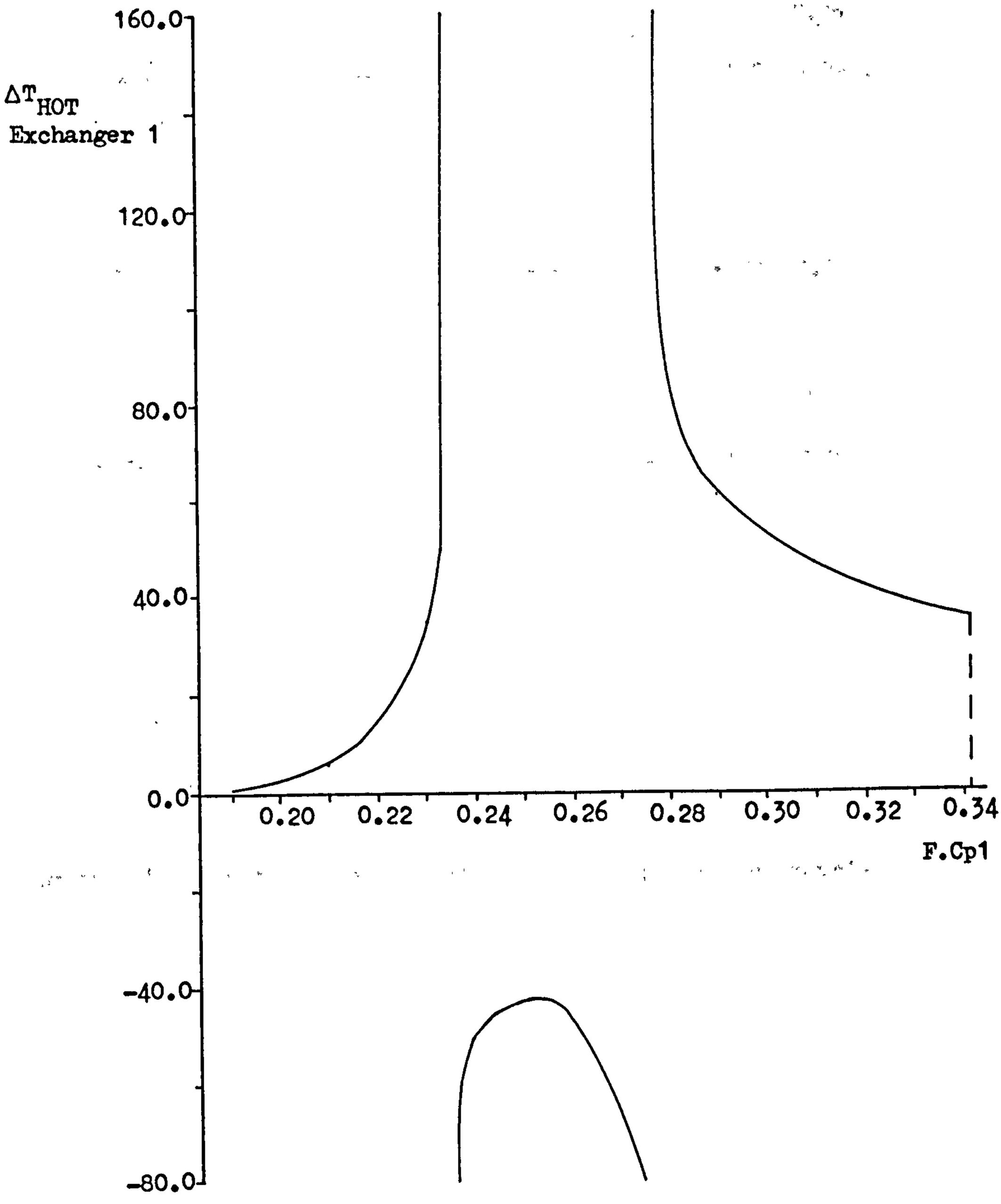


Figure 6.16 The variation of the hot end ΔT of exchanger 1 with F.Cp1 in Example Problem 6.3.

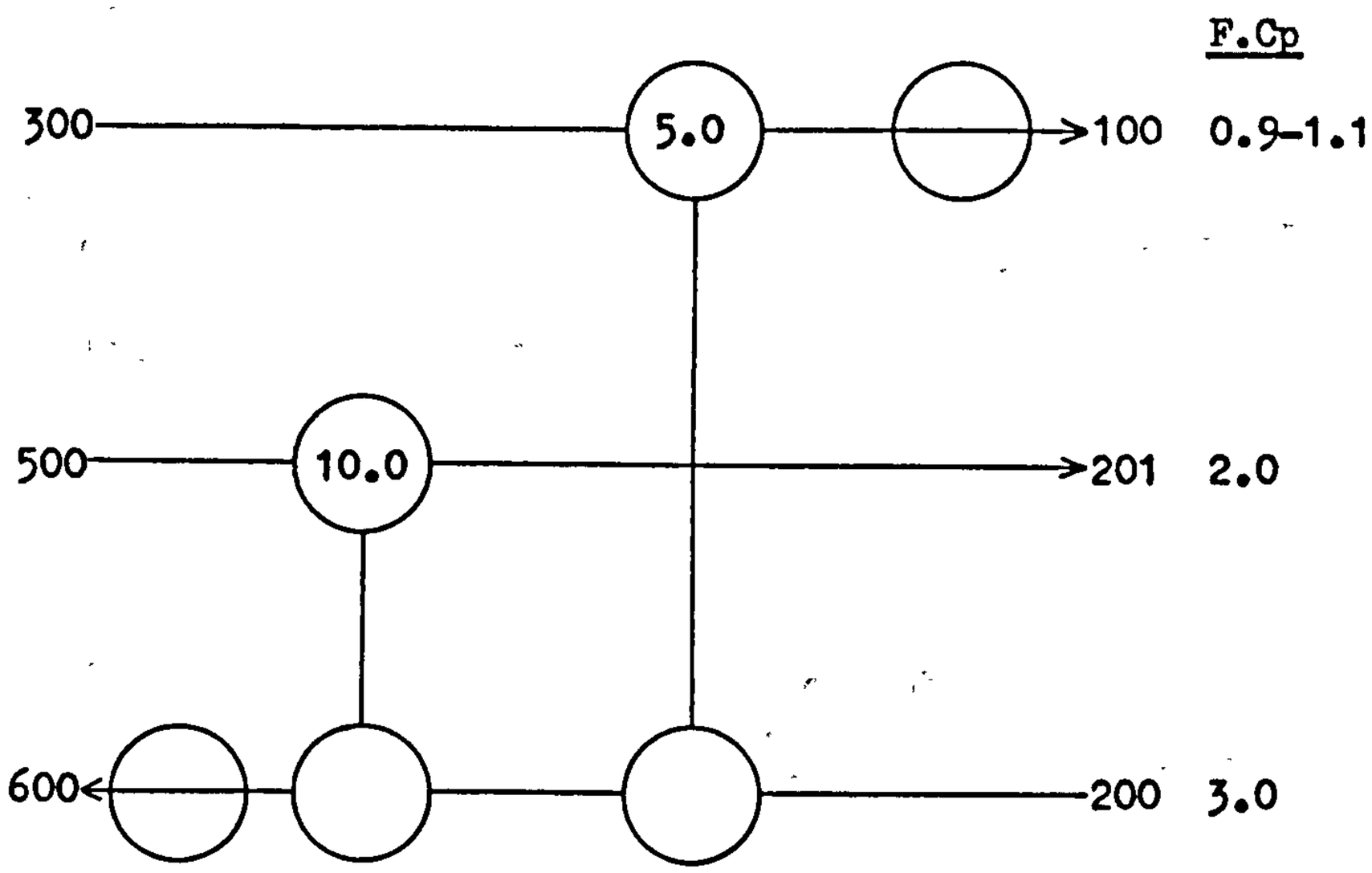


Figure 6.17 A network for the pinched problem used in Example Problem 6.4.

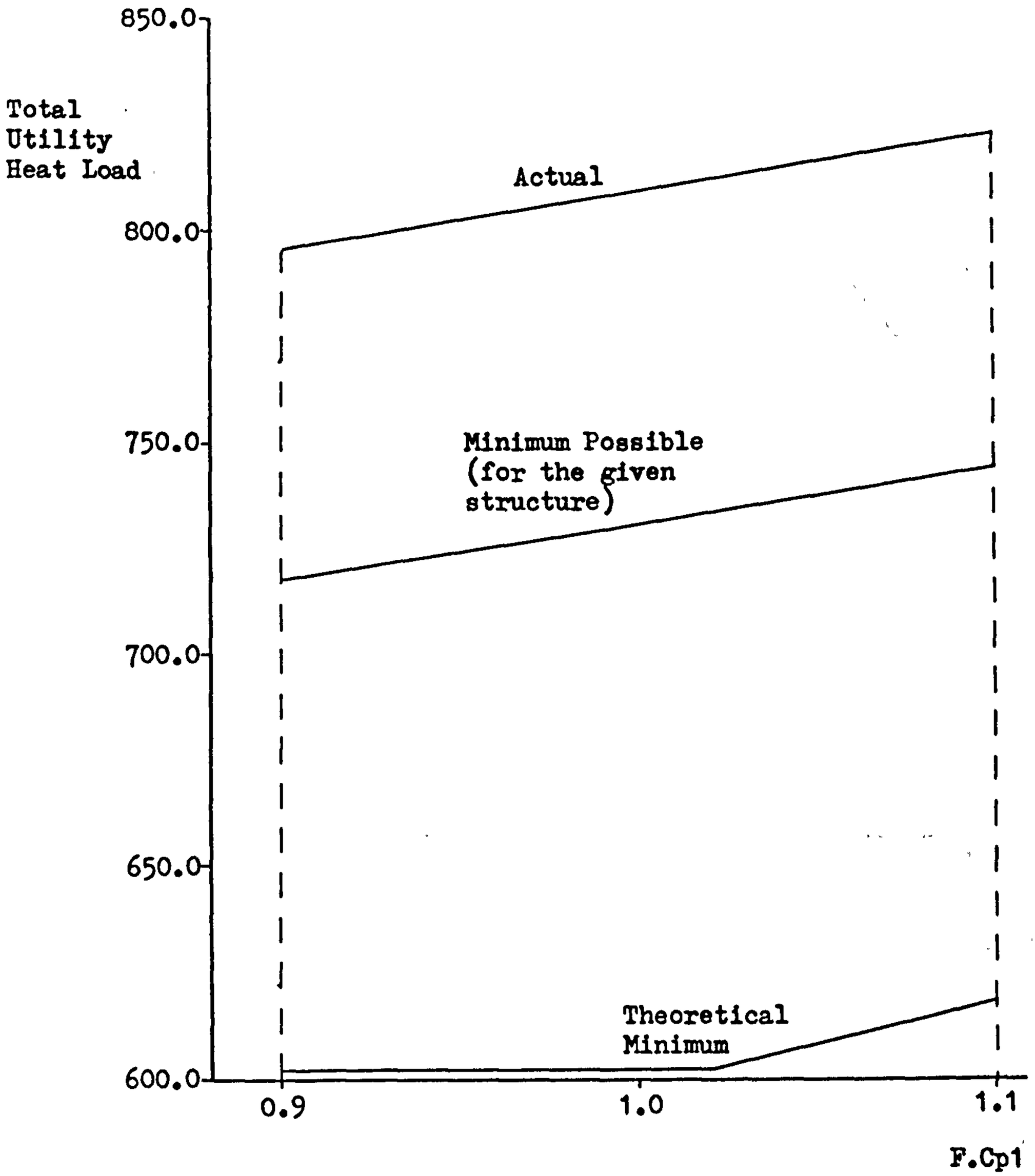


Figure 6.18 The variation of total utility use with F.Cp1 in Example Problem 6.4.

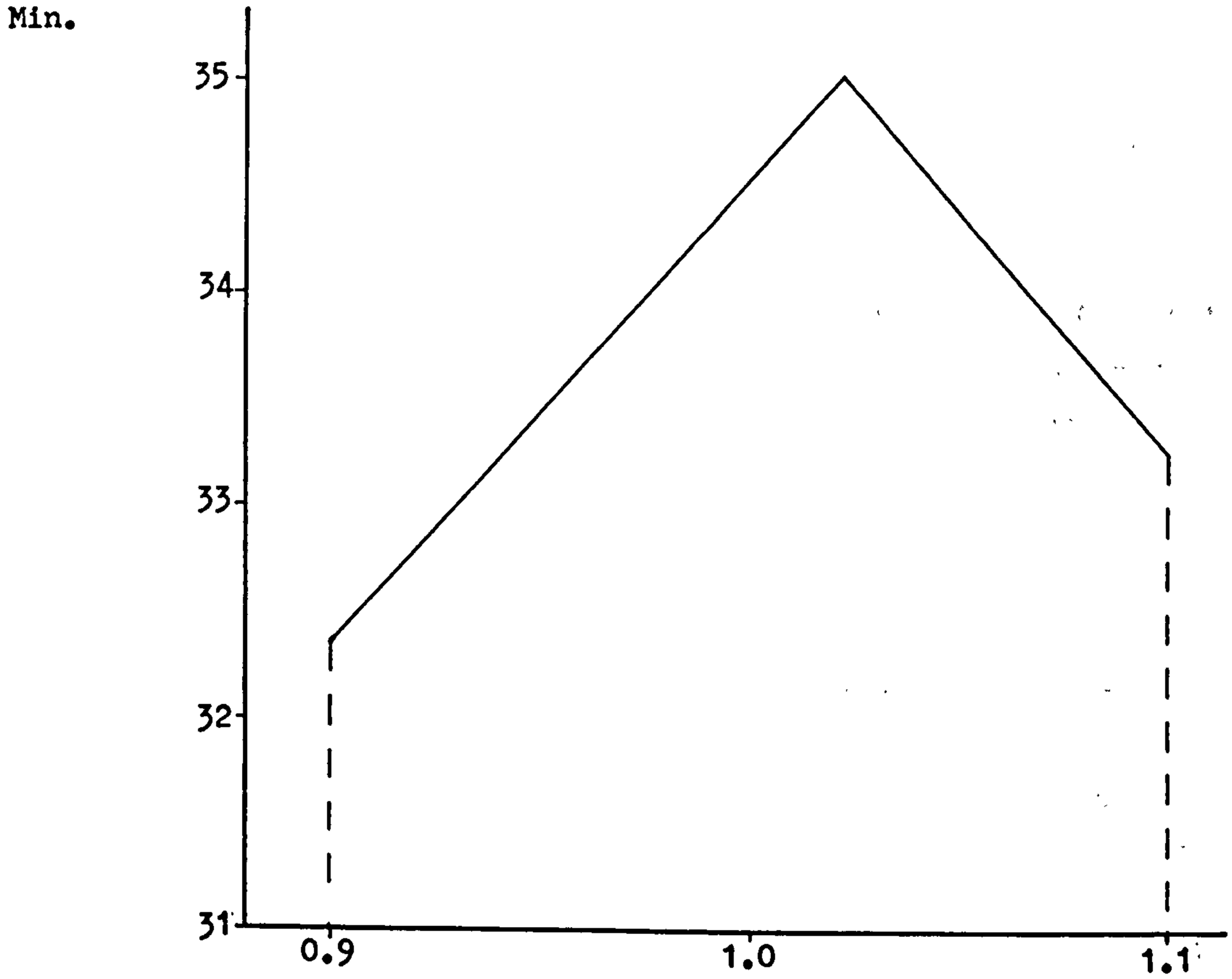


Figure 6.19 Actual total utility use as a percentage greater than theoretical total minimum use in Example Problem 6.4.

C H A P T E R 7

DESIGN EXAMPLES

7.1 An Unpinched Problem

The problem is based on literature problem 5SP1, the data for which are given by Linnhoff (1979).

Linnhoff (1979) uses the Thermodynamic-Combinatorial (TC) method to design five alternative networks solving the problem for a minimum temperature approach of 20 (^{o}F) or greater. These are the only networks not featuring stream-splits and having the minimum number of units which solve the problem for the given minimum allowable temperature approach.

Linnhoff notes that other solutions may exist if the temperature approach is relaxed to take values of less than 20. From the flexibility viewpoint it is useful to start from a network with a relatively high minimum temperature approach, as considerably smaller values are likely to be encountered elsewhere in an operating range. Starting with high temperature approaches will tend to increase the probability that the temperature approaches will remain feasible over the operating range.

As the problem is unpinched with only heating utilities used the meeting of target temperatures ensures that maximum possible heat is recovered and comparisons between networks can be made simply on the basis of the capital costs of equipment (exchanger areas). Utility (steam) usage is the same for each network for any operating conditions.

The five networks are shown in Figure 7.1. The stream conditions shown are those specified for problem 5SP1 and these are taken as nominal operating conditions. Bypasses have been assigned to each exchanger to ensure that the degrees of freedom of the networks are zero. The choice of whether to bypass the hot or cold side of an

exchanger is not within the scope of this work as it has no bearing on the flexibility of a network but generally the suggestion of Marselle (1980) that the bypass should be as near as possible to the temperature it is controlling is followed. The nomenclature and units are the same as have been used previously. In Figure 7.1 intermediate temperatures, exchanger U.A values and heat loads (including those for utility exchangers where appropriate) are indicated. The exchangers have been sized for nominal operating conditions.

In general it is a good idea to use the non-flexible synthesis method to design base networks for an extreme set of operating conditions, possibly one of the four cornerpoints identified by Morari and co-workers, as these are likely to provide a severe test for a network.

An operating range is formed by allowing all inlet heat capacity flowrates, F.Cp, to vary independently over the range 0.9 to 1.0 times the nominal values.

The constraint function maximization program is used to find a set of operating conditions, point I, which happens to be a cornerpoint and the same for all five networks, which produces either extreme sizing or structural infeasibilities.

The performance of the five networks at these conditions is shown in Figure 7.2. The required exchanger U.A values are indicated for each exchanger. The maximum required U.A value which has been found so far, at this stage the maximum of the values required for nominal operating conditions and the operating conditions shown in Figure 7.2, for each exchanger are given as aW (for the heater) and a1 to a5 for the exchangers (numbered from left to right in each case). Infeasible values are marked * and temperature crossovers are arrowed.

It can be seen that for network options (d) and (e) exchanger 4 cannot be sized correctly to handle this set of operating conditions

because of the negative hot side temperature approach. They are structurally infeasible.

As in the non-flexible design methods it is likely to be better economically to maintain the minimum number of exchangers and so at this stage networks (d) and (e) can be removed from consideration until it can be seen whether networks (a), (b) and (c) may be made fully feasible by oversizing the existing exchangers.

For network (c) it can be seen that the required area of exchanger 1 for conditions I gives a U.A of 3.59 whereas at nominal conditions the required U.A. value is 4.66. The maximum required U.A value found so far then, al, is 4.66 and if no larger required value is found this must become the installed value.

Continuing the search for infeasible operation other operating conditions are found which produce extreme sizing or structural infeasibilities. The conditions are not the same for each network. The performance of the three remaining networks at these condtions is shown in Figure 7.3. Network (a) is structurally infeasible at this set of conditions. If networks (b) and (c) are now sized according to the 'a' values indicated it is found that they can operate over the entire operating range. The final exchanger U.A values and the operating range are given in Figure 7.4.

All the exchangers have had to be oversized from the nominal design.

Using the values assumed by Linnhoff (1979) for the overall heat transfer coefficients of:

$$U \text{ (Heaters)} = 200 \text{ BTU/hr ft}^2\ ^\circ\text{F}$$

$$U \text{ (Exchangers)} = 150 \text{ BTU/hr ft}^2\ ^\circ\text{F}$$

and the costing formula Cost ($) = $350\ A(ft^2)^{0.6}$ the exchanger areas and costs can be calculated and these are given in Table 7.1.

As the operating cost of each network (the amount of steam used) is the same then it is clear that network (b) is the cheaper.

In this example two cornerpoints of the operating range were found to be critical in sizing the exchangers.

Using the same techniques the five networks may be designed to handle a reduced operating range with F.Cp values varying between 0.95 and 1.0 times the nominal values. The final networks designed to handle this range are given in Figure 7.5. In this case all the networks can be made feasible simply by appropriate oversizing.

By fixing F.Cp values at their nominal values and allowing all inlet temperatures to vary independently from 0.9 to 1.0 times the nominal values it is found that all the networks are structurally infeasible. The operating conditions producing infeasible operation are given in Figure 7.6 with the infeasibilities indicated. If this range of inlet temperatures must be accommodated then a combined network must be formed, with its associated probable extra cost in terms of the number of units used. Therefore there is a high incentive to make possible temperature deviations smaller, possibly by tighter design of the external process. In fact if the temperature variations can be kept to within 0.95 to 1.0 times the nominal values then all five networks may be made feasible by appropriate oversizing. These networks are shown in Figure 7.7. Although all networks can be sized to give feasible operation for F.Cp values in the range 0.95 to 1.0 times the nominal values with inlet temperatures taking nominal values or for inlet temperatures in the range 0.95 to 1.0 times the nominal values with nominal F.Cp values this does not imply that the networks can be sized to handle both ranges occuring jointly. In this case networks (a) and (c) are found to be structurally infeasible while networks (b), (d) and (e) can be sized to give feasible operation. The operating conditions producing infeasible operation and the feasible networks are given in Figure 7.8.

The areas of the exchangers may be calculated using the heat transfer coefficients given previously and the costs may be determined for the networks designed for the five operating range considered. The costs are given in Table 7.2 neglecting those networks which were found to be structurally infeasible.

From this table it can be seen that network (b) provides the cheapest alternative except for an operating range of inlet temperature variations from 0.95 to 1.0 times the nominal values when network (a) is cheaper.

Timing

As all critical points found were cornerpoints a simple cornerpoint checking routine could have been used to give the same solution in each case.

Comparing the computer time required to verify that network (b) is fully feasible for the joint operating range of F.Cp and inlet temperature variations gives:

Cornerpoint checking	912 c.p. seconds (CDC 6500)
Constraint maximization	438 c.p. seconds (CDC 6500)

Therefore not only does the constraint function maximization procedure check feasibility over the full operating range, rather than just the cornerpoints, it is also more than twice as fast in this case.

The computer time is dominated by network evaluations, that is the solving of the network's describing equations for particular operating conditions.

In the above example the number of cornerpoints is $4^5 = 1024$, therefore the cornerpoint checking procedure must make 1024 network evaluations. For constraint maximization 16 constraints must be maximized in turn and each maximization required, on average, about 30 iterations to converge therefore approximately 480 network evaluations are needed.

To demonstrate the methods of dealing with combined networks an example follows which forces the use of a combined network.

Network (b)

Area (ft^2)	W	1	2	3	4
Nominal Case Design	136	330	197	801	168
Final Design	166	358	666	1114	284

Cost of nominal design $53 260 Cost of final design $70 680

Network (c)

Area (ft^2)	W	1	2	3	4
Nominal Case Design	144	311	801	313	169
Final Design	177	363	1114	854	284

Cost of nominal design $55 740 Cost of final design $73 920

Table 7.1 Area and cost of networks designed for the range 0.9 to 1.0 of nominal F.Cp.

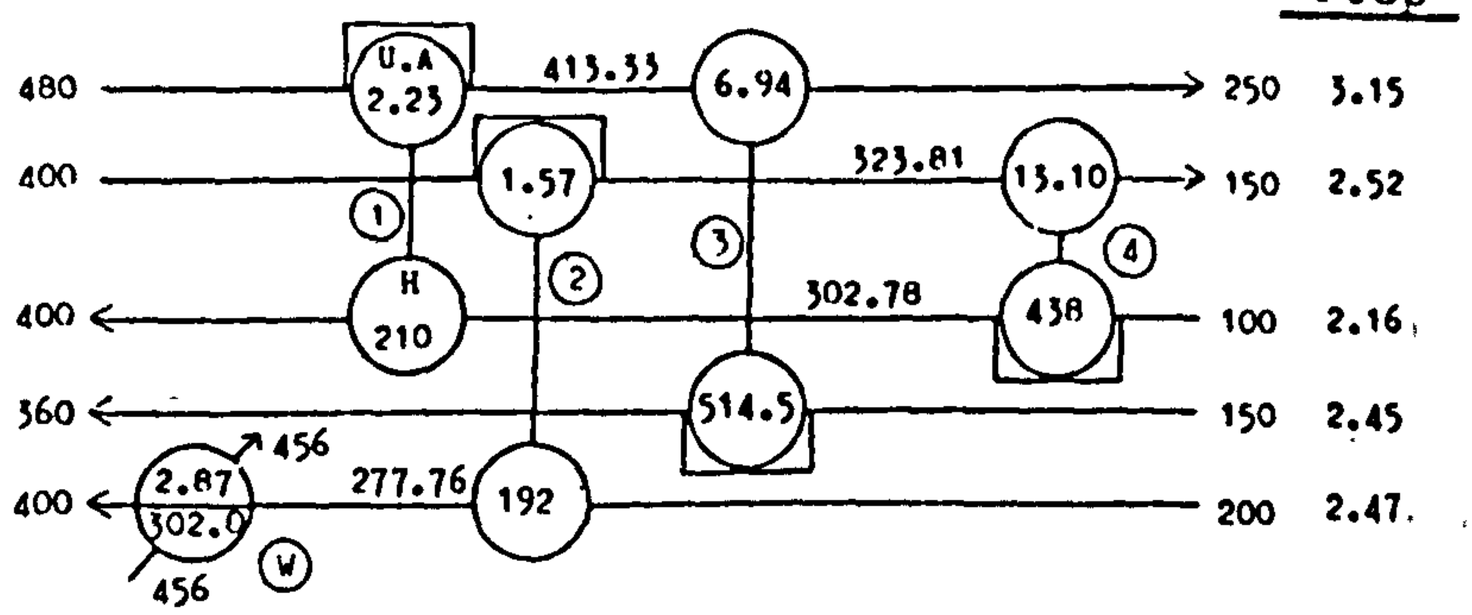
Cost ($)

Network \ Operating Range	Nominal Design	0.95-1.0 F.Cp	0.9-1.0 F.Cp	0.95-1.0 T_i	0.90-1.0 T_i	0.95-1.0 F.Cp and 0.95-1.0 T_i
(a)	53940	61150	--	60280	--	--
(b)	53260	59330	70680	66100	--	75770
(c)	55740	62950	73920	68800	--	--
(d)	56060	63900	--	70480	--	81360
(e)	53380	61220	--	66770	--	91050

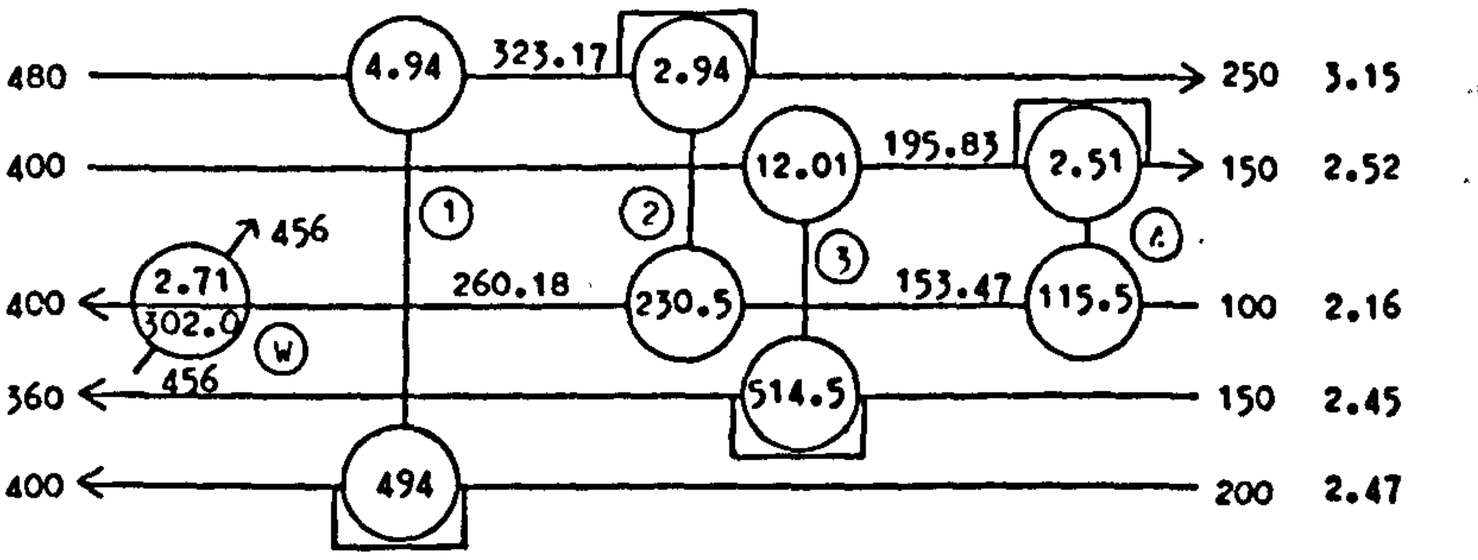
Cost as a percentage greater than the cheapest nominal design

Network \ Operating Range	Nominal Design	0.95-1.0 F.Cp	0.9-1.0 F.Cp	0.95-1.0 T_i	0.9-1.0 T_i	0.95-1.0 F.Cp and 0.95-1.0 T_i
(a)	1.3	14.8	--	13.2	--	--
(b)	0	11.4	32.7	24.1	--	42.3
(c)	4.7	18.2	38.8	29.2	--	--
(d)	5.3	20.0	--	32.3	--	52.8
(e)	0.2	14.9	--	25.4	--	71.0

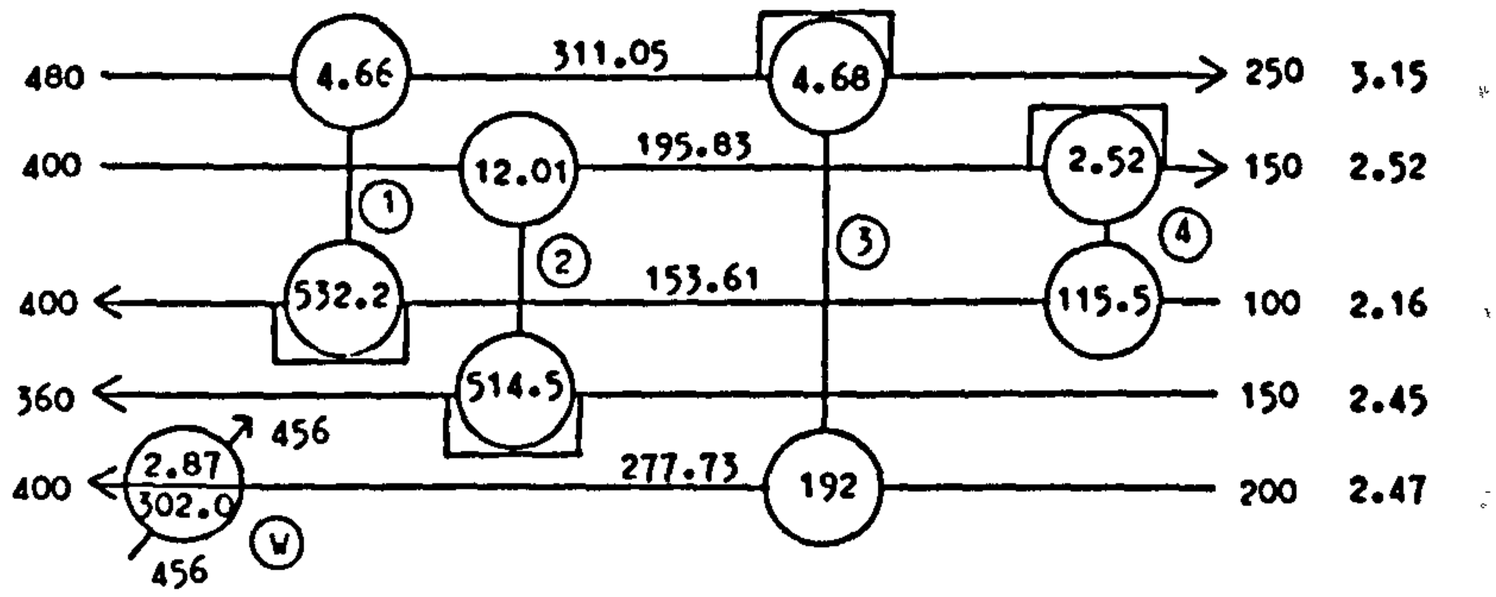
Table 7.2 The cost of the networks designed for the five operating ranges considered.



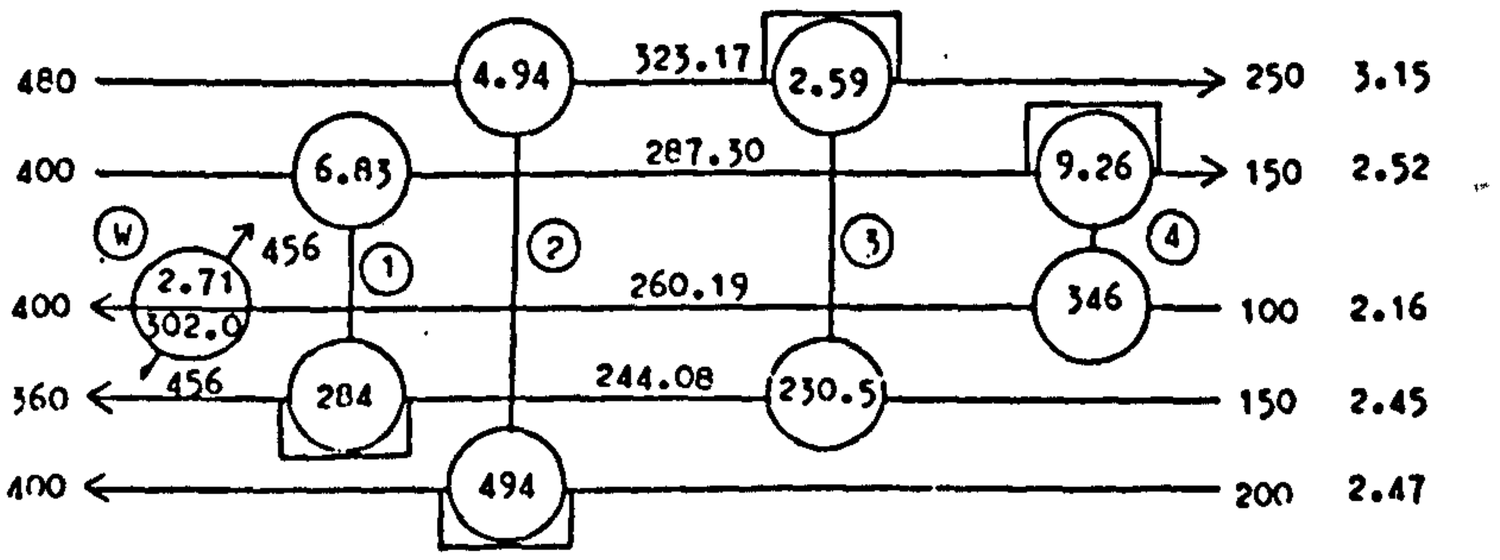
(a)



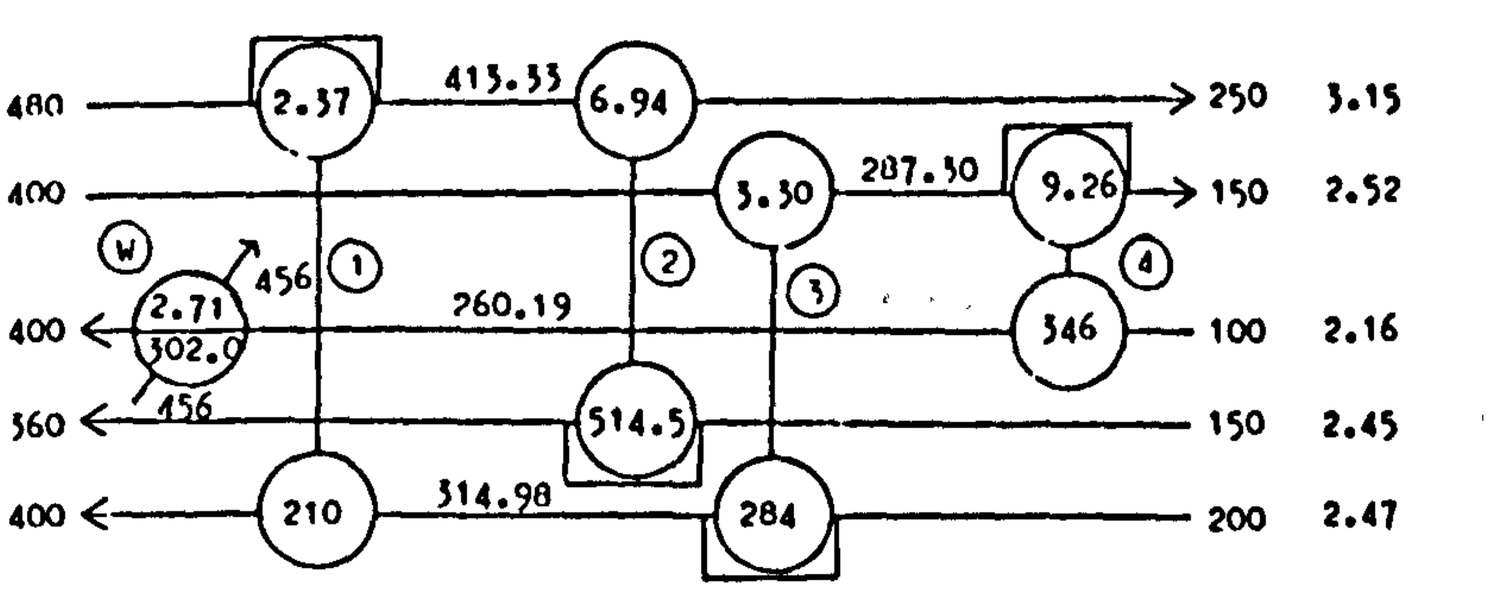
(b)



(c)



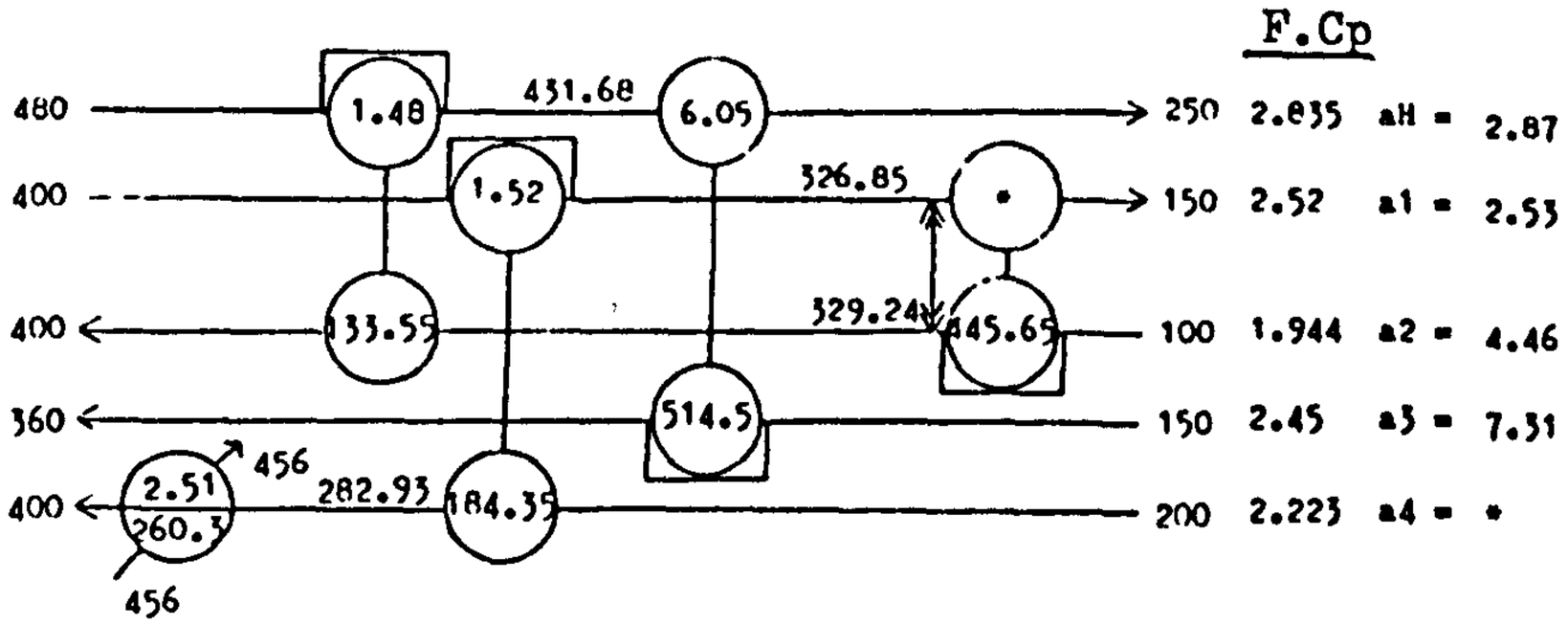
(d)



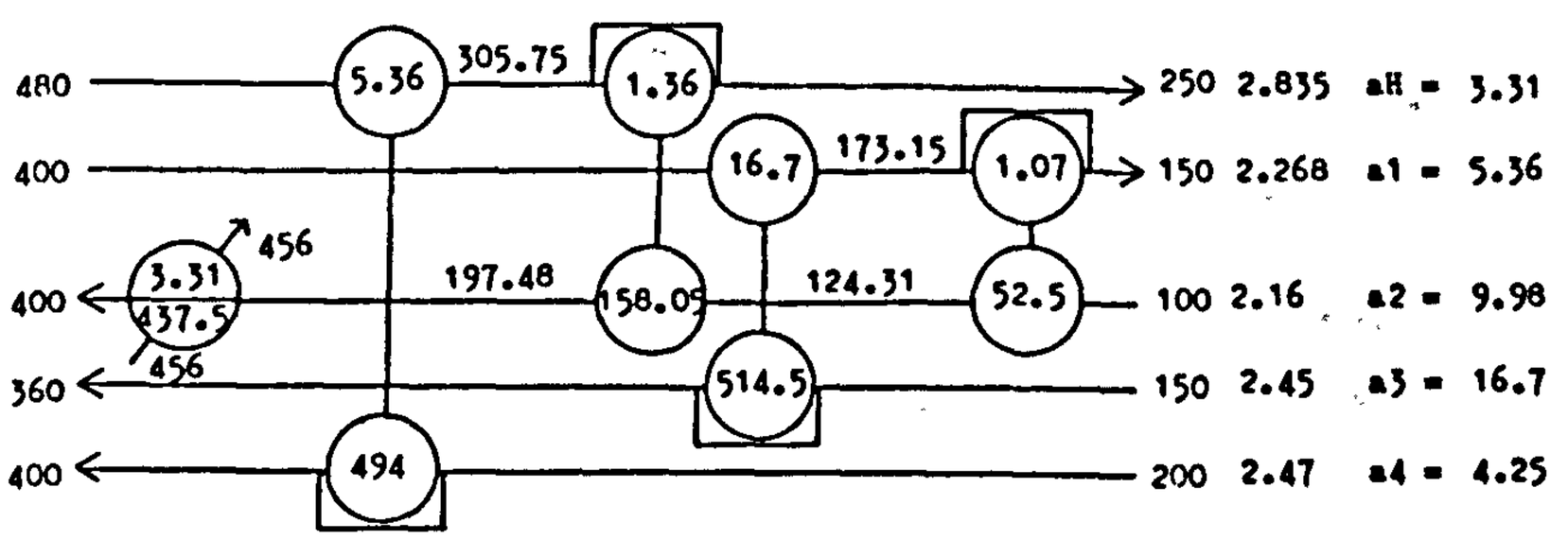
(e)

Figure 7.1 Five networks solving problem 5SP1.

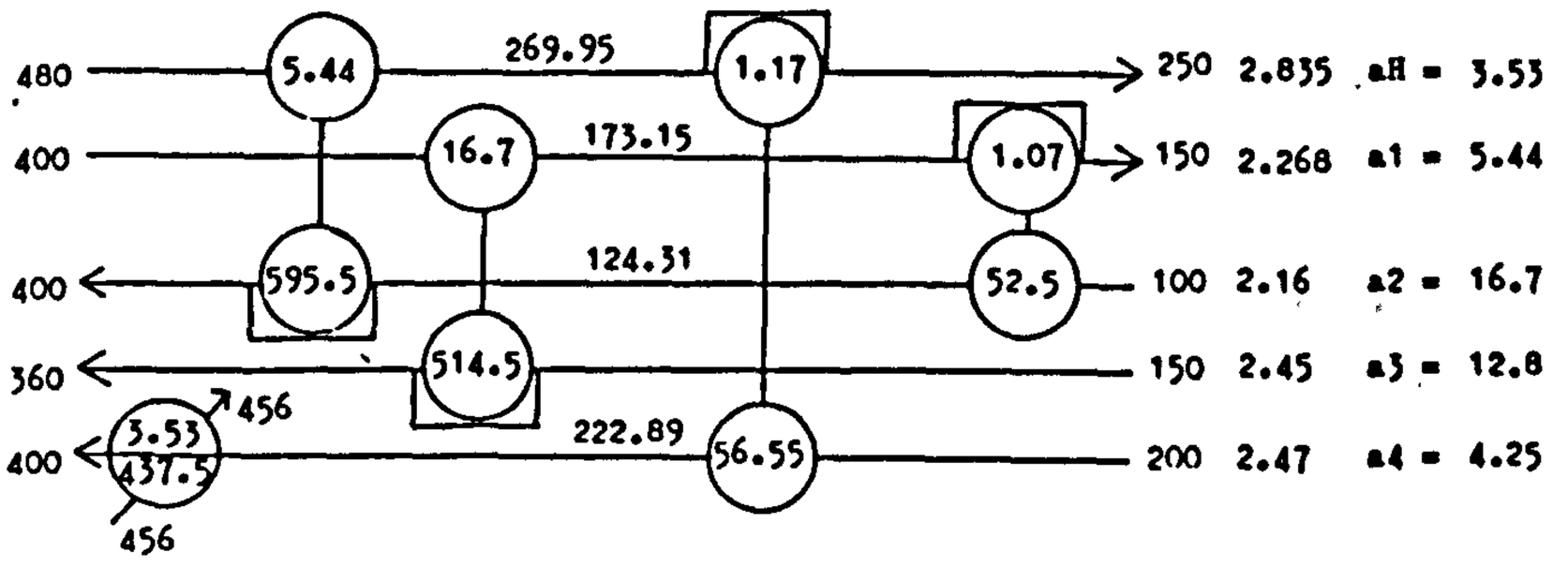
Figure 7.2 Performance of the five networks at point I in the operating range.



(a)



(b)



(c)

*

(d)

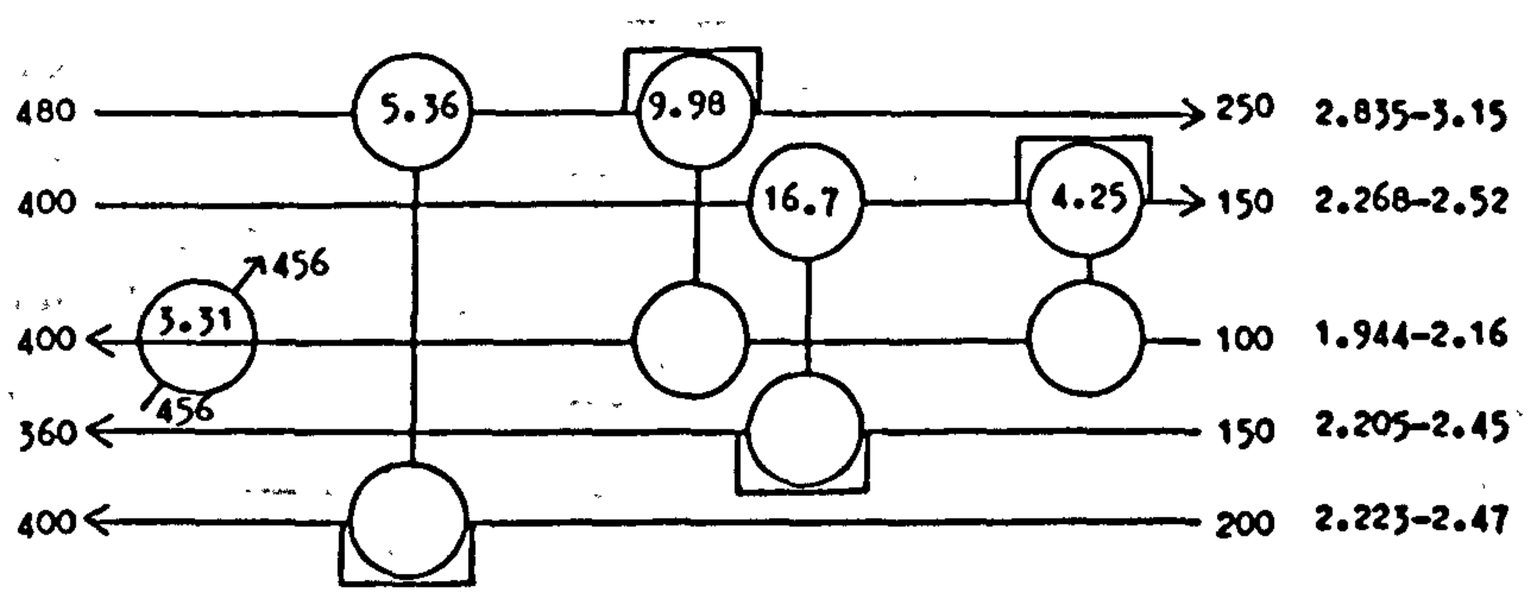
*

(e)

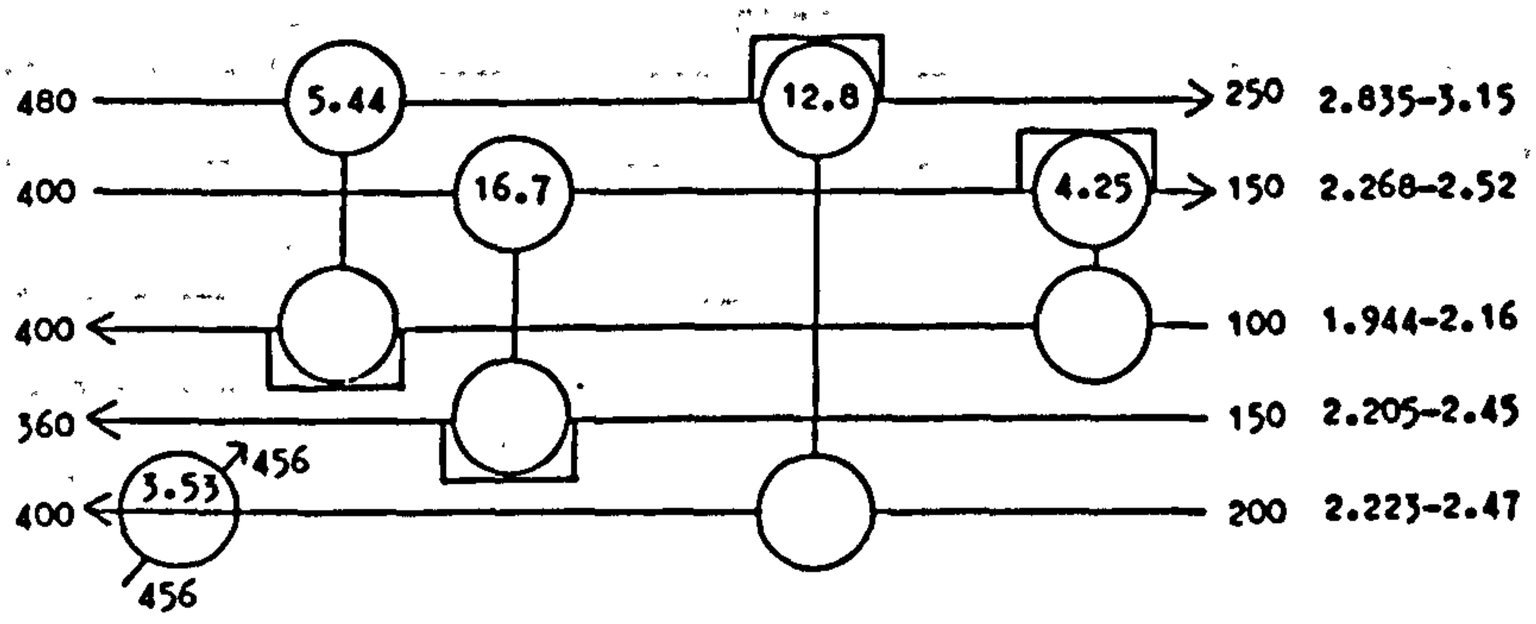
Figure 7.3 Performance of the remaining networks at other operating conditions.

(a)

F.Cp



(b)

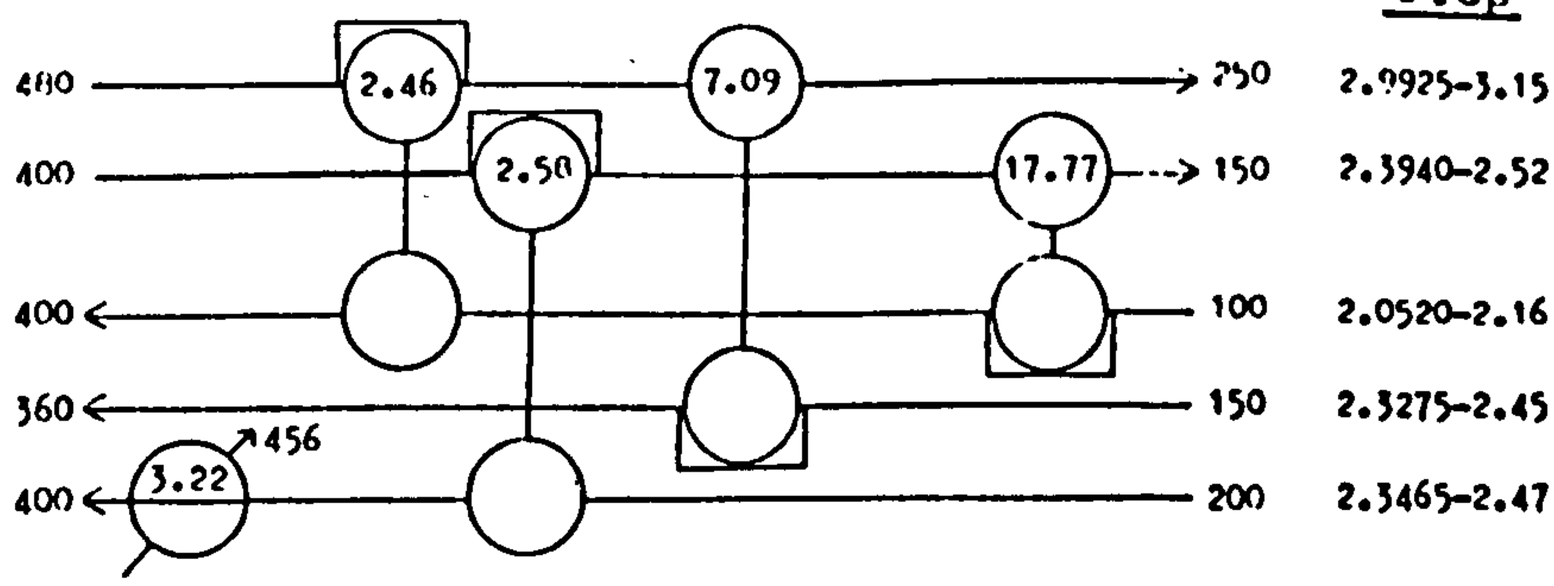


(c)

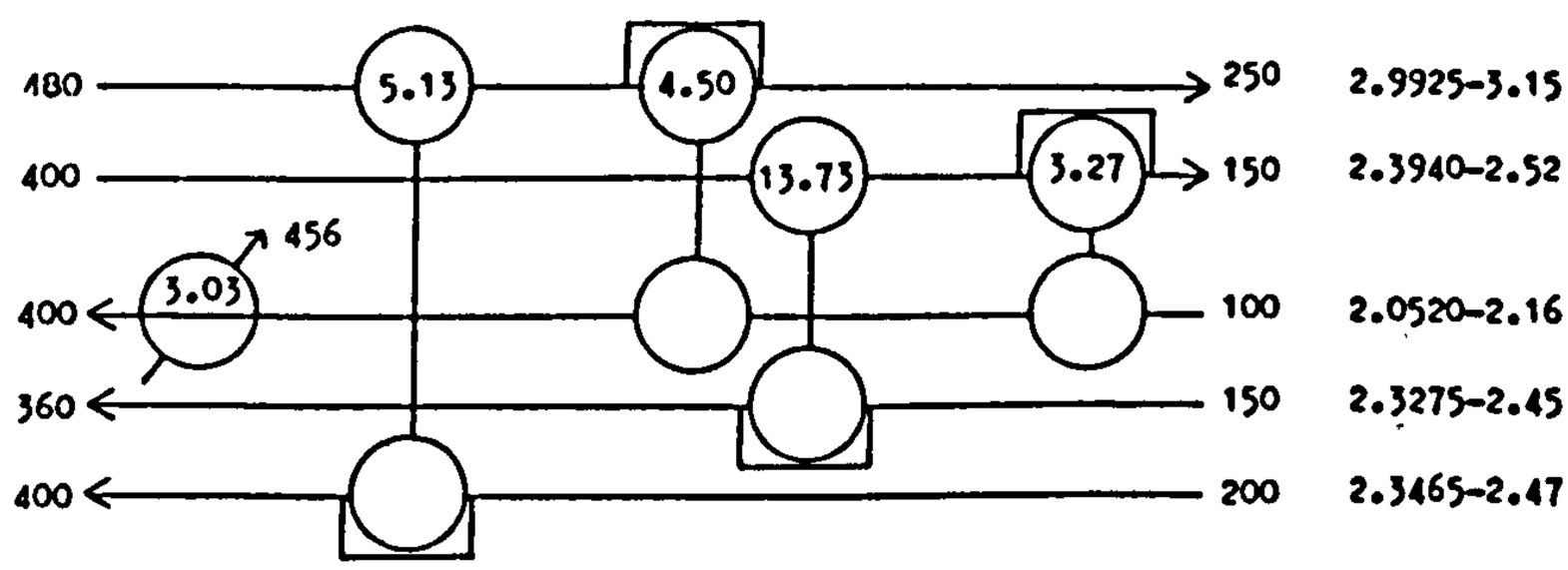
(d)

(e)

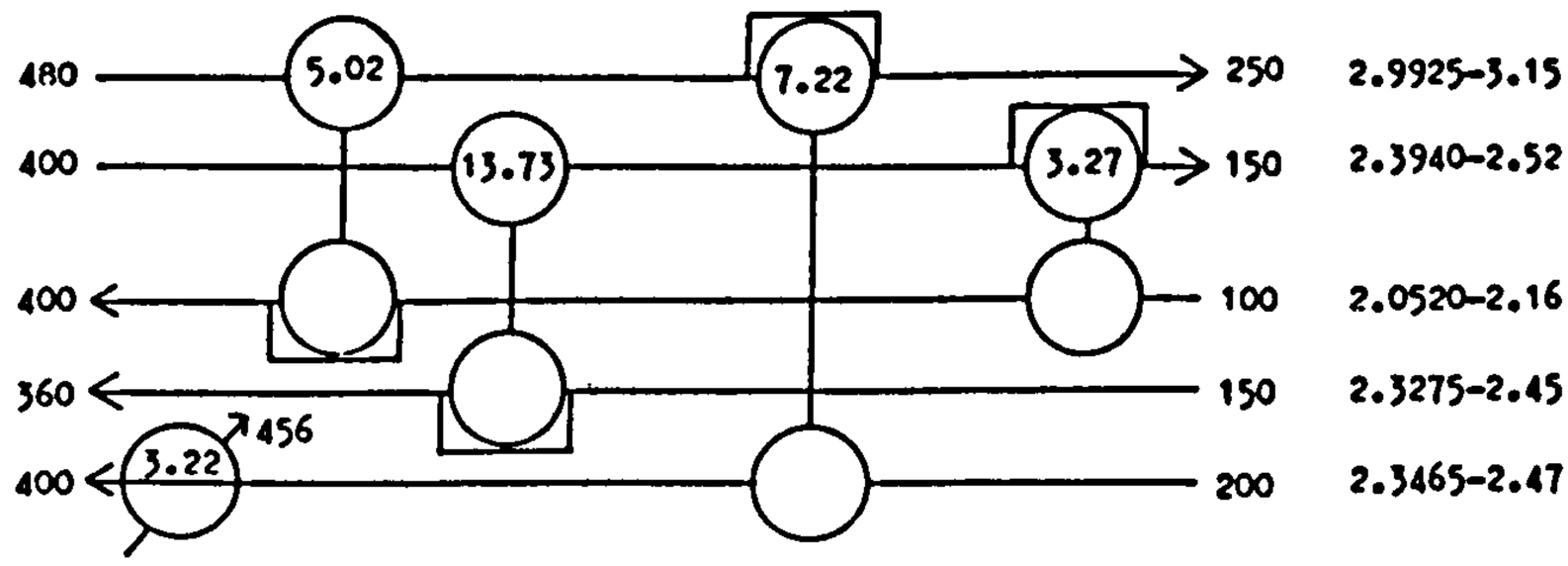
Figure 7.4 Networks able to handle a range of F.Cp values from 0.9 to 1.0 times nominal.



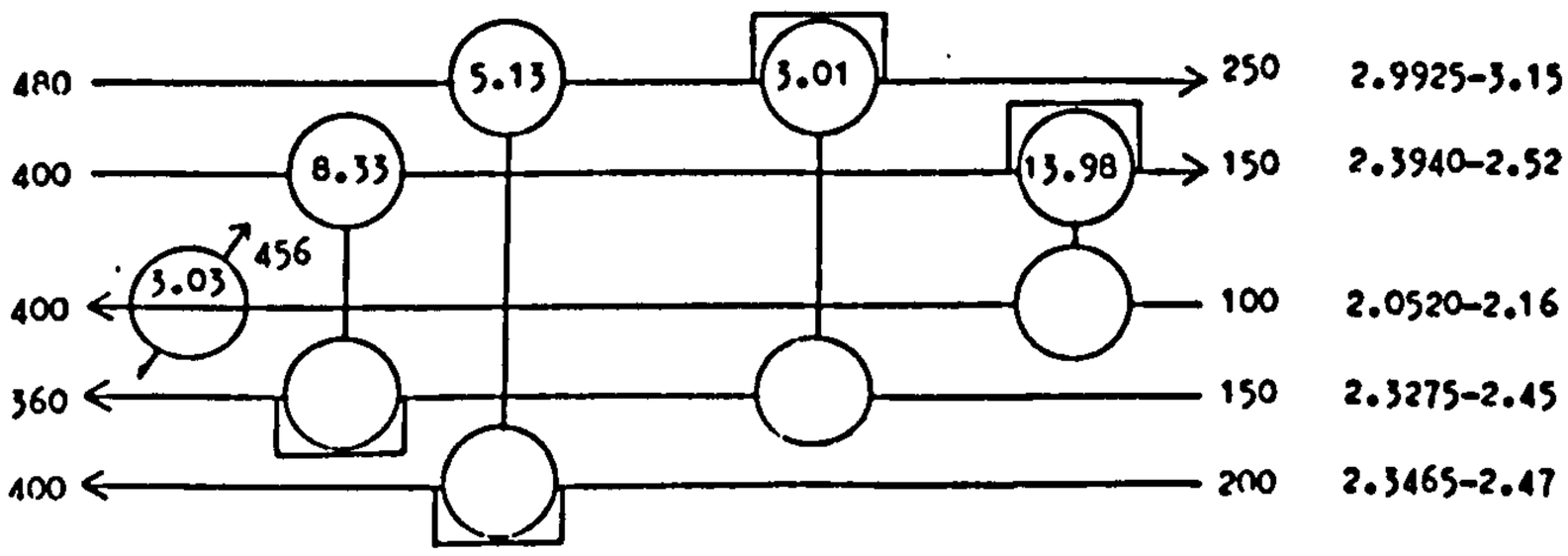
(a)



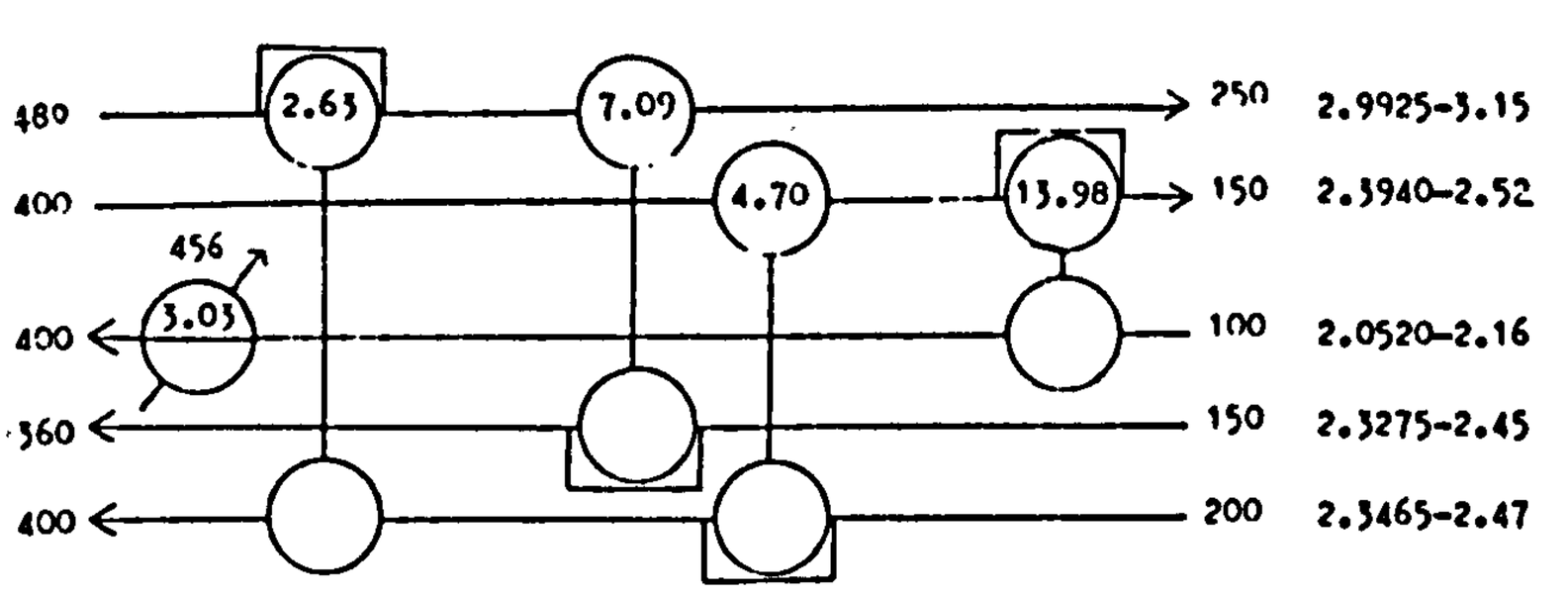
(b)



(c)



(d)



(e)

Figure 7.5 Networks able to handle a range of F.Cp values from 0.95 to 1.0 times nominal.

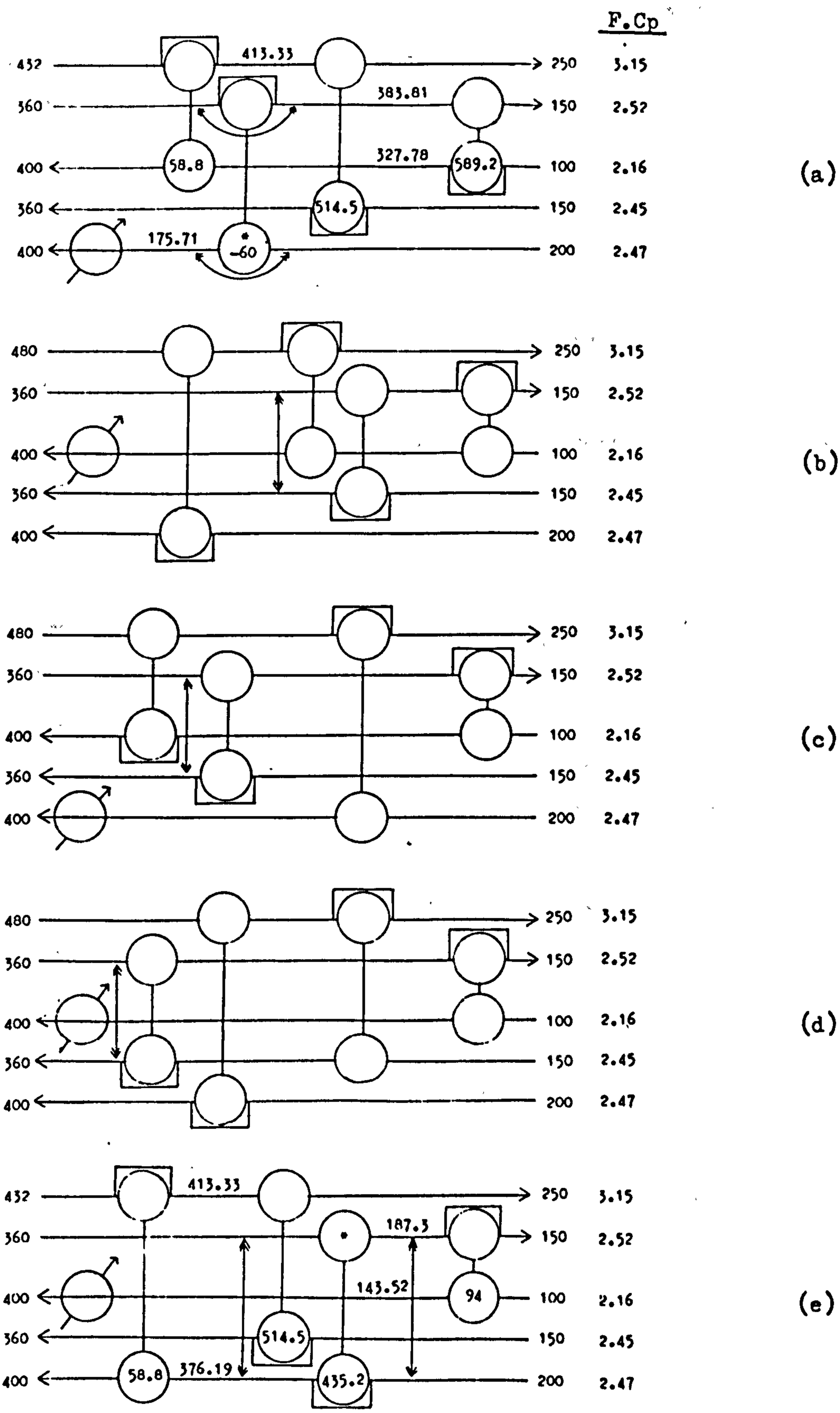


Figure 7.6 Infeasible operation of networks for a range of inlet temperatures from 0.9 to 1.0 times nominal.

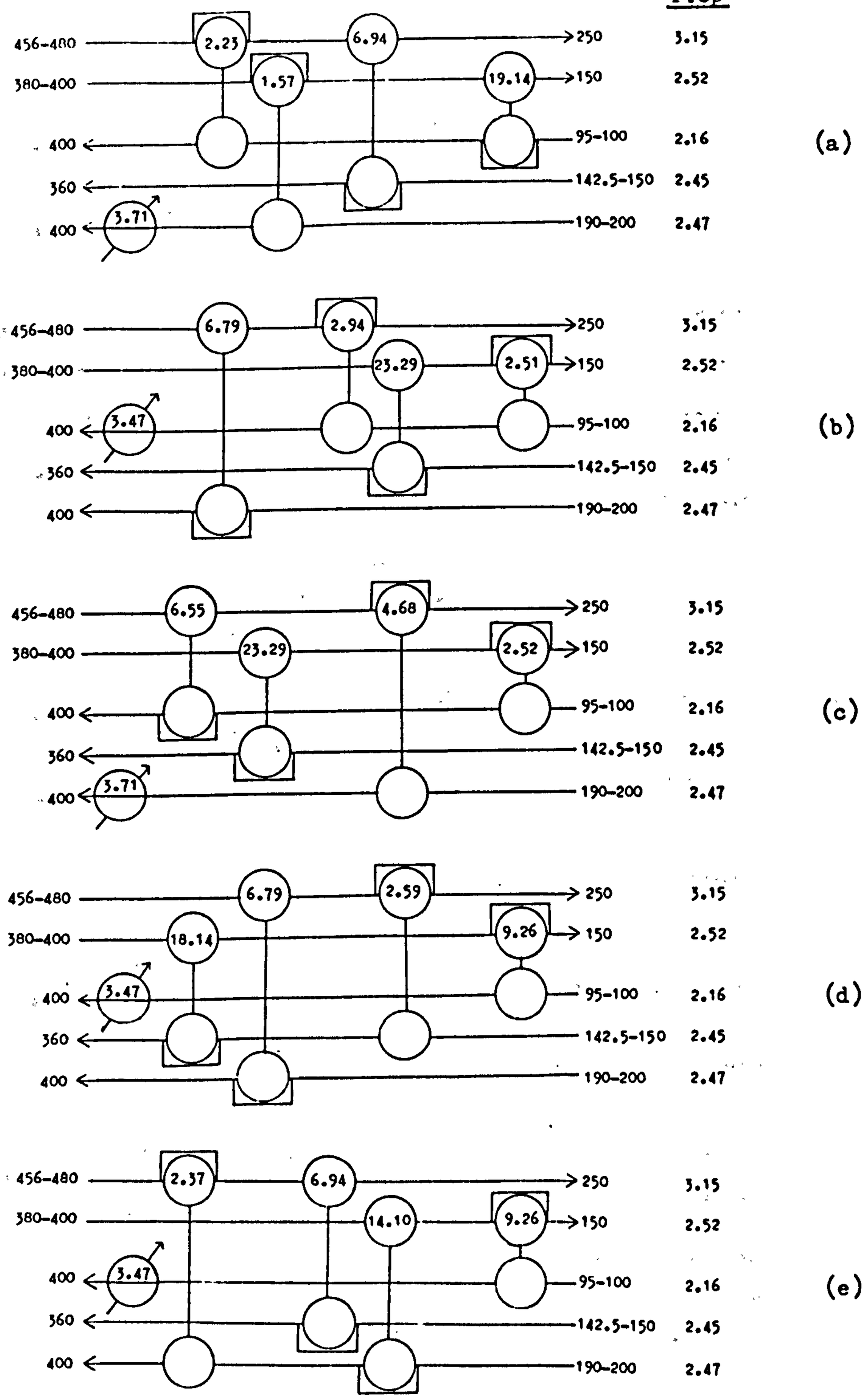


Figure 7.7 Networks able to handle a range of inlet temperatures from 0.95 to 1.0 times nominal.

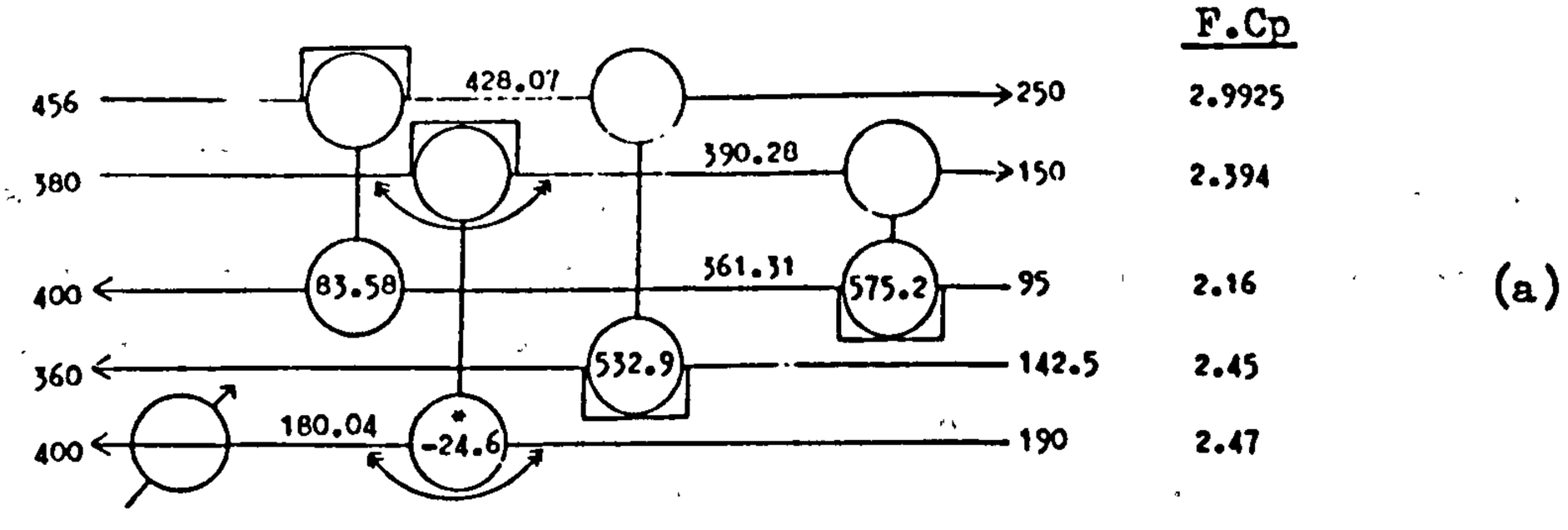


Figure 7.8 Infeasible operating conditions and networks able to handle F.Cp values from 0.95 to 1.0 times nominal and inlet temperatures from 0.95 to 1.0 times nominal.

7.2 A Combined Network

The stream data used in this problem are given in Table 7.3. The stream data have been chosen to force the use of a combined network and are atypical in that it has been specified that the F.Cp values for streams 1 and 2 can take two discrete sets of values i.e. (1.25, 1.1) or (0.76, 1.4), not a continuous range between these values. In practice such a situation could arise due to a change in feedstock.

The F.Cp value of stream 5 varies continuously between 0.4 and 0.5.

At two points, I and II, in the operating range two network designs A and B perform as shown in Figure 7.9. The problem is unpinched throughout the operating range and only cooling utilities are used,therefore utility use is the theoretical minimum for all operating points. No bypasses are shown in networks A and B but as each network has the minimum number of units all exchangers must be bypassed for regulatory control purposes.

As can be seen, for operating point I network A cannot operate feasibly but network B can, whereas for point II network A can operate feasibly but network B cannot. Therefore neither network A nor network B can operate feasibly over the entire operating range. However, for all operating points in the range either network A or network B is feasible with the U.A values indicated. A combined network of A and B will be feasible.

The combined network in Figure 7.10 contains networks A and B as special cases. If exchanger 3 is fully bypassed network A is obtained with the other three bypasses able to provide regulatory control. If exchanger 2 is fully bypassed network B is obtained. For any particular set of operating conditions a control scheme would have to detect whether network A or network B was required, isolate the corresponding exchanger and choose the appropriate bypasses to use for regulatory control.

Now that a combined network must be used, however, it is advantageous to make use of the interaction between networks A and B with a view to eliminating the need for optimizing (in this case switching) control. In this case for regulatory control only three of the four exchangers need to be bypassed. Eight combined network structures can be formed by combining networks A and B and assigning three bypasses. The eight alternatives are shown in Figure 7.11.

Of these eight options (1) and (5) are unacceptable as the placement of bypasses makes the problem badly posed (see Chapter 3). Of the remaining networks preliminary computer modelling shows alternative (7) to be the most promising.

The network, shown in Figure 7.12, has design flexibility as defined by Linnhoff. That is a unique solution for A1, A2, A3 and A4 does not exist. If A3 is specified then the minimum required A1, A2 and A4 can be calculated for any particular operating point. Critical points I and II yield the highest required values for A1, A2 and A4. The problem is similar to that of sizing exchangers in a stream-split as described in Chapter 8.

A3 must be chosen so that:

1) The cost of an exchanger is assumed to $\alpha\ A^{0.6}$ therefore to find a minimum cost solution

$$\underset{A3}{\text{Min}} \sum_{i=1}^{4} Ai^{0.6}$$

2) The network can operate feasibly for the critical points I and II.

Using the procedures discussed in Chapter 8 the solution is found to be:

U.A1	= 1.838	U.A3	= 1.3
U.A2	= 0.667	U.A4	= 1.034

This network is 36.5% cheapter than that shown in Figure 7.10, which was the simple combined network, based on the cost of exchangers 1 to 4. Utility use is identical but there may be slight differences in the required areas of the utility exchangers, which could be considered when finding the optimum value of A3.

The performance of this optimum network at points I and II in the operating range is shown in Figure 7.13. Further checking reveals that this network can handle all operating conditions.

In this example although initial indications were that a combined network with optimizing control would be required, by treating the combined network structure as a new base network it has been possible to design a cheaper network (by approximately 36.5%) which requires regulatory control only, and hence does not require all exchangers to be fitted with a controlled bypass.

A combined network may not only prove effective when no network with the minimum number of units can operate feasibly over the entire operating range but also if excessively large exchanger areas are needed to make a network with the minimum number of units fully feasible. In such a case it may be possible by combining two networks to produce a cheaper final design.

Stream	T_i	T_t	F.Cp		
1	300	100	1.25	OR	0.76
2	300	100	1.1		1.4
3	150	250	1.0		
4	150	200	2.0		
5	110	210	0.4 to 0.5		

Table 7.3 Stream data used in a combined network example

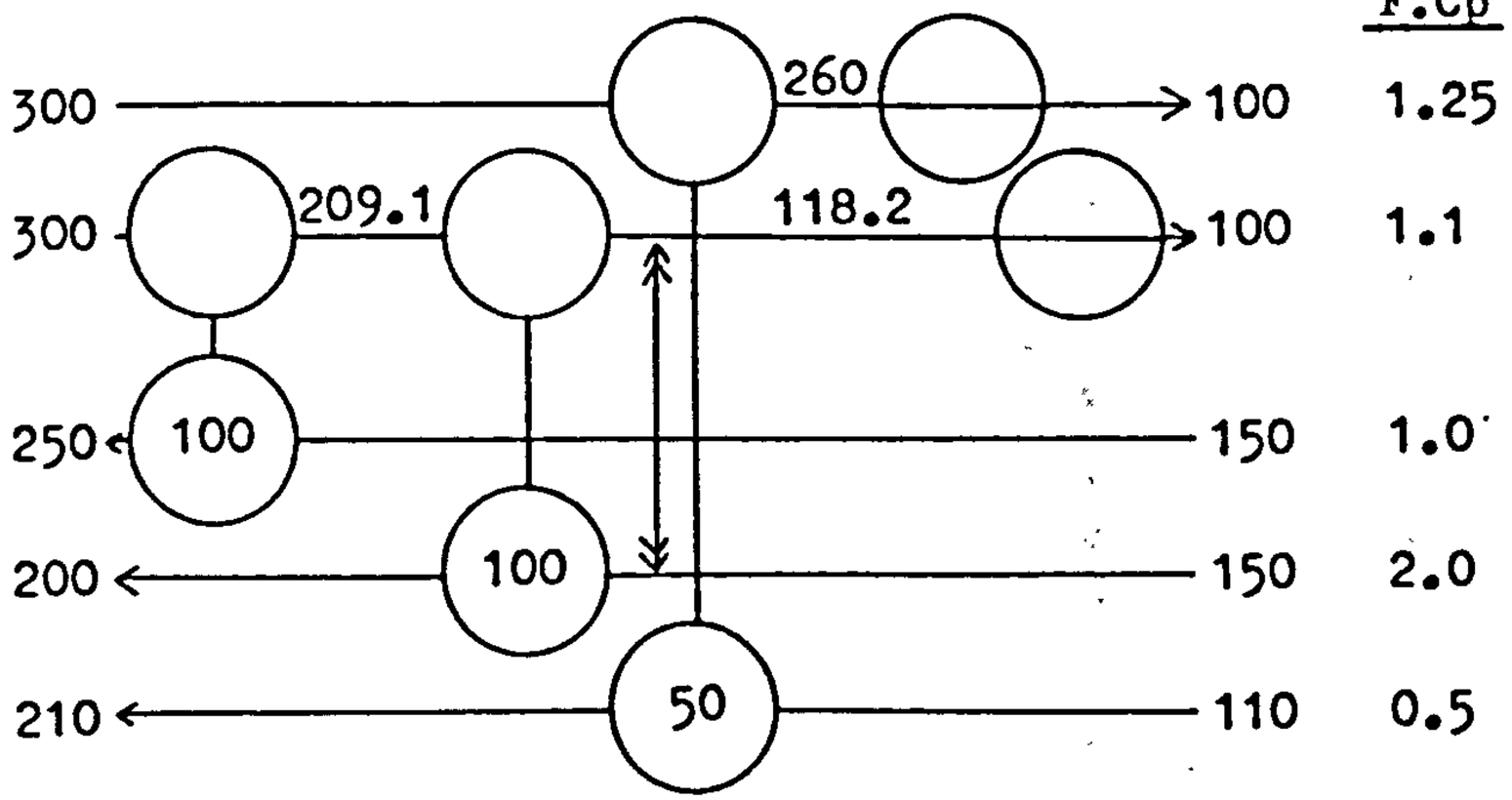
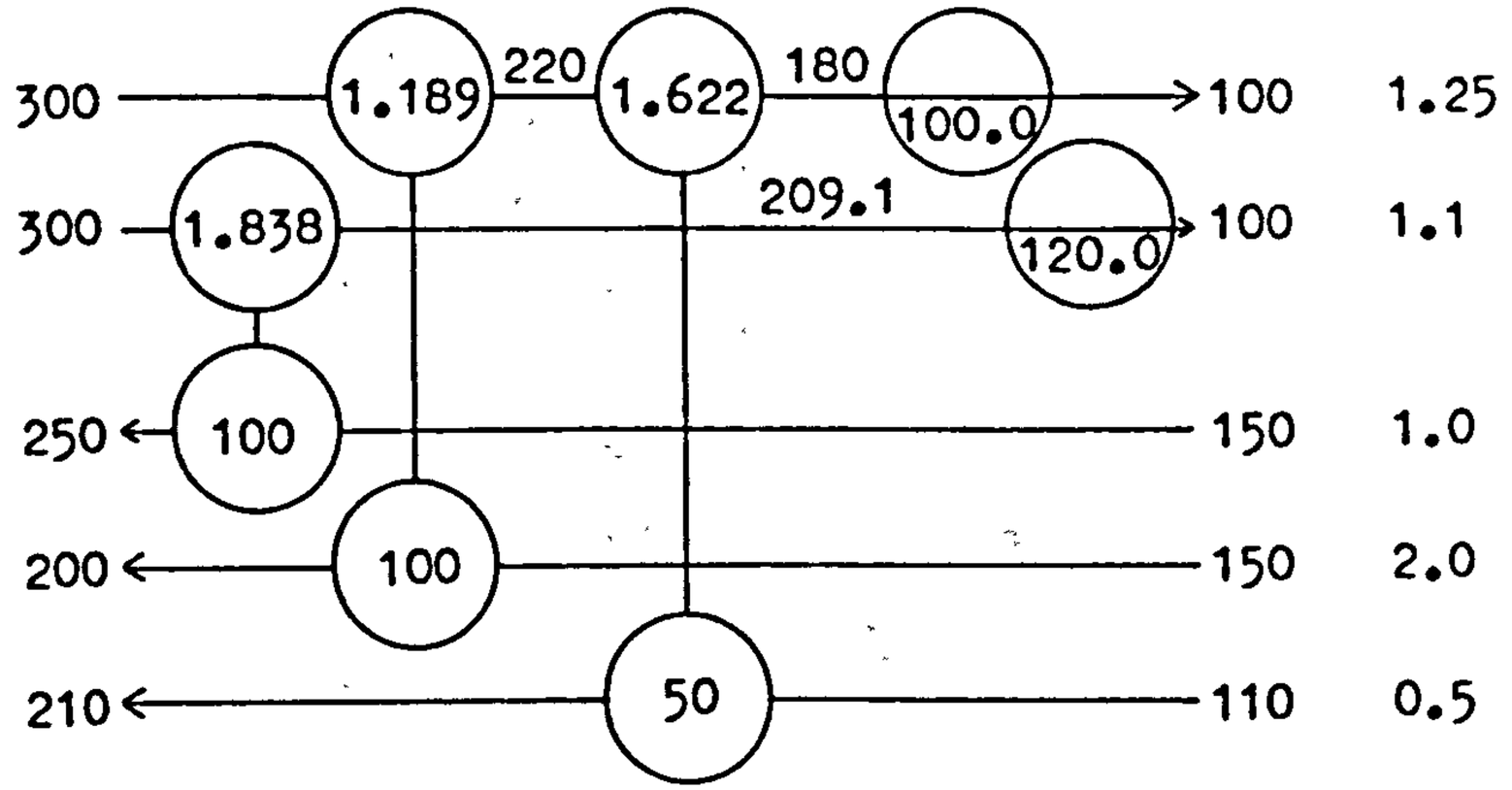
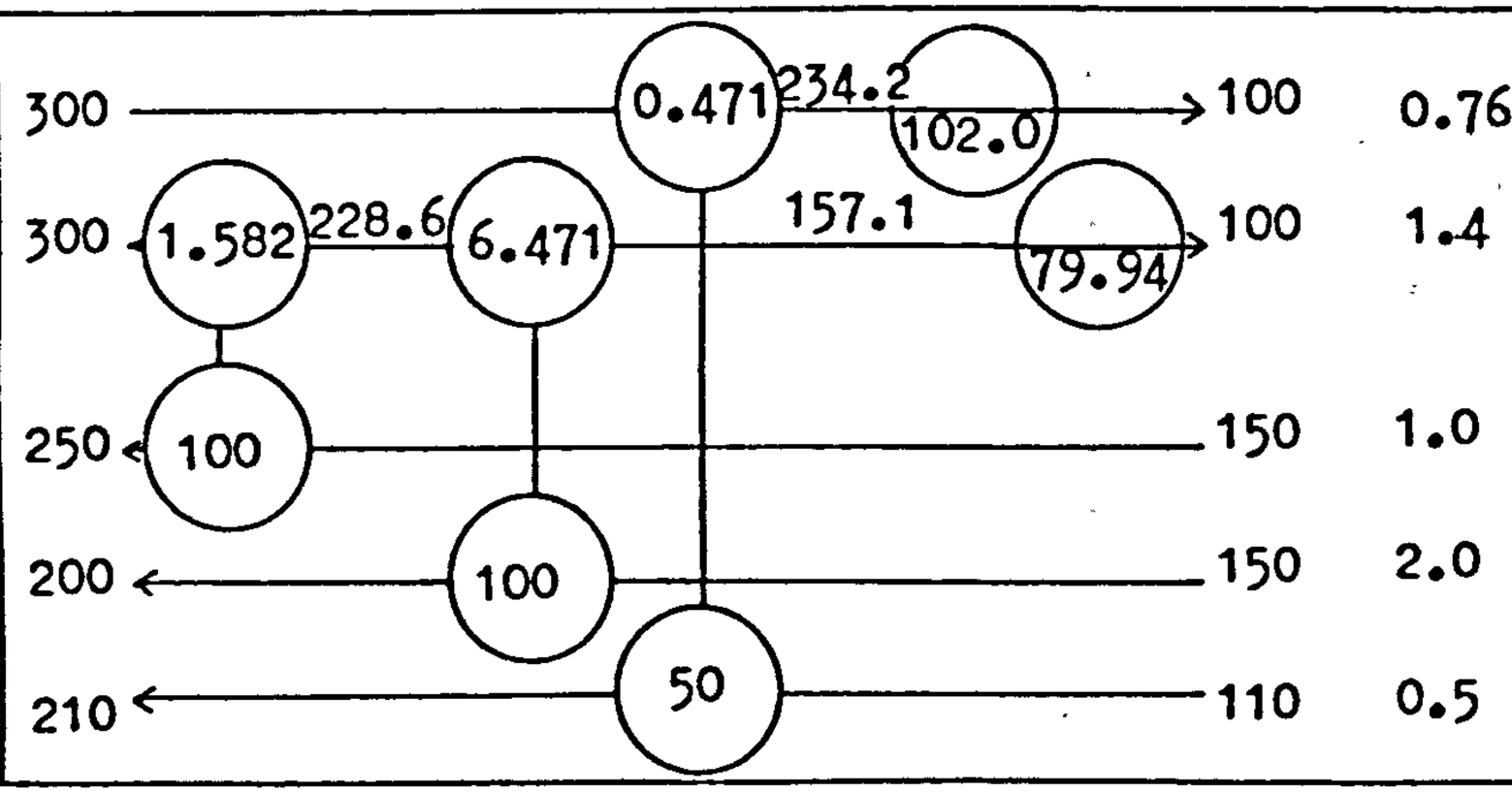
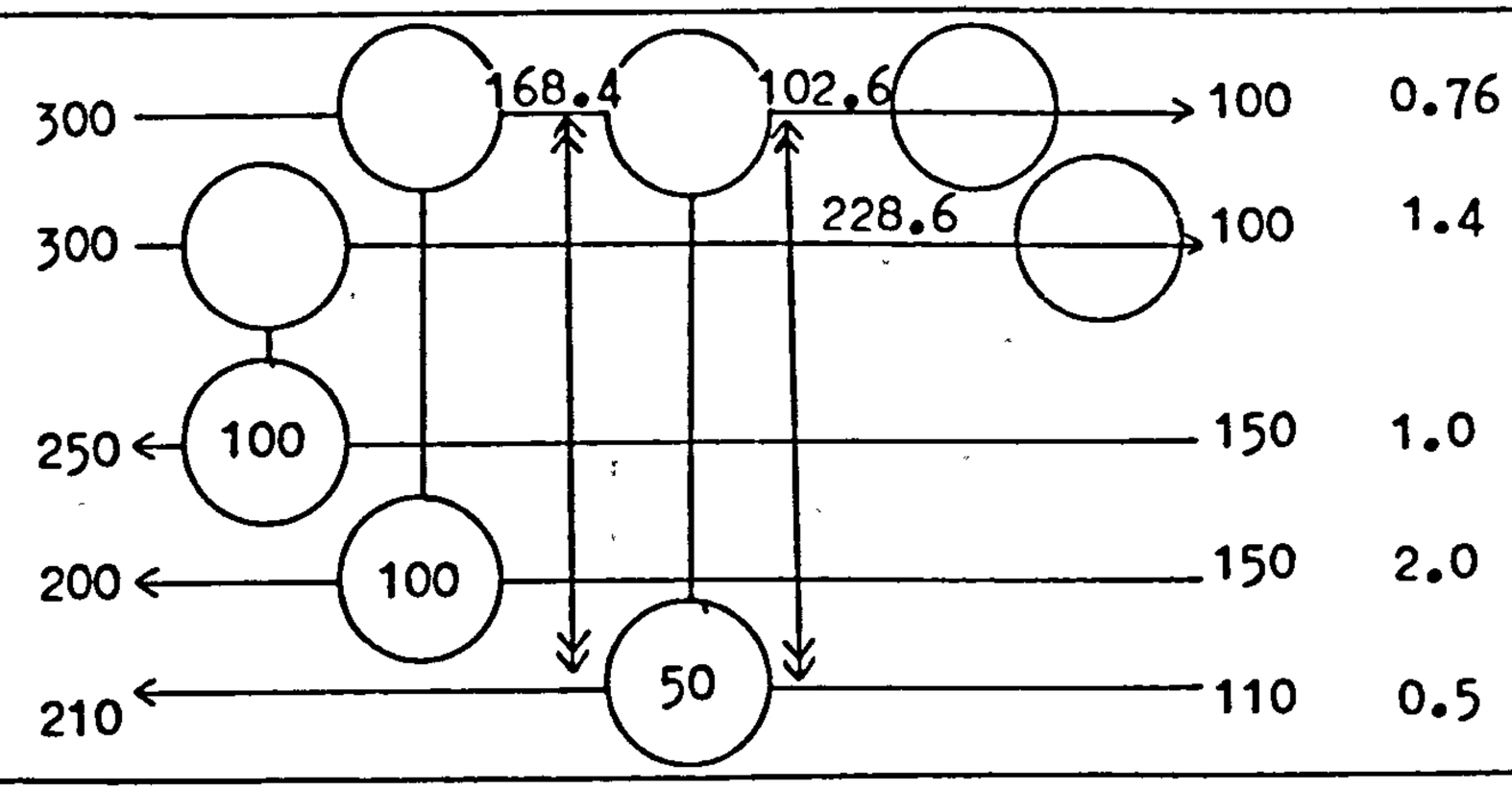
	Network A	Network B
Point I		
Point II		

Figure 7.9 The performance of networks A and B at operating points I and II.

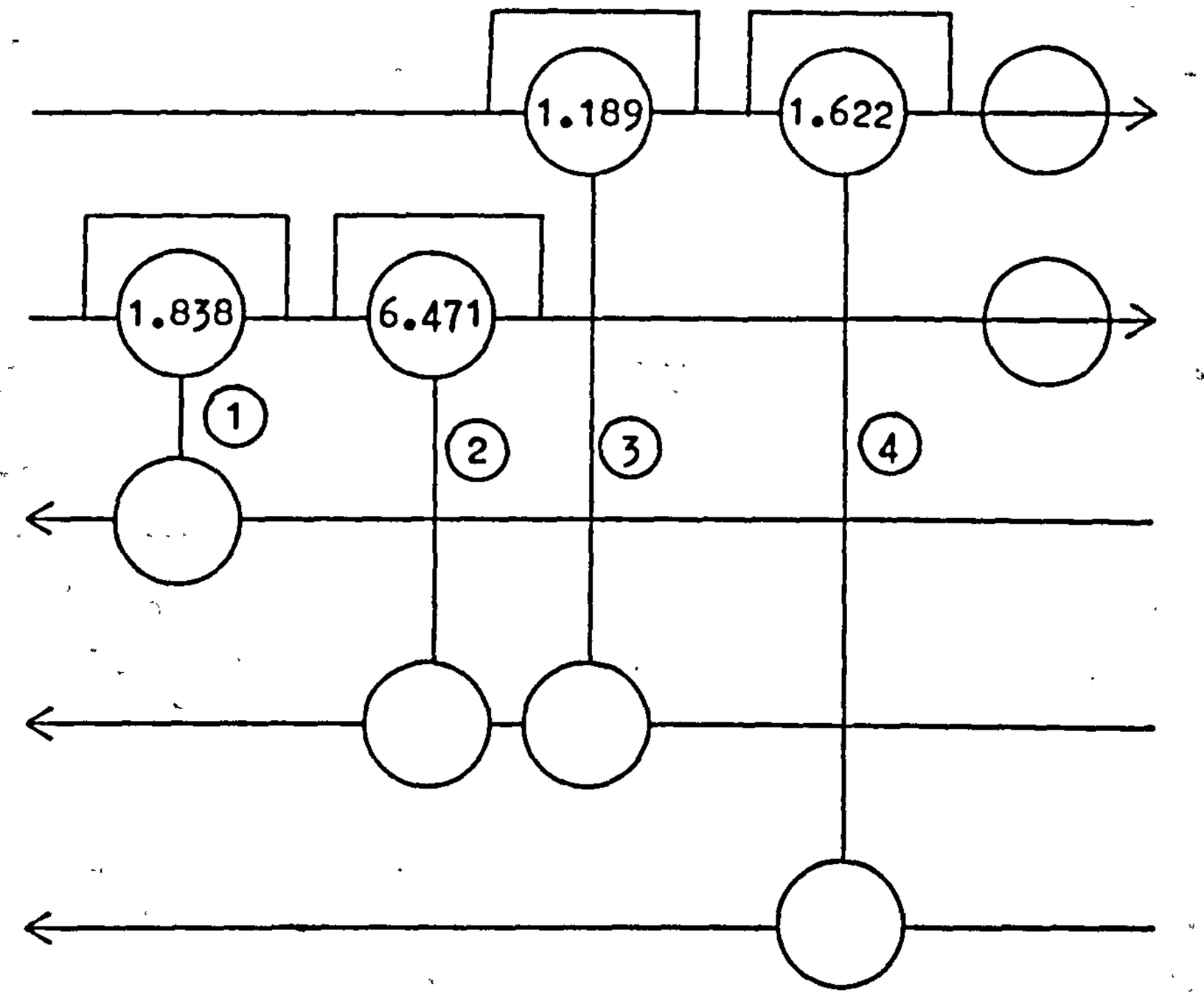
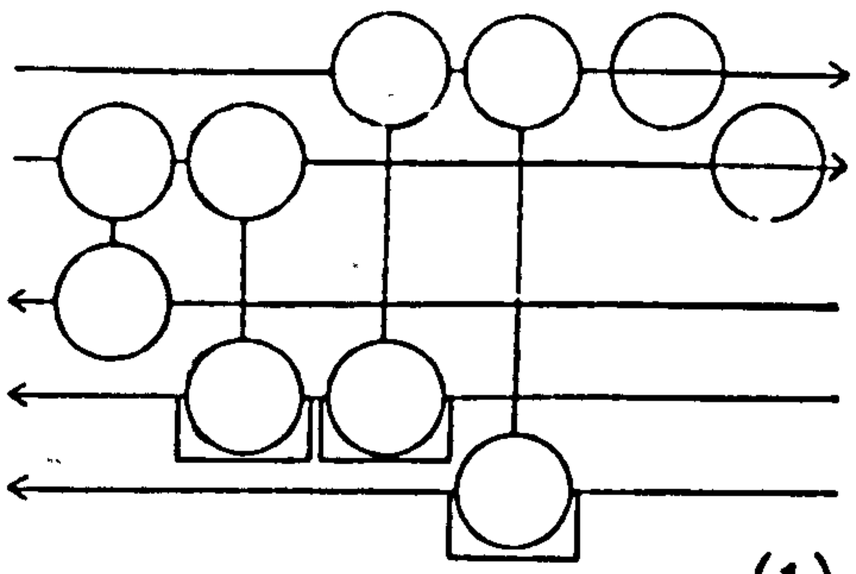
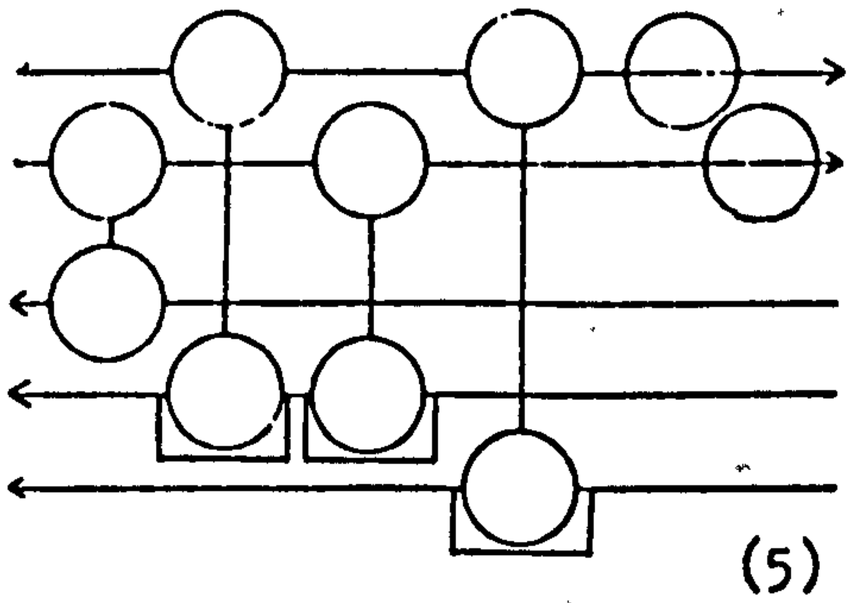
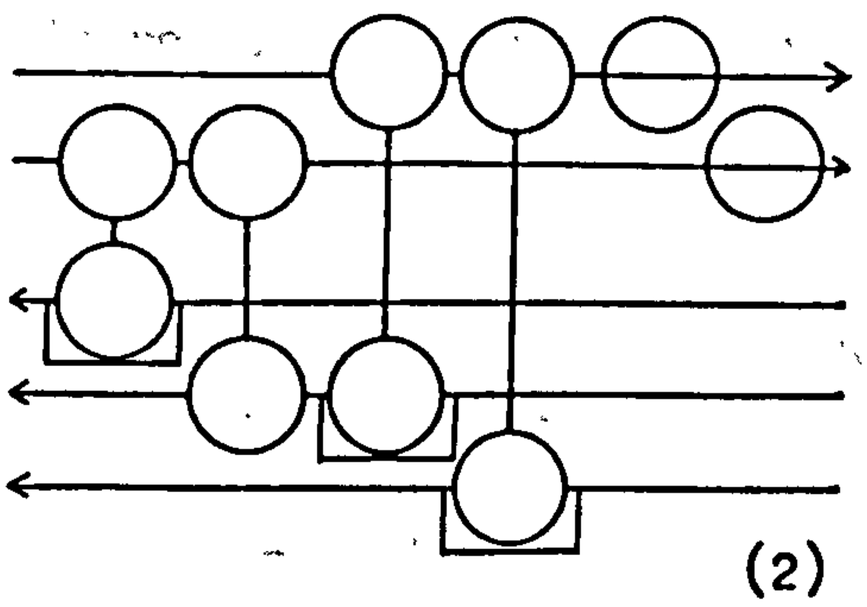
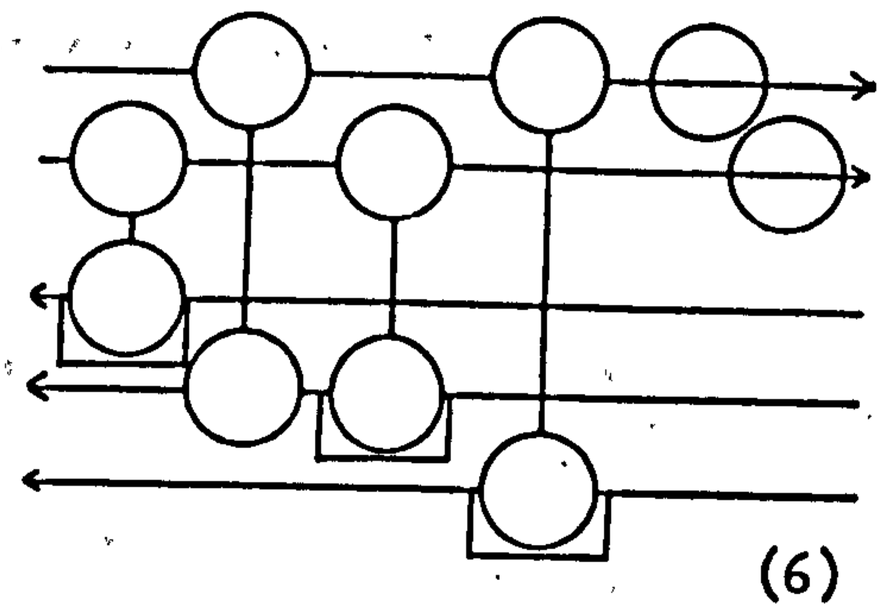


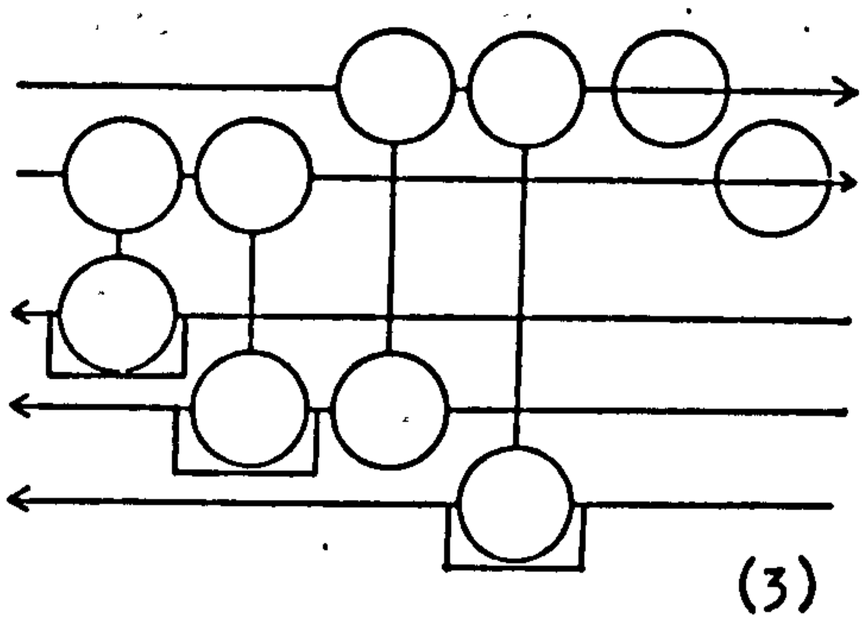
Figure 7.10 A combined network including networks A and B as special cases.

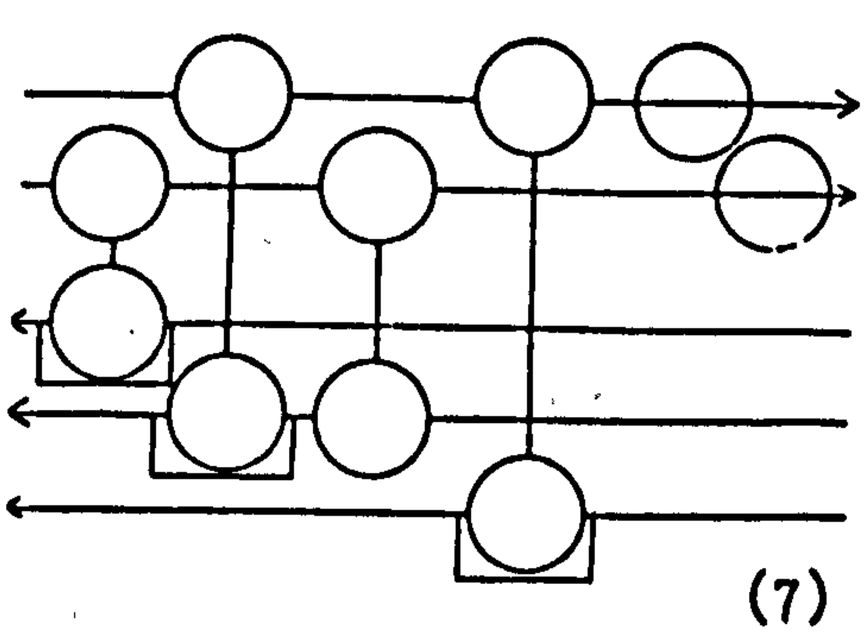


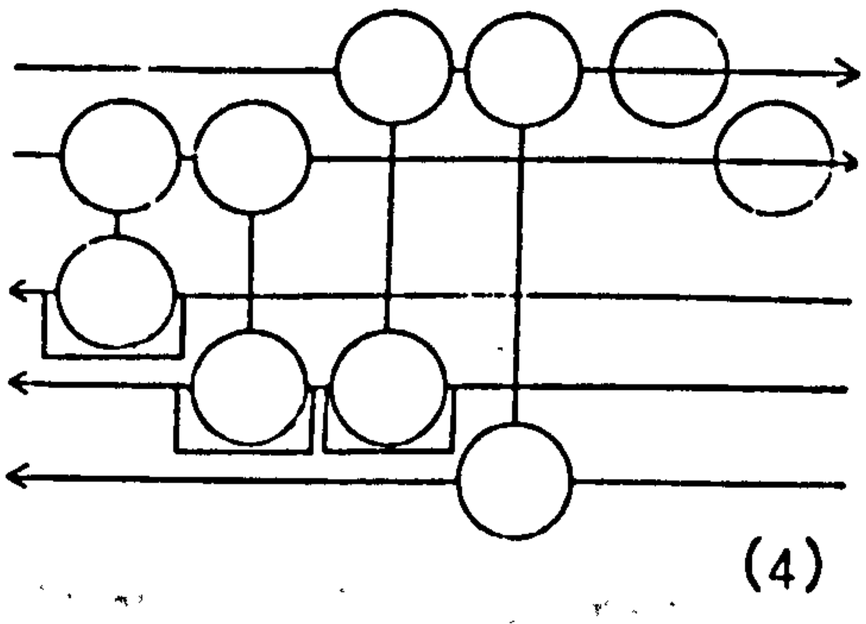












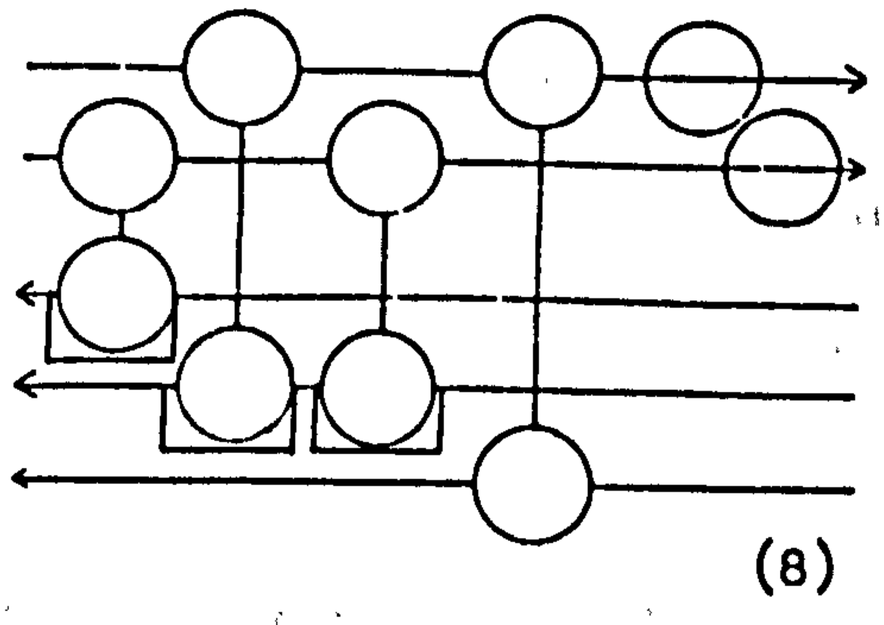


Figure 7.11 The eight alternative networks formed by combining networks A and B and assigning three bypasses.

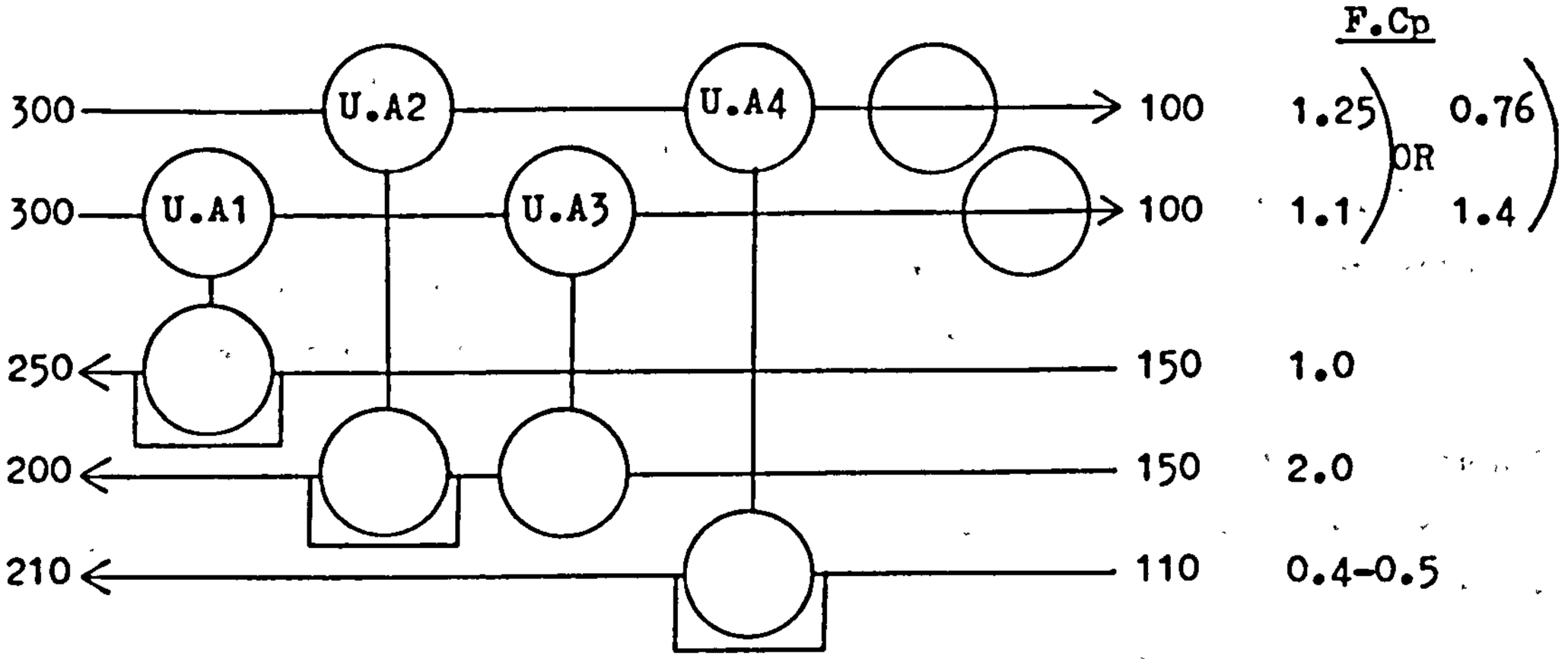


Figure 7.12 The most promising network alternative.

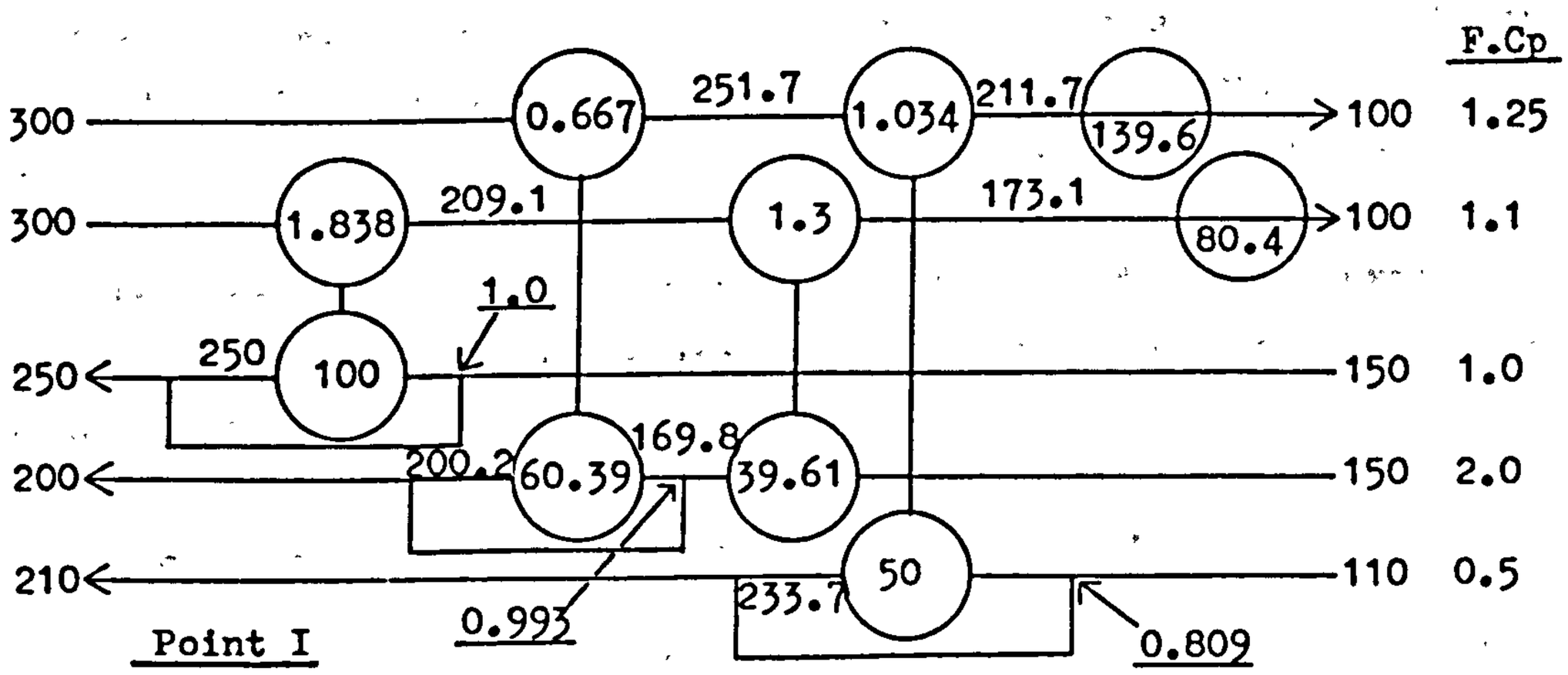


Figure 7.13 Performance of the optimum network at operating points I and II.

7.3 Pinched Problem 1

As has been noted previously the target used by Morari and co-workers that a network should maintain a fixed minimum allowable temperature approach throughout an operating range is inappropriate, and this can be demonstrated in the following example.

The network used is given in Figure 7.14 (a) and has been designed for a minimum temperature approach of 10. As two bypasses are present and the degrees of freedom of the network are one, one bypass must be used for regulatory control and one for optimizing control.

An operating range is formed by allowing the F.Cp values to vary by up to -0.1 of the nominal values given in Figure 7.14 (a). At a different operating point the network can perform as shown in Figure 7.14 (b). To maintain a minimum temperature approach, ΔT_{min}, of 10 both split fractions have had to be set to values less than 1.0.

For exactly the same operating conditions and network except for the fact that there is no bypass on exchanger 1 the performance is as shown in Figure 7.14 (c).

Comparing the performance in Figures 7.14 (b) and 7.14 (c)

1) The use of optimizing control in (b) to maintain ΔT_{min} = 10 results in a total utility requirementof 181.0.
   In (c) no optimizing control is needed and utility use is less, 175.9.

2) The ΔT_{min} value in (c) is 7.7 compared to 10 in (b). This does not mean that larger exchangers are required in (c), rather that advantage is being taken of exchangers which are already oversized for that particular duty.

A fixed ΔT_{min}, although useful in non-flexible design methods, is not a satisfactory utility use target for flexible designs. The application of the improved utility use target proposed in the present work may be demonstrated in a further number of pinched problem design examples.

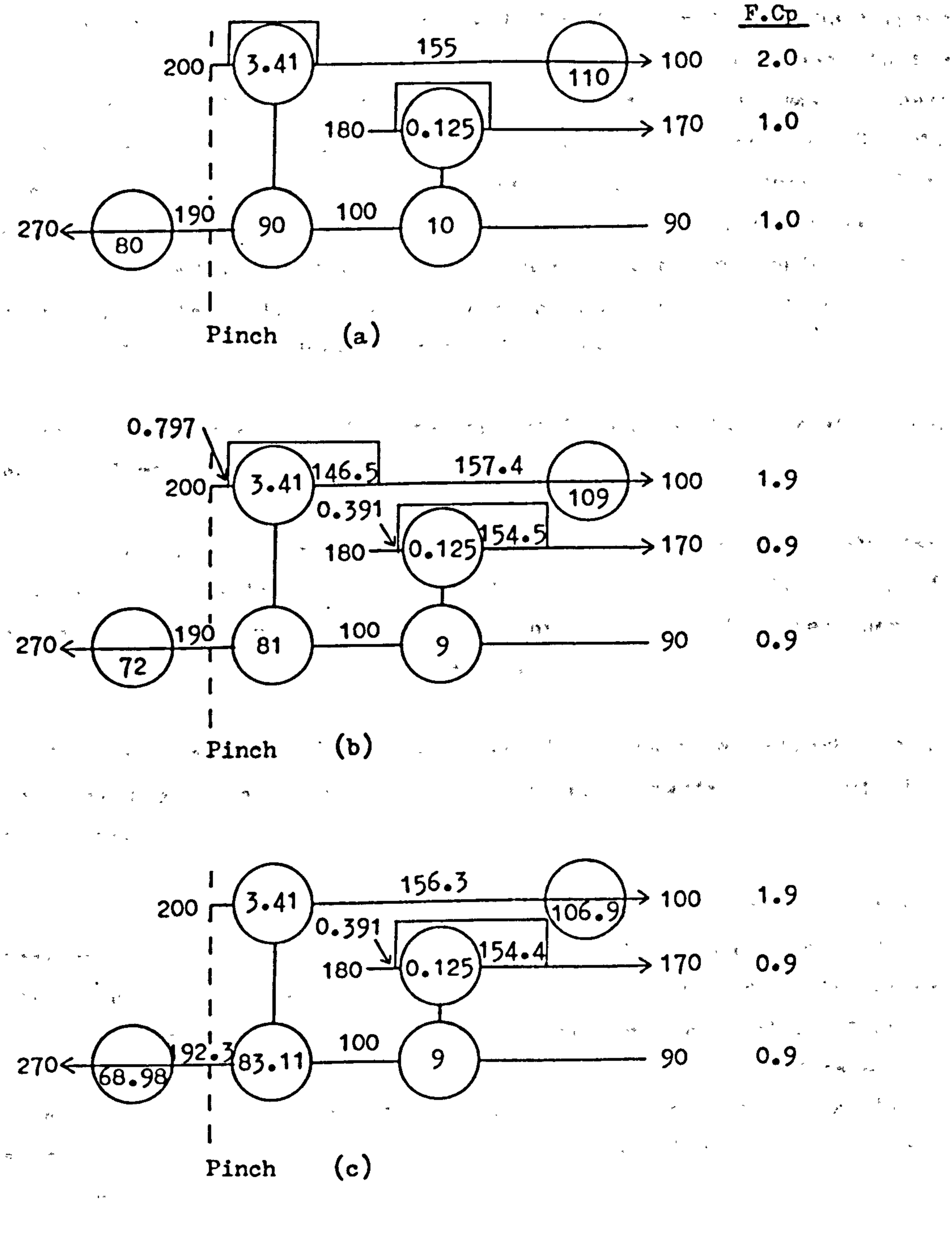


Figure 7.14 Pinched Problem 1.

7.4 Pinched Problem 2

The network used is given in the I.Chem.E. User Guide (1982) (page 43) with the units converted to be consistent with those used in the present work. The network has been designed to give maximum heat recovery based on a minimum allowable temperature approach of 18. The network is given in Figure 7.15 and the exchanger U.A values have been calculated for nominal conditions ('). The operating range has been formed by allowing hot stream F.Cp values to vary independently from 1.0 to 1.05 times the nominal values and cold stream F.Cp values to vary from 0.95 to 1.0 times the nominal values.

Utility use for nominal operating conditions is 99.9% greater than theoretical minimum use calculated for $\Delta T_{min} = 0$.

An analysis of the degrees of freedom indicates that only two exchangers need be bypassed to provide regulatory control. Also, if exchanger 1 is left unbypassed then the problem is badly posed therefore one of the bypasses must be exchanger 1.

The performance of the remaining three possible networks at a point I in the operating range, corresponding to maximum total utility use, is shown in Figure 7.16.

For operating point I both networks (b) and (c) show structural infeasibilities. Network (a) is feasible for these conditions and has a utility use of 21.7% greater than theoretical minimum which is better than for the nominal case design and therefore acceptable. Further study reveals that network (a) with the U.A values indicated is feasible over the entire operating range and that utility use is always less than or equal to the value which was acceptable for the nominal case design.

It is not clear, however, that further heat cannot be recovered for operating point I if all the exchangers are bypassed and optimizing control implemented with the objective of minimizing the utility use.

This problem was studied using the computer modelling program to simulate optimizing control for the operating point I. As noted in Chapter 3 for each move of the optimizing controls the regulatory controls must be able to bring the network to a steady state therefore from Figure 7.16 it is clear that the bypasses on exchangers 1 and 4 must be used for regulatory control, as in network (a), with the additional bypasses on exchangers 2 and 3 used for optimizing control. It is found that the minimum total utility use occurs when exchanger split fractions 2 and 3 are equal to 1.0, which is equivalent to having no bypass present. This is expected as to use the minimum amount of utilities the maximum amount of heat possible must be transferred in the stream/stream exchangers. To achieve this the maximum possible flow must pass through the exchangers and in the case of optimizing bypasses there is no reason why the exchanger split fractions should not be 1.0. For this reason network (a) in Figure 7.16 actually uses the minimum possible amount of utilities for the particular set of U.A values chosen for all operating points. This is achieved automatically without the need for optimizing control.

In summary, network (a) in Figure 7.16

1) Can operate over the entire operating range
2) Requires only regulatory control, no optimizing control
3) Automatically achieves minimum possible utility use for all operating conditions (for the given network)

Thus if a network has zero degrees of freedom not only is the control task purely regulatory and therefore easier to implement but there may be no penalty in terms of extra utility use. As noted in Chapter 5 the development of a design method which guarantees flexibility only for networks with zero degrees of freedom is less restrictive than might be at first thought.

In this example optimizing control is superfluous and, as can be seen from Figure 7.16, if the wrong choice is made of which bypasses to use for optimization, potentially harmful to the operability of the network.

To determine whether the degrees of freedom of a network may generally be restricted to zero further examples are studied.

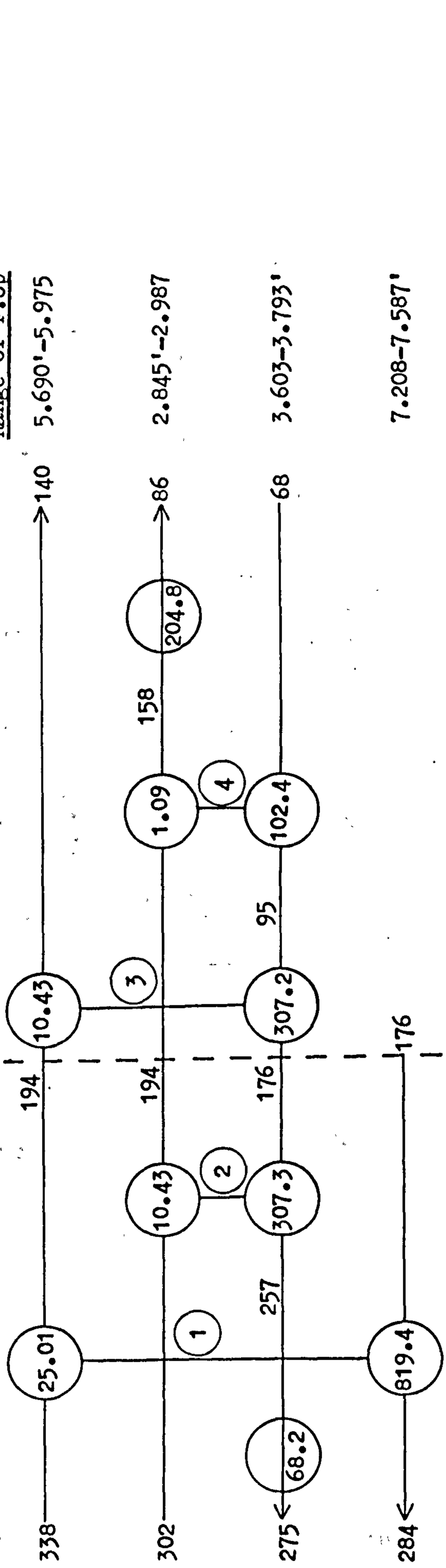


Figure 7.15 The network used in Pinched Problem 2.

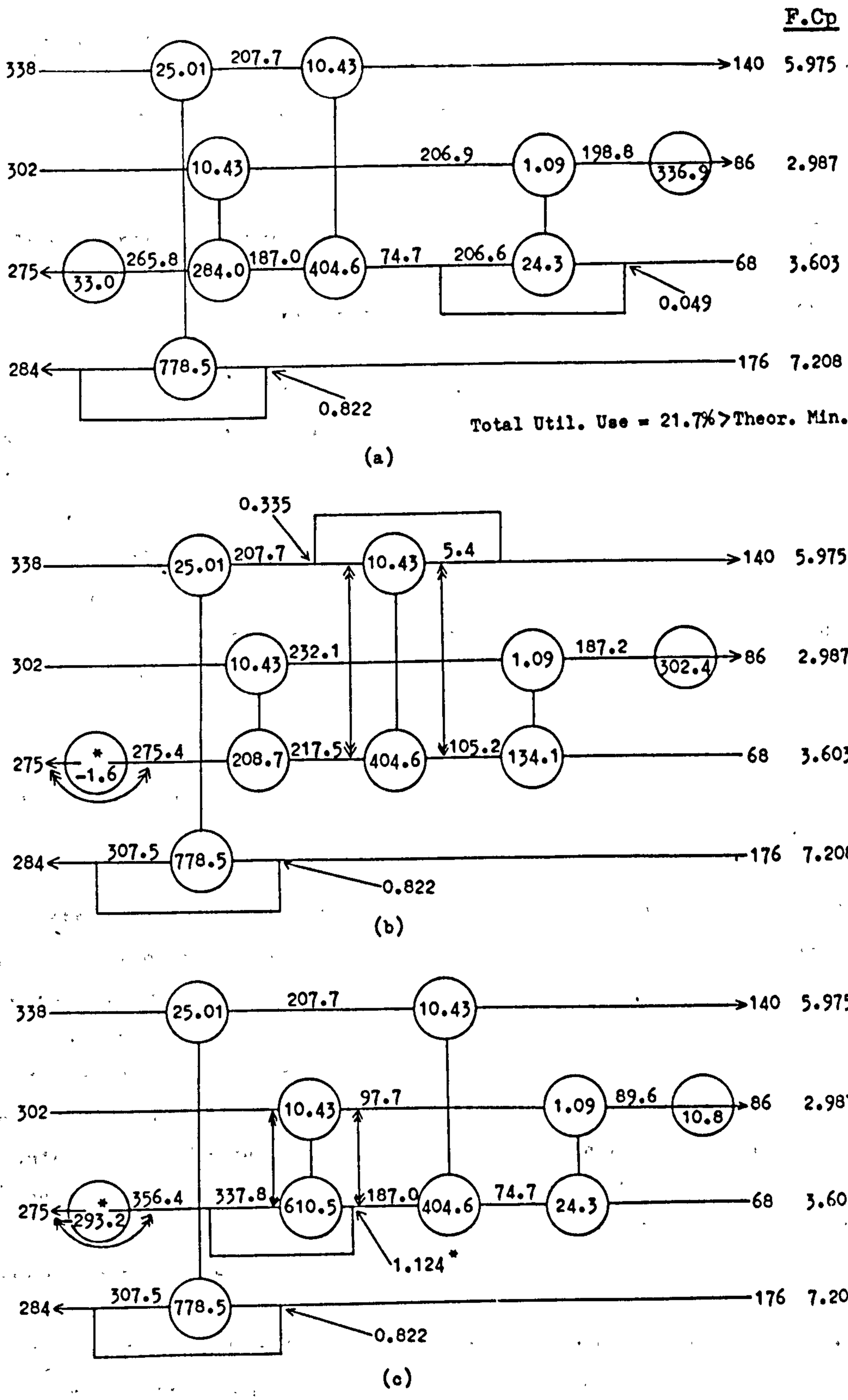


Figure 7.16 The performance of three alternatives at operating point I.

7.5 Pinched Problem 3

The network, given in Figure 7.17, is taken from unpublished work by Linnhoff and is designed to recover the maximum amount of heat based on $\Delta T_{min} = 18$.

The operating range indicated has been formed by allowing hot stream F.Cp variations from 0.95 to 1.0 times the nominal values and cold stream F.Cp variations from 1.0 to 1.05 times the nominal values.

For regulatory control purposes only one exchanger need be bypassed, therefore three alternatives exist. These alternatives are studied at an operating point I corresponding to minimum possible utility use, based on percentage use greater than theoretical minimum for network (a), the results being given in Figure 7.18. Network (c) cannot operate feasibly for these operating conditions as exchanger 3 is undersized for the required duty, indicated by the required exchanger split fraction of greater than 1.0.

Alternatives (a) and (b) are both feasible but network (a) has better utility use figures, 5.8% greater than theoretical minimum compared with 17.8%.

Further study reveals that both networks (a) and (b) are feasible over the entire operating range but that the utility use, measured as a percentage greater than theoretical minimum, differs as shown in Table 7.4. The operating points producing minimum and maximum utility use are different for networks (a) and (b). The utility use must be the same for the nominal operating conditions as these are the conditions for which the exchanger U.A values have been determined, therefore all exchanger split fractions must be 1.0 and all three alternative networks (a), (b) and (c) will have the exchanger heat load and intermediate temperatures given in Figure 7.17.

As in Pinched Problem 2 if all the exchangers are bypassed and optimizing control simulated it can be demonstrated that alternative (a), which clearly has the better utility use figures, in fact achieves

the minimum possible utility use for all operating points, and does not require optimizing control. This confirms the conclusion that optimizing control may not be needed if a network features more than the minimum number of units. Further, if all exchangers were bypassed and optimizing control used to manipulate two split fractions Figure 7.18 demonstrates that economic penalities would result from an incorrect choice of which bypasses to use for optimization.

1) If bypasses on exchangers 2 and 3 were used for optimizing control with the objective of minimizing utility use they would be fully open for all operating conditions and therefore superfluous.

2) If bypasses on exchangers 1 and 3 were used for optimizing control and the bypass on exchanger 2 for regulatory control then either the bypasses on exchangers 1 and 3 would be fully open for all operating conditions but, as this corresponds to network (b), the minimum possible utility use for the given U.A values would not be achieved, or the optimizing/regulatory system would attempt to duplicate case (1) above.

3) If bypasses on exchangers 1 and 2 were used for optimizing control then they must be manipulated not only to give minimum possible utility use but also to enable the regulatory controller on exchanger 3 to bring the target temperature of stream 1 to its required value. This is a very difficult control objective in this case as it can be seen from network (c) in Figure 7.18 that there are some values of optimizing exchanger split fractions, in this case equivalent to both set at 1.0, for which the regulatory system will be unable to bring the target temperature to its desired value. The utility use figures in this case are also worse than the other two options.

If network (b) is preferred, possibly due to dynamic control considerations, it will be seen that the maximum utility use is 34.8%

greater than theoretical minimum and is greater than that which was acceptable for nominal condtions. The designer may judge the violation to be negligible or redesign the network to achieve acceptable utility use. If possible it is better to design the original base network for a point having the 'worst' utility use figures. This is simulated in this work by choosing operating ranges such that the nominal designs, which are generally taken from the work of Linnhoff and co-workers, feature the worst utility use figures. In practice an operating range will be given and the designer may be able to estimate a priori from the problem data a set of conditions providing the most severe test for heat recovery. In some cases, as for network (b) above, maximization of the utility use constraint function will show the guess to have been incorrect and will identify the actual critical point.

From the design examples presented so far there would seem to be no method of determining a priori from the network structure or problem data which exchangers should be left unbypassed for any particular problem. This conclusion is further reinforced by a final example.

Network	Maximum Utility Use	Minimum Utility Use	Utility Use at Nominal Conditions
(a)	28.5	5.8	28.5
(b)	34.8	13.6	28.5

Table 7.4 Comparison of the utility use of networks (a) and (b)

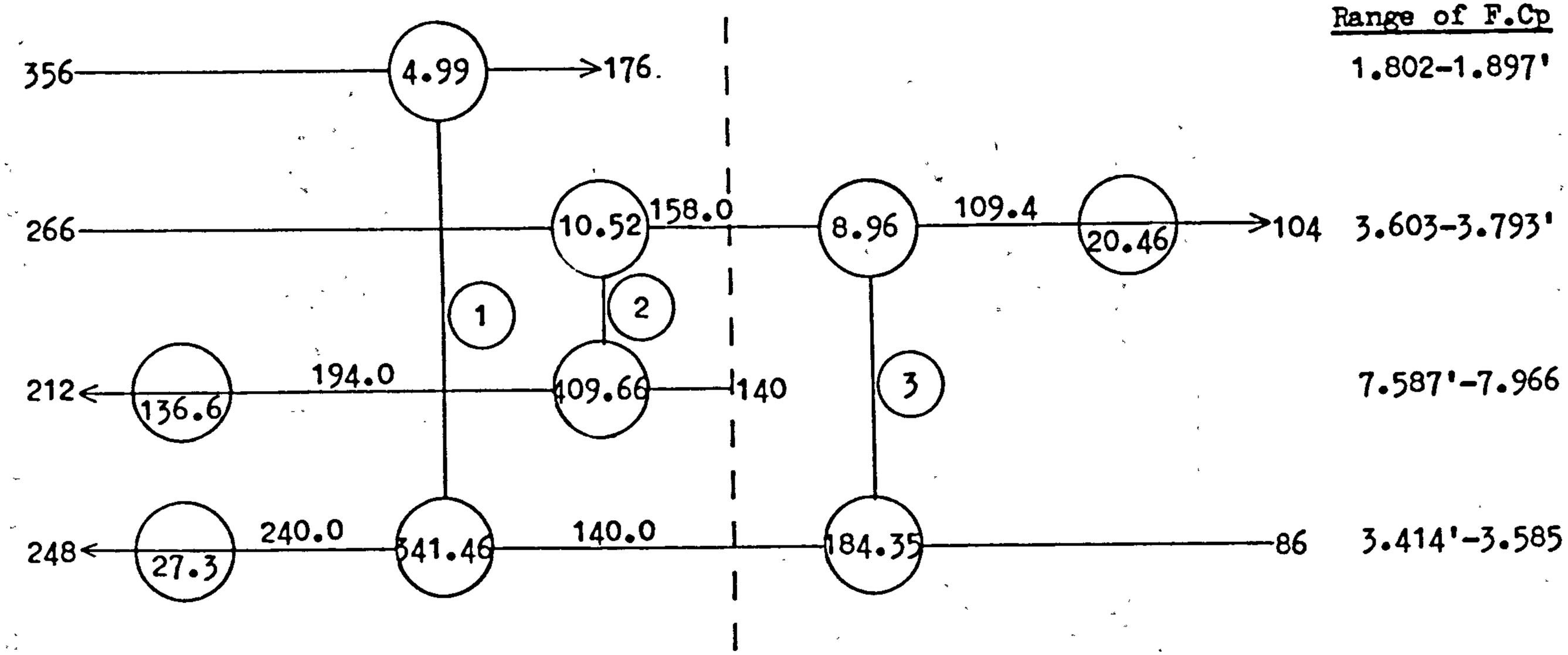
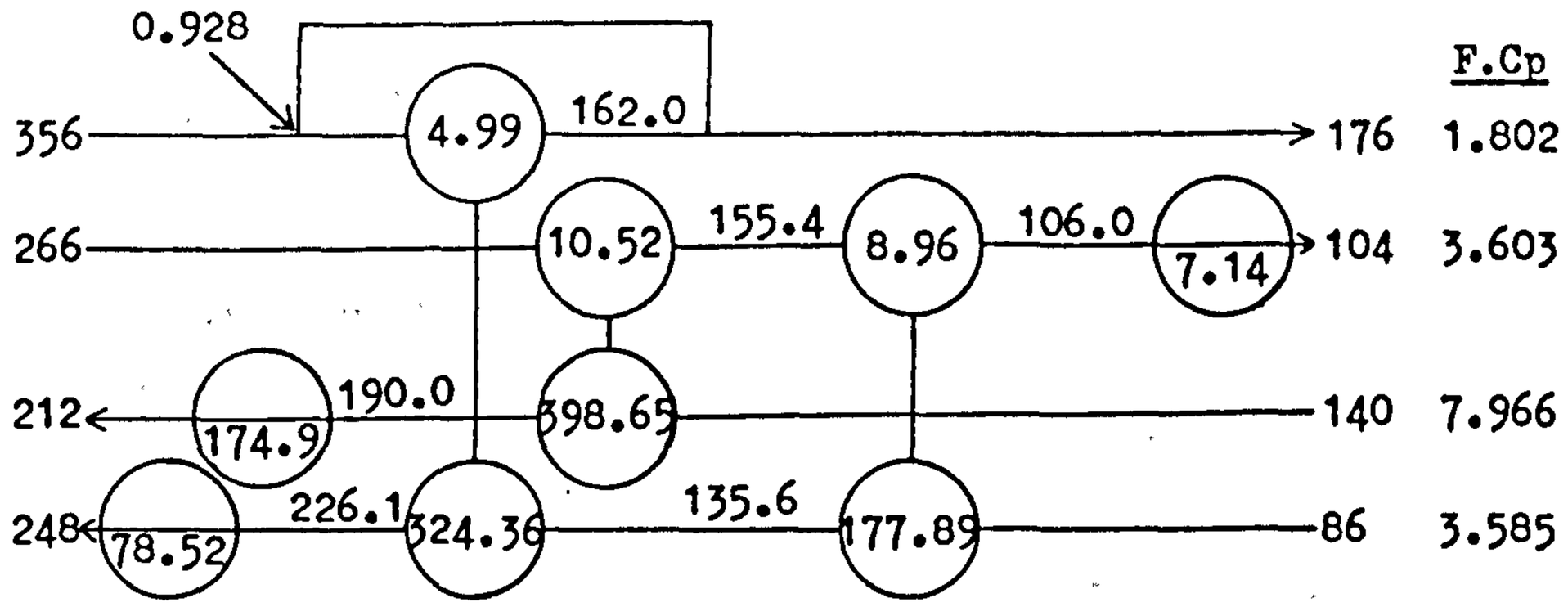
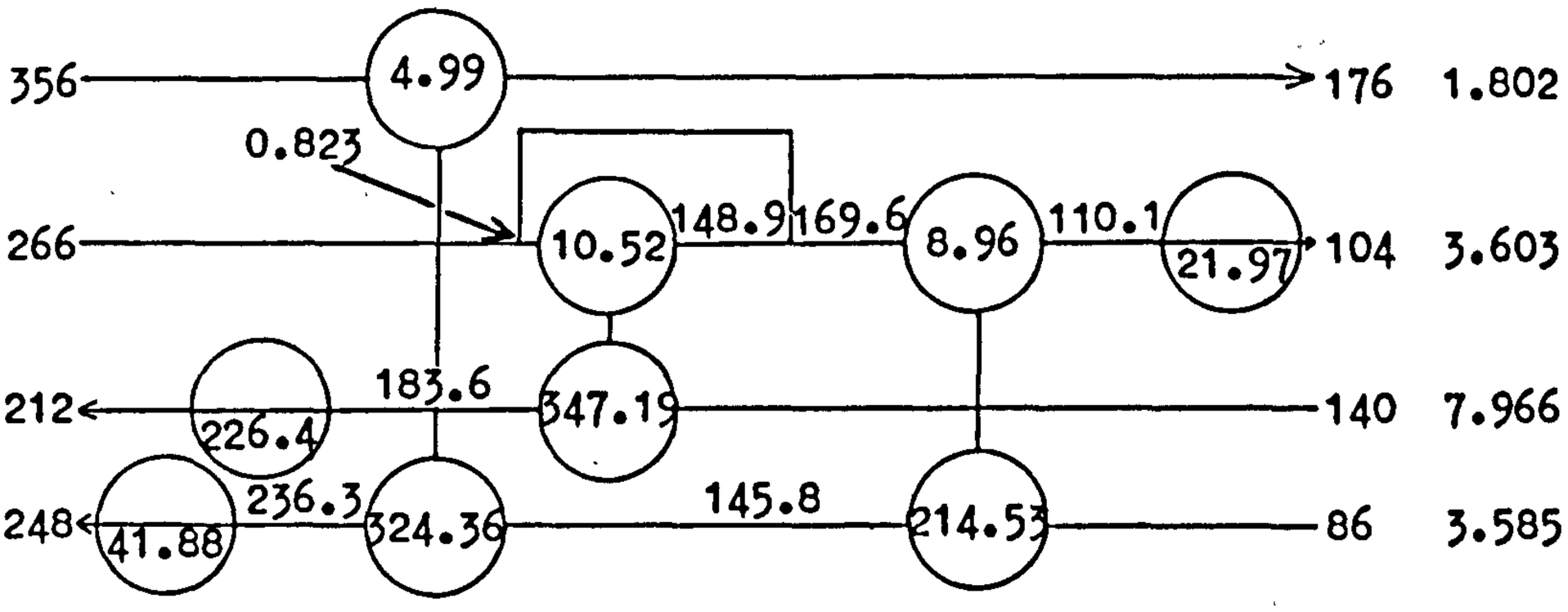


Figure 7.17 The network used in Pinched Problem 3.



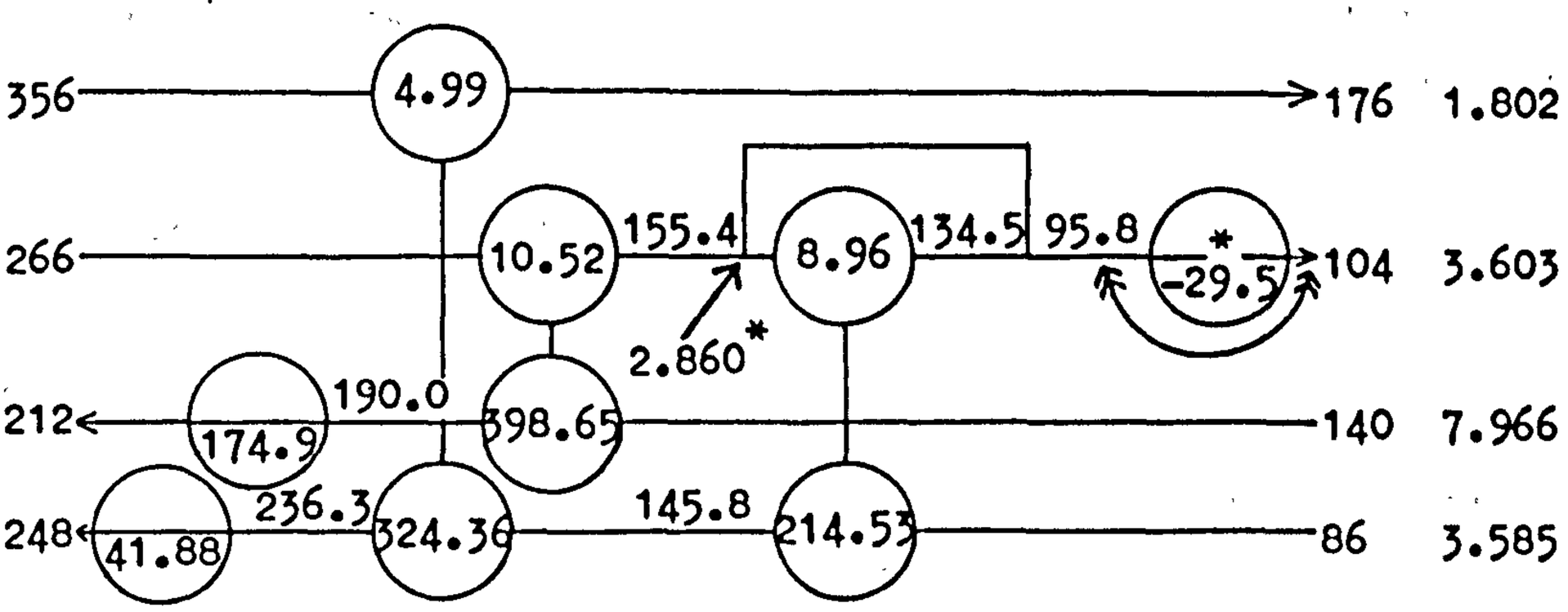
Total Util. Use = 5.8% > Theor. Min.

(a)



Total Util. Use = 17.8% > Theor. Min.

(b)



(c)

Figure 7.18 The performance of three alternatives at operating point I.

7.6 Pinched Problem 4

The network given in Figure 7.19 is taken from unpublished work by Linnhoff. The operating range indicated has been formed by allowing hot stream F.Cp variations of 0.95 to 1.0 times the nominal values and cold stream F.Cp variations of 1.0 to 1.05 times the nominal values. Only one bypass needs to be introduced to provide regulatory control therefore four alternative networks exist. The performance of the four alternatives at a point I in the operating range is given in Figure 7.20.

At operating point I networks (c) and (d) both show sizing infeasibilities with exchangers 3 and 4 respectively undersized. Network (b) shows a structural infeasibility with a negative heat load required in exchanger 2. Network (a) can operate feasible for these operating conditions and further investigation reveals that the network is feasible for the entire operating range with the utility use always better than or equal to that for the base design. Utility use for network (a) is also the minimum possible for the given exchanger areas.

Thus in this example it has been demonstrated that a correct bypass placement ensures maximum possible heat recovery and operation over the full operating range without changing the base design whereas an incorrect bypass placement would require oversizing of exchangers or changing the network structure to ensure operation over the full range of conditions. This example confirms the observation made by Kittrell and Watson (1966) that "uncertainties do not necessarily justify over-capacity, they justify a good strategy for selecting the initial design".

If the operating range is changed so that hot stream F.Cp values vary between 1.0 and 1.05 time the nominal values and cold stream F.Cp values vary between 0.95 and 1.0 times the nominal values then Figure 7.21 shows the performance of the same four alernative networks at an operating point I' in the new range. Network (a) shows a sizing

infeasibility at this point with exchanger 1 being undersized. In the case of network (b) this is one of the rare occasions when the equation solving routine used in the computer program fails to find a solution, this indicates, however, that in operation network (b) would certainly be unable to meet the required target temperatures. Networks (c) and (d) both operate feasibly at this point with virtually identical total utility use even though heat loads for individual cooling utility exchangers vary, hence different areas will be required in each case. Further study reveals that networks (c) and (d) are feasible for the entire operating range and have the same total utility use for all operating conditions. The utility use of 87.8% greater than theoretical minimum at point I' is greater than that for the base case and therefore the designer must decide whether the maximum utility use value is acceptable. If it is not networks (c) and (d) must be resized to recover an acceptable amount of heat and network (a) may also be resized. In this case it is unclear which of the networks would prove to be cheaper.

This example demonstrates that the choice of which exchangers to bypass is dependent on the operating range used. For the first operating range considered, network (a) was the only one which could operate over the full operating range without any oversizing. For the second operating range, networks (c) and (d) could operate over the full range without oversizing (with, however, higher than nominal utility use) but network (a) would require ovesizing simply to ensure that target temperatures could be met.

Further study also reveals that the choice of which exchangers to bypass is also dependent on the actual area of the exchangers used. For example network (a) in Figure 7.20 was fully feasible but if the exchangers were resized to recover more heat, for exactly the same operating range, then configuration (a) might become inadequate and (b), (c) or (d) might be the best choice.

A number of conclusions can be drawn from the six example problems above and used in other design problems.

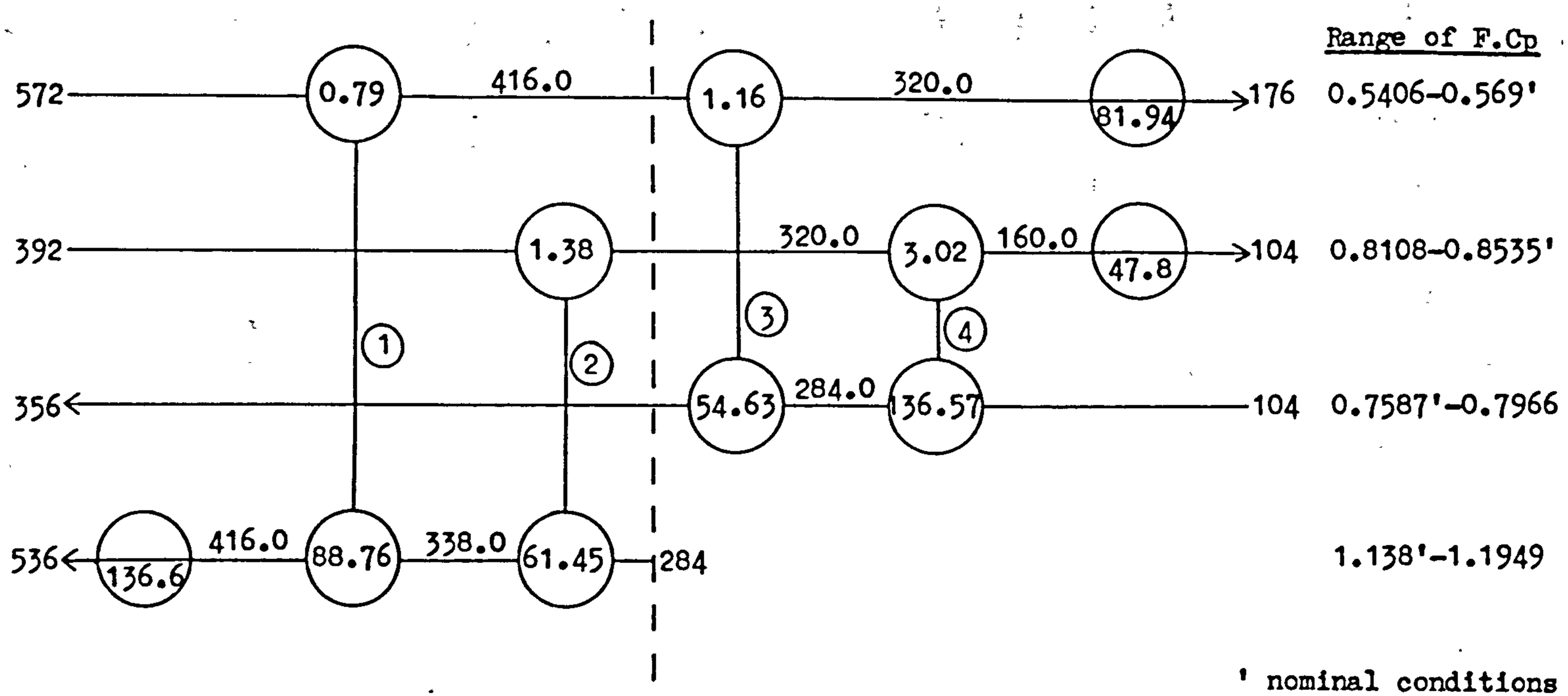
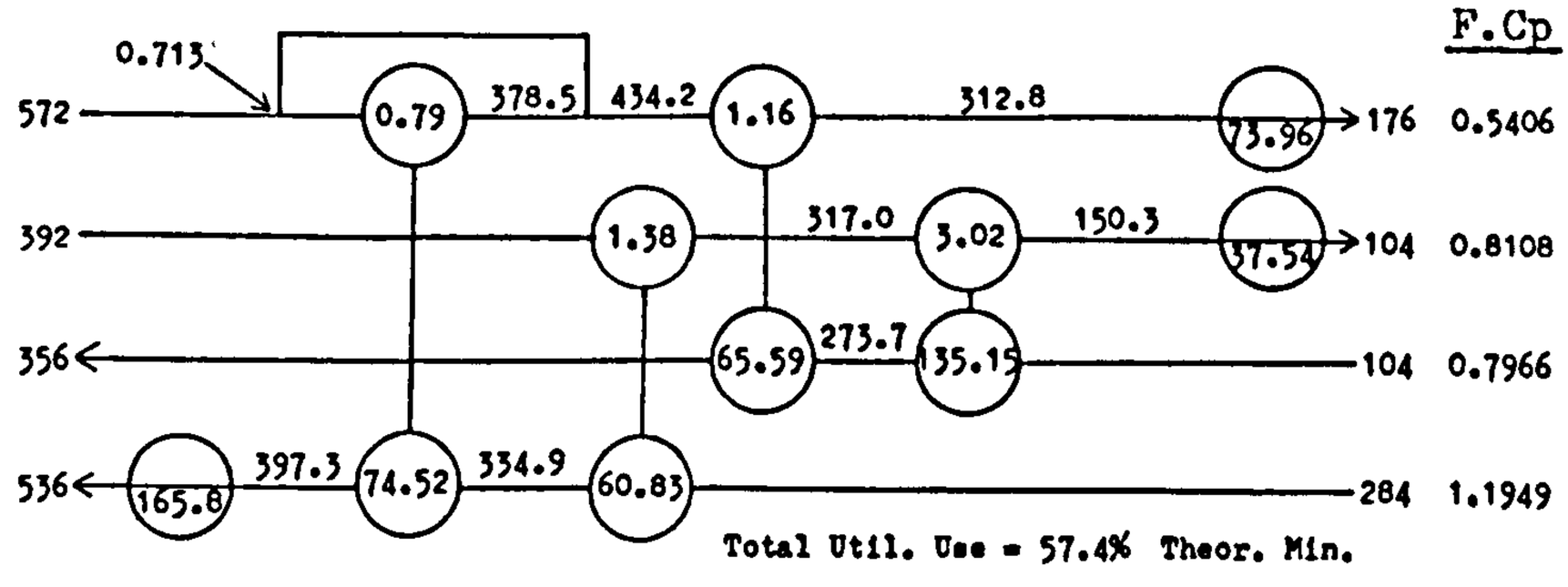
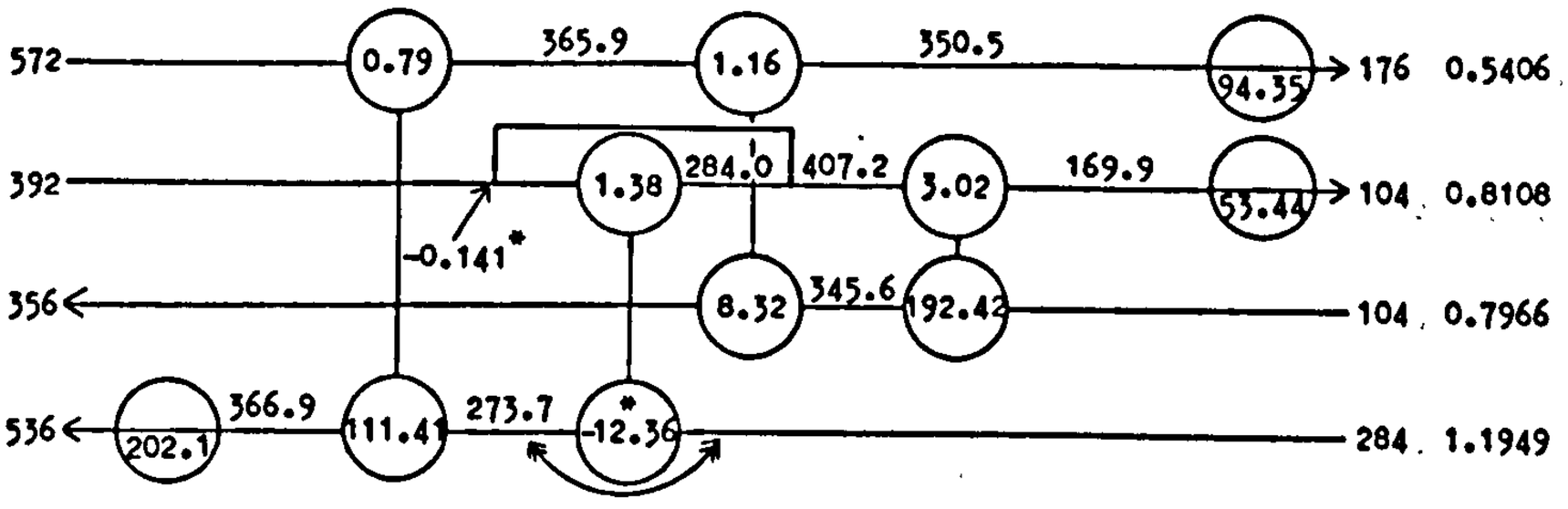


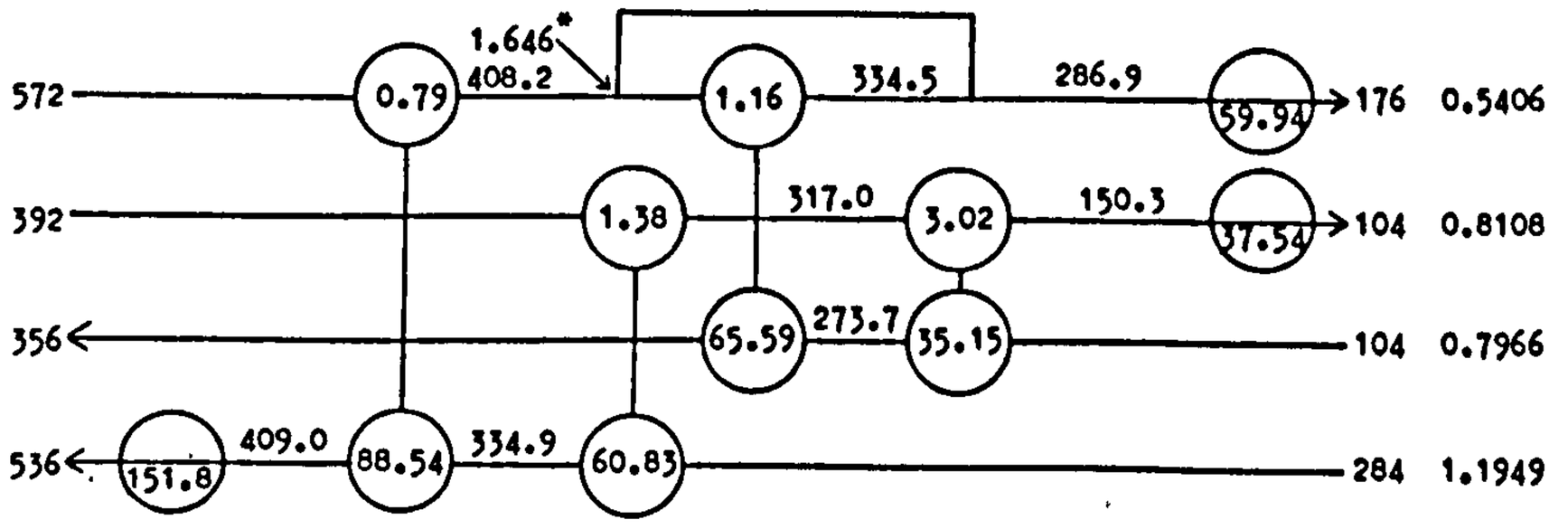
Figure 7.19 The network used in Pinched Problem 4.



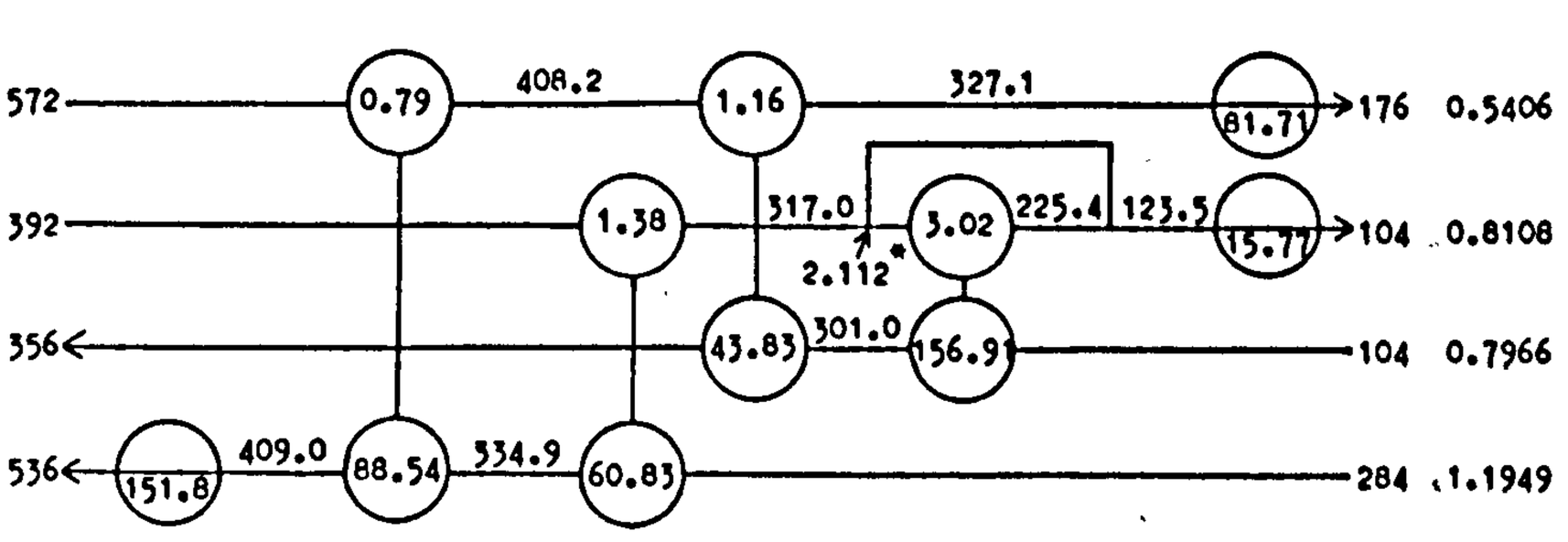
(a)



(b)

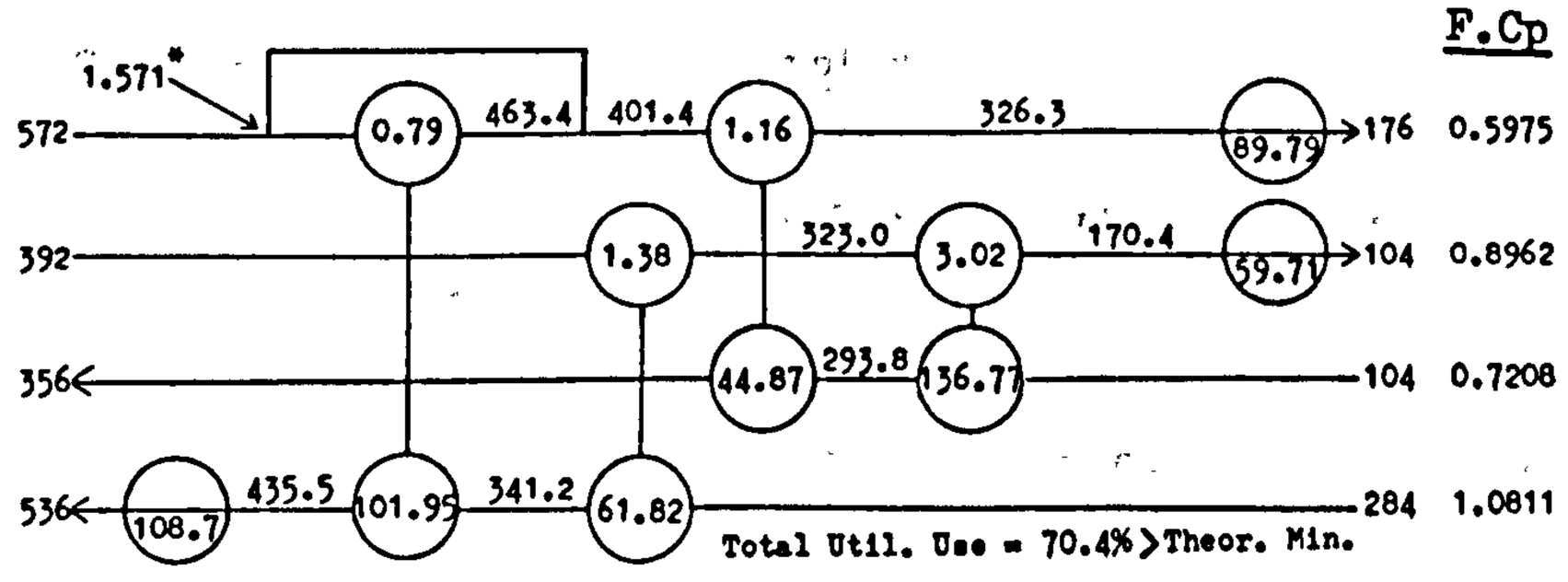


(c)

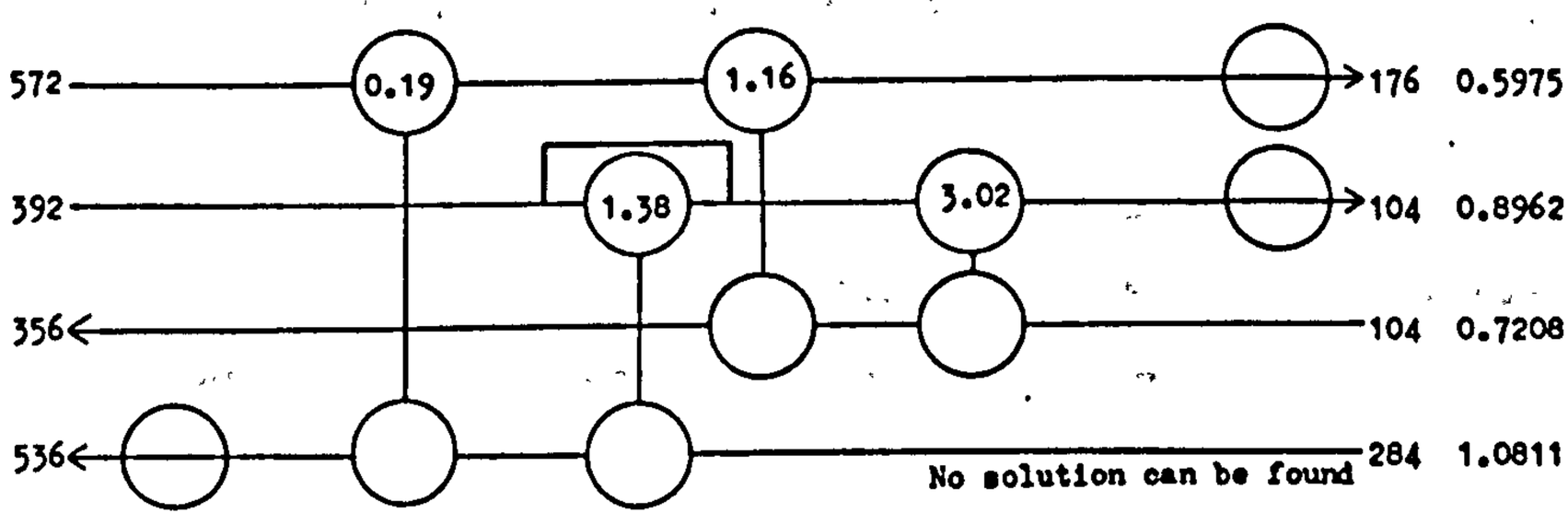


(d)

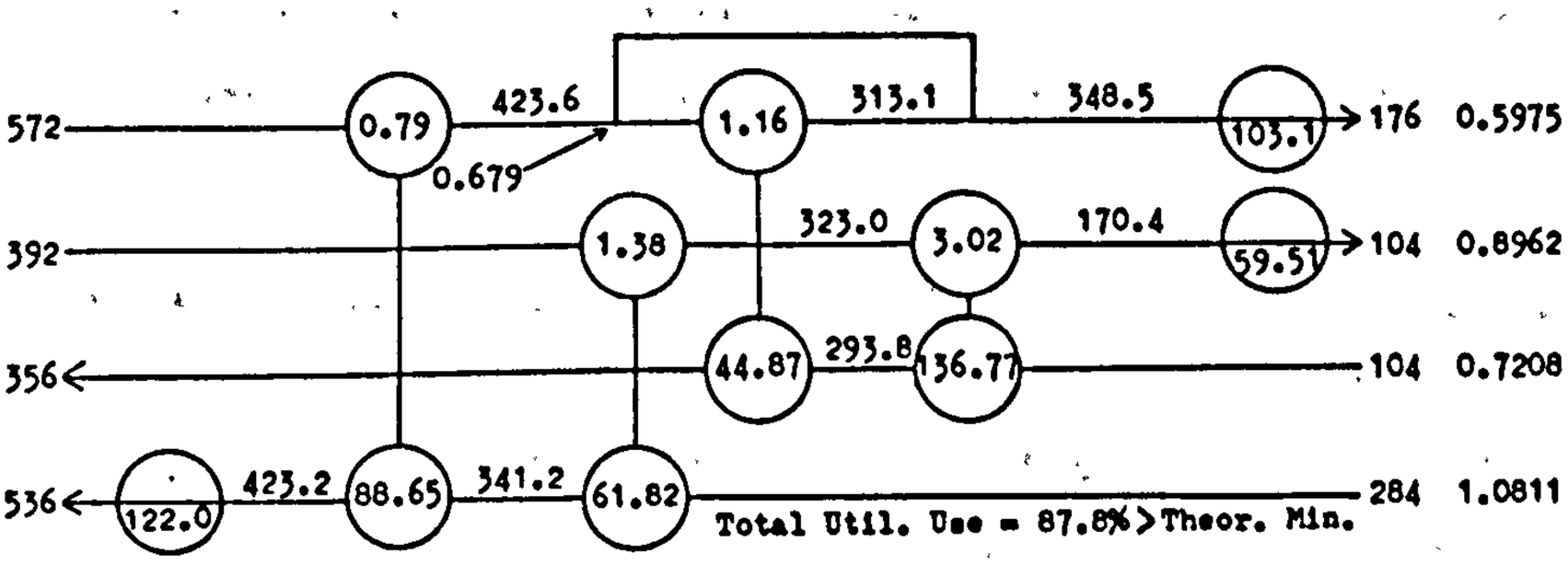
Figure 7.20 The performance of the four alternatives at operating point I.



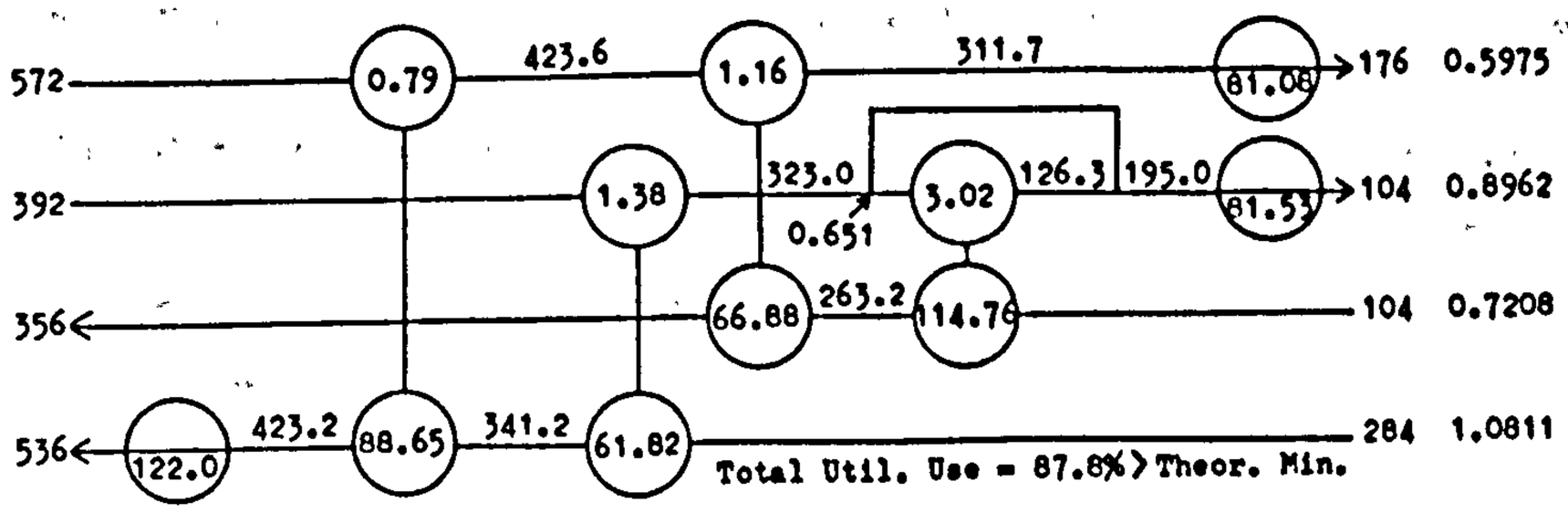
(a)



(b)



(c)



(d)

Figure 7.21 The performance of the four alternatives at operating point I'.

7.7 Conclusions from Design Examples

1) Alternative networks designed for a single set of operating conditions may not perform in the same way if they are required to accommodate a range of operating conditions. Some may require sizing modifications, some may require structural modifications and some may be adequate without any modification.

2) The cheapest network of a number of alternatives designed for nominal operating conditions may not give rise to the cheapest alternative after the networks have been modified to accommodate a range of operating conditions.

3) The best alternative network designed for a range of operating conditions may not give rise to the best alternative after the networks have been modified to accommodate a different range of conditions.

4) It is not generally possible to judge a priori which one of a number of alternative networks designed for nominal operating conditions will prove to be the best alternative after the networks have been modified to accommodate a range of operating conditions.

5) A number of critical operating conditions from within the operating range may be identified such that if a network can operate feasibly for the critical conditions then it can operate over the full operating range. The critical conditions for alternative networks designed for the same operating range may be different.

6) Not all exchangers in a network need necessarily be bypassed.

7) Networks which are identical apart from the choice of which exchangers are bypassed and which are left unbypassed can be regarded as alternative networks in (1), (3), (4) and (5) above.

8) In some cases the correct placement of bypasses can ensure that only regulatory control is required and that the maximum amount of heat possible is recovered for the given network. In these cases optimizing control is at best superfluous and at worst, if the incorrect bypasses are chosen for use as optimizing control, does not permit full heat recovery and prevents operation over the full range of conditions.

9) When forming combined networks it may be possible by the careful placement of bypasses and sizing of exchangers to eliminate the need for switching control, ensure maximum possible heat recovery and produce a network which is cheaper than that required if switching control were used.

10) For all the design examples studied the critical operating points were found to be cornerpoints of the operating range. Usually the critical cornerpoints included points other than the four identified by Morari and co-workers with the critical points being network dependent.

C H A P T E R 8

ADDITIONAL TOPICS

8.1 Non-Independent Inlet Variations

As has been demonstrated in Chapter 4 for a network with zero degrees of freedom, feasible operation over a range of inlet variables is guaranteed if:

$$\max_{\theta^j \varepsilon D} \; g_j(d, \theta^j) \leq 0 \quad \text{for each } j \varepsilon J \tag{8.1}$$

where g_j is the set of network constraints and θ is the set of uncertain variables. For example, if a network has only one variable inlet stream the operating range may be defined by:

$$D = \left\{ \begin{array}{l} 0.0 \leq Tl \leq 1.0 \\ \\ 0.0 \leq F.Cpl \leq 1.0 \end{array} \right\} \tag{8.2}$$

where Tl and F.Cpl are scaled values of the F.Cp value and temperature of the variable stream. Therefore in this case (8.1) becomes:

$$\begin{array}{l} \max \\ \text{w.r.t. Tl and F.Cpl} \\ \text{s.t. } 0.0 \leq Tl \leq 1.0 \\ \qquad 0.0 \leq F.Cpl \leq 1.0 \end{array} \quad g_j(d, Tl, F.Cpl) \leq 0 \text{ for each } j \varepsilon J \tag{8.3}$$

Because the maximization is subject only to fixed bounds on the variables Tl and F.Cpl it can be carried out reliably using, in the present work, NAG routine EØ4JAF. The assumption made is that Tl and F.Cpl can vary independently over the given range (8.2) with any combination being possible. The operating range is as shown in Figure 8.1 (a) with any combination of Tl and F.Cpl in the shaded region assumed to be possible.

At this stage in specifying an appropriate operating range it is possible to eliminate highly unlikely operating conditions. For example, it may be estimated that F.Cpl will be within the range 0.0 to 1.0 for 95% of the time and Tl in the range 0.0 to 1.0 for 100% of the

time. The resulting network, designed to operate over the specified range will operate feasibly for at least 95% of the time, which may be an adequate reliability.

In practice the variables may not be independent. In this case as Tl and F.Cpl are properties of the same stream they may be linked. For example, if the stream is the output from a reactor it may be expected that low flowrates, and hence F.Cpl values, will occur with high temperatures, Tl, and vice versa. The operating range may be as shown in Figure 8.1 (b) with the possible combinations of F.Cpl and Tl in the shaded region. For example, for an F.Cpl value of 1.0, Tl is expected to be between 0.0 and 0.5.

A cheaper network design may be produced by designing for this reduced operating range since the network need not accommodate a range of F.Cpl, Tl combinations which will not arise but which would be taken account of if the full operating range were used.

The mathematical description of the reduced operating range is:

$$0.0 \leq F.Cpl \leq 1.0 \qquad \text{or} \qquad 0.0 \leq F.Cpl \leq 1.0$$
$$0.0 \leq Tl \leq 1.0 \qquad\qquad 0.0 \leq Tl \leq 1.0 \qquad (8.4)$$
$$-Tl + 0.9 \leq F.Cpl \leq -1.6Tl + 1.8 \qquad 0.9 - F.Cpl - Tl \leq 0.0$$
$$F.Cpl + 1.6Tl - 1.8 \leq 0.0$$

Adapting the procedure (8.3), which was for the full operating range, to this reduced range yields:

$$\begin{aligned} &\max \\ &\text{w.r.t. Tl and F.Cpl} \\ &\text{s.t. } 0.0 \leq Tl \leq 1.0 \\ &\qquad 0.0 \leq F.Cpl \leq 1.0 \\ &\qquad 0.9 - F.Cpl - Tl \leq 0.0 \\ &\qquad Fl + 1.6Tl - 1.8 \leq 0.0 \end{aligned} \qquad g_j(d, Tl, F.Cpl) \leq 0 \text{ for each } j \varepsilon J \qquad (8.5)$$

As this maximization problem is subject not only to fixed bounds on the variables but also to inequality constraints, the maximization would be likely to be more time-consuming and less reliable than that required

for (8.3). However, the resulting network could be cheaper and better suited to its required task.

In practice, if more than two variables were involved, specifying the interactions between them might be difficult. Specifying the interaction between a process stream's F.Cp value and temperature, as above, might be possible but the most likely source of determinable interactions is overall mass balance considerations. In a heat exchanger network all process streams appear to be completely separate but when viewed in the context of the overall process they will not be. A decrease in the flowrate of a feedstock to the process will propagate through the system and lead to a decrease in the flowrate of a number of intermediate streams which may be involved in the heat exchanger network. If these effects can be quantified the required operating range for the heat exchanger network can be reduced. Any shape of operating range can be specified although the more complex the relationships the more difficult the maximization problem is likely to become. In some cases it may be possible to directly relate the F.Cp value of one process stream to the F.Cp value of other streams; for example in any device which mixes two streams the outlet flowrate will be equal to the sum of the inlet flowrates under steady-state conditions. As the procedures developed in the present work are only concerned with steady-state conditions, for every set of inlet flowrates to the process the outlet flowrate from the mixer may be calculated exactly. If this stream appears in the heat exchanger network its F.Cp value need not be treated as an uncerain variable as it will be able to be calculated directly from other F.Cp values.

To demonstrate the advantages of reducing an operating range an example problem is studied.

8.1.1 Example problem

In Chapter 7 an unpinched problem based on literature problem 5SP1 was studied. For an operating range formed by allowing all F.Cp values to vary independently from 0.95 to 1.0 times the nominal values, the

cheapest network of the five alternatives considered is given in Figure 8.2

During the design procedure only sizing infeasibilities were detected and therefore any reduced feasible range will also exhibit only sizing infeasibilities.

Assume that because of overall mass balance considerations the F.Cp values are linked as shown in Figure 8.3 which shows the shaded range containing possible combinations of the F.Cp values. AN is the nominal F.Cp value of stream N. It can be seen that the F.Cp values for streams 1 to 5 cover the range 0.95 to 1.0 times their respective nominal values, as in the case considered in Chapter 7 where they were allowed to vary independently, but in this case the F.Cp values are non-independent. Low values of F.Cp1 occur with low values of the other F.Cp values and similarly for high values. e.g. If F.Cp1 = 2.9925 (0.95 times nominal) then F.Cp2, F.Cp3, F.Cp4 and F.Cp5 will all take values in the range 0.95 to 0.96 times their respective nominal values (AN).

Thus it can be seen that in addition to the simple bounds on the F.Cp values:

$$\begin{aligned} 2.9925 &< F.Cp1 < 3.15 \\ 2.394 &< F.Cp2 < 2.52 \\ 2.052 &< F.Cp3 < 2.16 \\ 2.3275 &< F.Cp4 < 2.45 \\ 2.3465 &< F.Cp5 < 2.47 \end{aligned} \qquad (8.6)$$

there are additional constraints of the form:

$$\frac{0.04AN}{0.1575} F.Cp1 + 0.19AN < F.CpN < \frac{0.04AN}{0.1575} . F.Cp1 + 0.2AN \; ; \; N=2, 5 \qquad (8.7)$$

Substituting the appropriate values:

$$
\begin{aligned}
&0.64\ F.Cp1 + 0.4778 < F.Cp2 < 0.64\ F.Cp1 + 0.504 \\
&0.549\ F.Cp1 + 0.410 < F.Cp3 < 0.549\ F.Cp1 + 0.432 \\
&0.622\ F.Cp1 + 0.466 < F.Cp4 < 0.622\ F.Cp1 + 0.49 \qquad (8.8) \\
&0.627\ F.Cp1 + 0.469 < F.Cp5 < 0.627\ F.Cp1 + 0.494
\end{aligned}
$$

A specially modified form of the computer program described in Chapter 5 using NAG routine EØ4UAF rather than EØ4JAF was used to maximize each required exchanger split fraction with respect to the uncertain variables F.Cp1 to F.Cp5 subject to the two sets of constraints (8.6) and (8.8). EØ4JAF could not be used because the constraints on the variables are not only simple bounds. Two critical points, I and II, were found in the operating range for which the required exchanger split fractions, and hence required exchanger areas, were maximized. These two operating points are shown in Figure 8.4 with the required exchanger U.A values for these points indicated. The U.A values required to make the network operate feasibly over the reduced operating range are found by taking the maximum of the required U.A values for points I and II. Using the same assumptions as in Chapter 7, the exchanger areas and costs may be calculated and compared for the nominal design, the network designed to operate over the full range (Figure 8.2) and the network designed to operate over the reduced range. These results are given in Table 8.1 and demonstrate that in this case there is a large financial incentive in ensuring that the network is designed to handle the reduced operating range, if this is applicable, rather than making the design task easier by simply designing for the full operating range.

Morari and co-workers do not consider the problems of non-independent variables in their work although in this case critical points I and II are found to correspond to cornerpoints of the reduced range, see Figure 8.5.

Exchanger Areas (ft^2)

Operating range designed for \ Exchanger	W	1	2	3	4
Nominal conditions only	136	336	197	801	168
Full range	152	343	301	916	219
Reduced range	138	330	211	821	177

Cost ($)

Nominal conditions design = 53 260
Full operating range design = 59 330 (11.4% > Nominal Design)
Reduced operating range design = 54 200 (1.8% > Nominal Design)

Table 8.1 Cost of networks under consideration

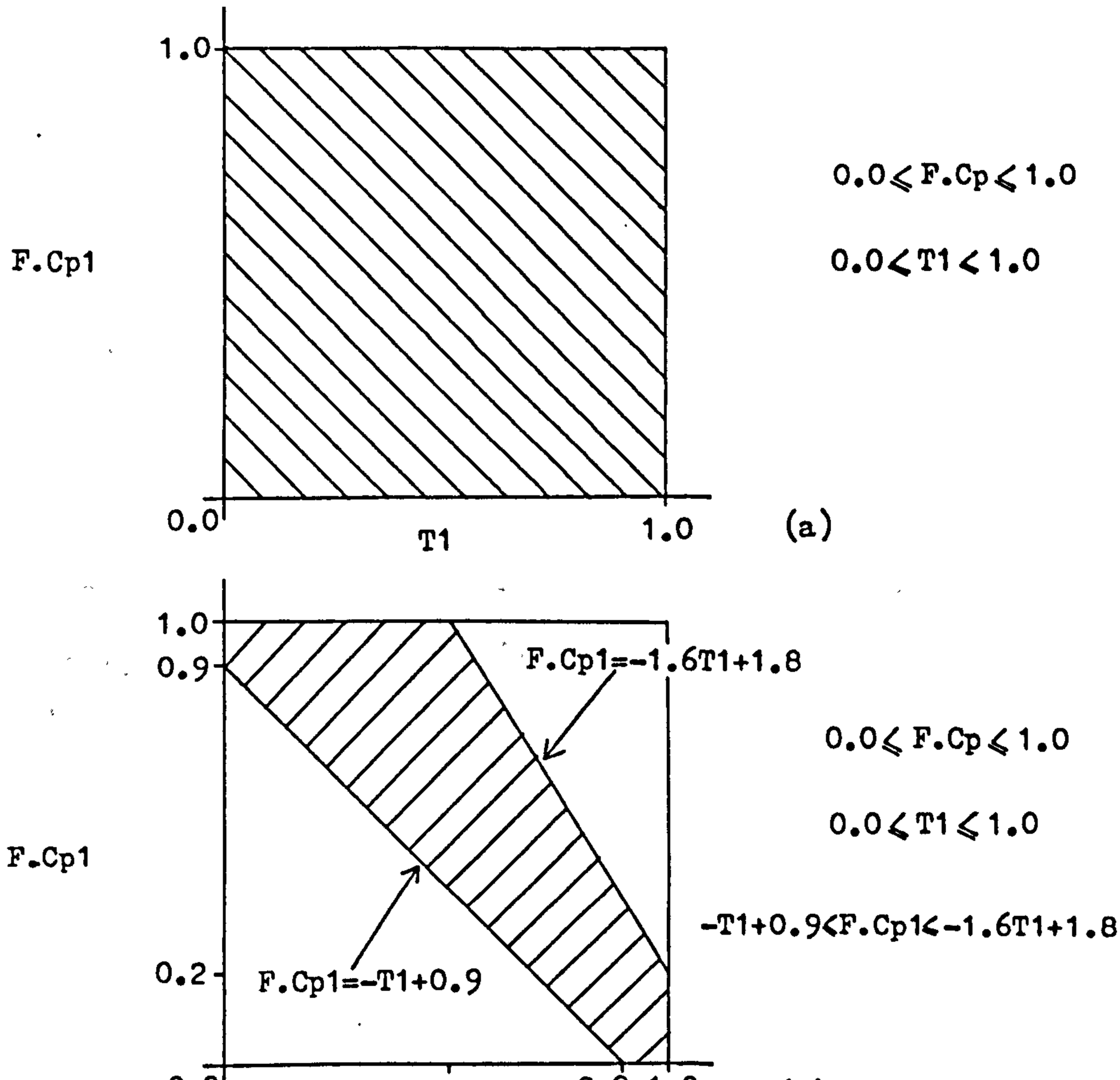


Figure 8.1 Operating ranges for two uncertain inlet variables.

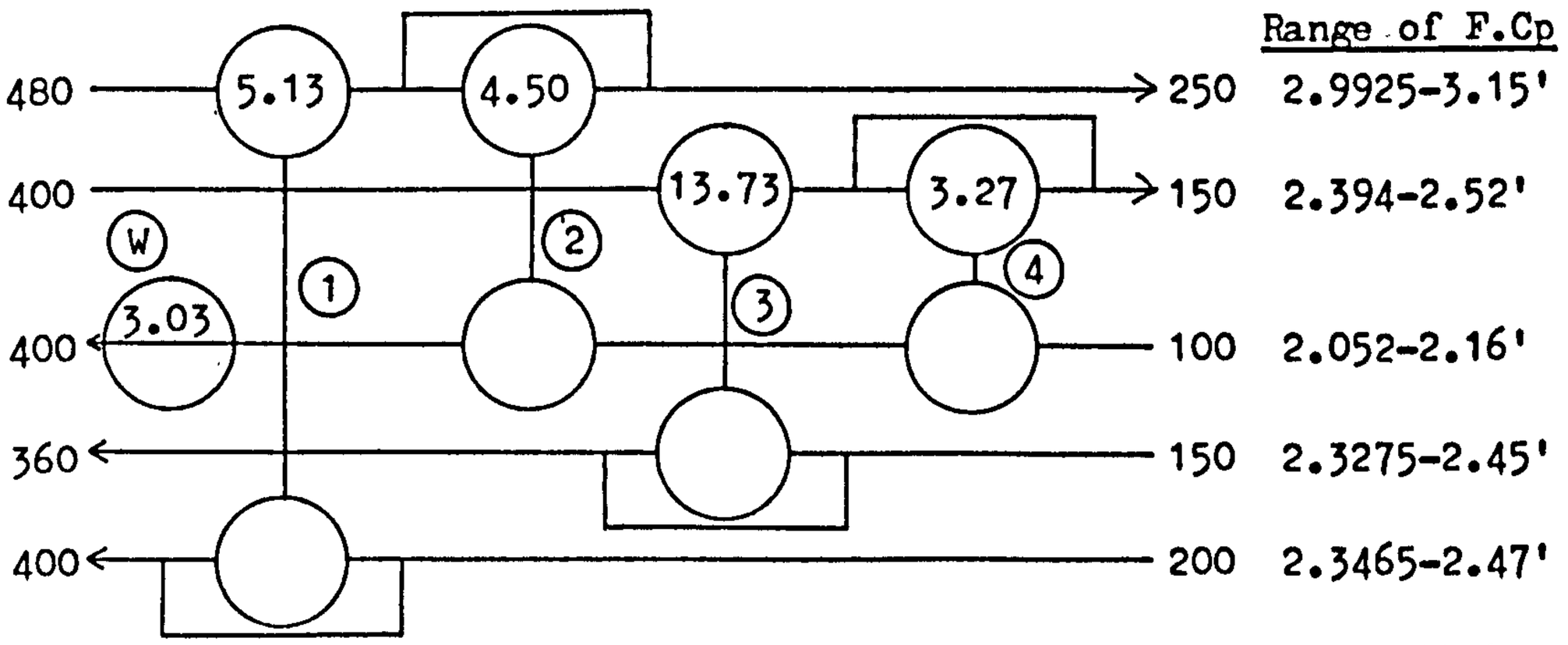


Figure 8.2 The cheapest network of those considered in Chapter 7 for problem 5SP1 able to handle independently varying F.Cp values from 0.95 to 1.0 times nominal.

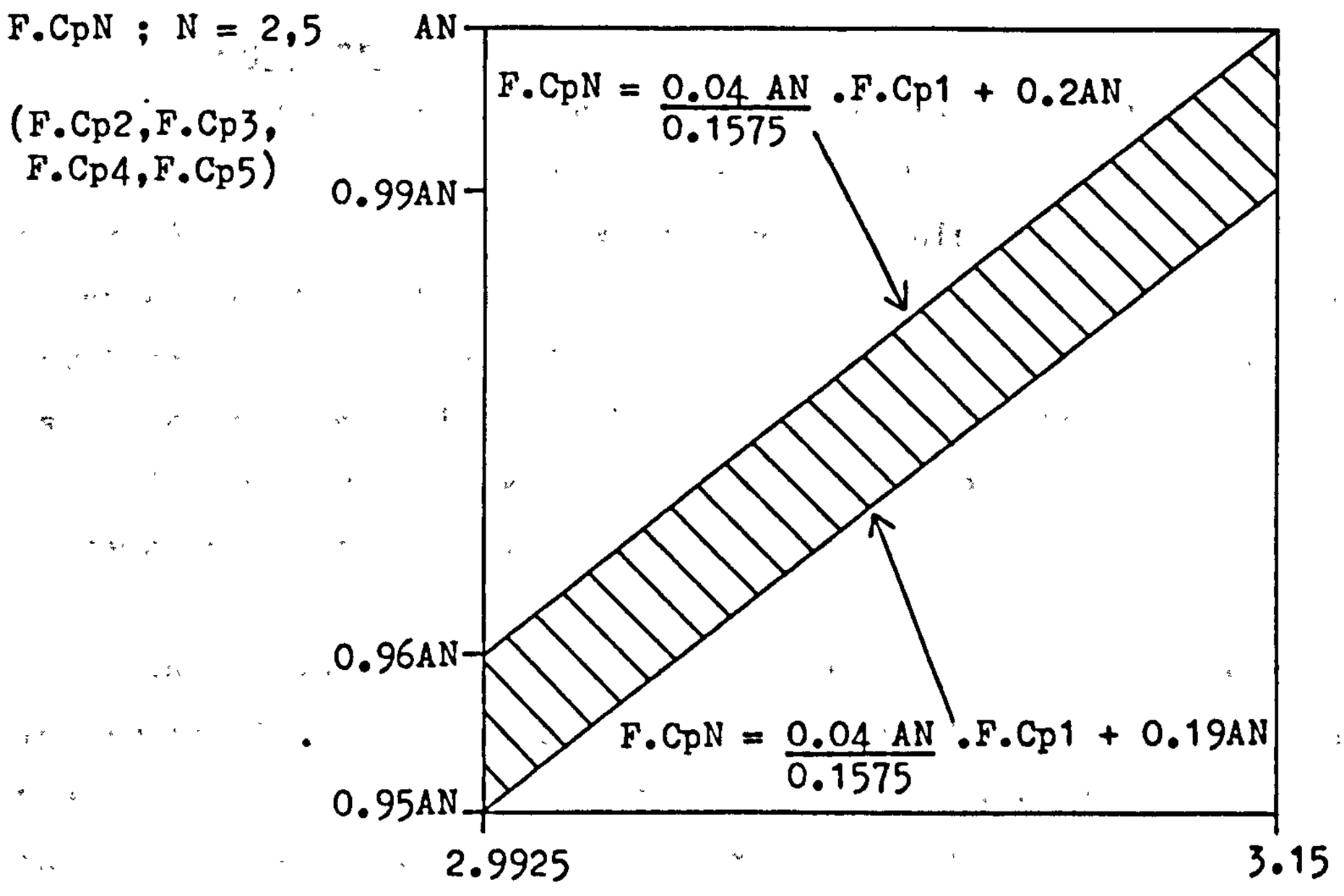


Figure 8.3 The reduced operating range.

8.2 Networks which have Design Flexibility

As noted in Chapter 6 networks which have more than the minimum number of units or include stream-splits have design flexibility as described by Linnhoff and Flower (1978). This effectively means that the designer is free to choose exchanger areas according to some criteria in addition to the requirement that the network should satisfy the required duty. The design problem for the case in which a network must be able to operate over a given operating range may be demonstrated by considering an example problem.

8.2.1 Example problem

The cheapest network not using stream-splits designed for problem 5SP1 with F.Cp variations of 0.95 to 1.0 times nominal is given in Figure 8.2. A possible network based on this case with a stream-split is given in Figure 8.6. The required U.A values for the exchangers not involved in the stream-split will be the same as for the unsplit case. As the stream-split is regarded as a splitter in the degrees of freedom analysis only three bypasses need be assigned to ensure that the degrees of freedom of the network are zero.

The stream-split introduces design flexibility, meaning that for any particular single set of operating conditions different values of the split fraction R will produce different required values for U.A1 and U.A2. This is useful for flexible design as since a range of U.A values may produce feasible operation for single sets of conditions, values may be chosen which can also handle other operating points.

As an example of this design flexibility, Figure 8.7 gives three different nominal case designs with different required values for U.A1 and U.A2. The designs are based on the following criteria.

Figure 8.7 (a) - Minimum Total Area

U.A1 and U.A2 are chosen to minimize U.A1 + U.A2 which for an identical U value for each exchanger corresponds to the minimum total area of the two exchangers.

Figure 8.7 (b) - Minimum Cost

U.A1 and U.A2 are chosen to minimize $(U.A1)^{0.6} + (U.A2)^{0.6}$ which for identical U values corresponds to the minimum cost design based on the usual literature costing formula given by Linnhoff (1979).

Figure 8.7 (c) - Isothermal Mixing

U.A1 and U.A2 are chosen so that the temperatures of the streams entering the mixer are equal.

Once a value of U.A1 is set, the required value of U.A2 can be calculated and the minimum cost and area designs are found using a simple search procedure.

Economically the minimum cost solution is preferable. Taking the minimum cost design, Figure 8.7 (b), and using the computerized constraint function maximization procedure a most infeasible operating point, I, can be found. The performance of the network designed for nominal conditions at this set of operating conditions is shown in Figure 8.8. In this case exchanger 1 is undersized, this being indicated by the required exchanger split fraction of the bypass being greater than 1.0. It would be possible to oversize this exchanger to make the network feasible for this set of conditions but the exchanger areas would not be economically optimum. The economically optimum design for operating point I is given in Figure 8.9. This network is feasible for nominal operating conditions but it is not feasible for another operating point II as is shown in Figure 8.10. Figure 8.11 shows the economically optimum design for operating point II and Figure 8.12 shows the infeasible operation of this network at operating point I.

To summarize, starting from nominal operating conditions two critical operating points I and II have been found. For all three sets of operating conditions values of U.A1 and U.A2 giving the lowest cost network have been found. These values are given in Table 8.2. None of the three combinations of U.A1 and U.A2 gives feasible operation for all three operating points.

A method is needed to find the cheapest U.A1, U.A2 combination which is feasible for all three sets of operating conditions. The resulting network should then be fully feasible for the entire operating range. (In fact, ensuring feasibility at critical points I and II only may be sufficient).

8.2.1.1 Formulation of problem

For each set of operating conditions, the variables which are independent of the chosen values of U.A1 and U.A2 can be calculated. These are indicated in Figure 8.13 with the variables dependent on U.A1 indicated.

The only variables common to all three cases are U.A1 and U.A2. The vector x of all variables dependent on U.A1 and U.A2 is:

$$x = (T11, T12, T13, R11, R12, U.A1, T21, T22, T23, R21, R22, U.A2, T31, T32, T33, R31, R32)$$

For each set of conditions certain equality constraints (Q) link the unknown variables. These are based on heat balances on the hot and cold streams of exchangers and overall heat balances.

They are:

Nominal Conditions

$$Q(1) = (480.0-T12) \times (1.0-R11) \times 3.15 - 494.0 = 0.0 \quad (8.9)$$

$$Q(2) = (T13-200.0) \times R12 \times 2.47 - 494.0 = 0.0 \quad (8.10)$$

$$Q(3) = 494.0 - U.A1 \times ((480.0-T13) - (T12-200.0))/ \ln(480.0-T13)/(T12-200.0)) = 0.0 \quad (8.11)$$

$$Q(4) = (480.0-T11) \times R11 \times 3.15 - 230.5 = 0.0 \quad (8.12)$$

$$Q(5) = 230.5 - U.A2 \times (219.82-T11+153.472)/\ln(219.82/(T11-153.472)) = 0.0 \quad (8.13)$$

Critical Point I

$$Q(6) = (480.0-T22) \times (1.0-R21) \times 3.15 - 469.3 = 0.0 \quad (8.14)$$

$$Q(7) = (T23-200.0) \times R22 \times 2.3465 - 469.3 = 0.0 \quad (8.15)$$

$$Q(8) = 469.3 - U.A1 \times ((480.0-T23) - (T22-200.0))/ \ln((480.0-T23)/(T22-200.0)) = 0.0 \quad (8.16)$$

$$Q(9) = (480.0-T21) \times R21 \times 3.15 - 255.2 = 0.0 \quad (8.17)$$

$$Q(10) = 255.2 - U.A2 \times (186.81-T21+168.82)/\ln(186.81/(T21-168.82)) = 0.0 \quad (8.18)$$

Critical Point II

$$Q(11) = (480.0-T32) \times (1.0-R31) \times 3.15 - 494.0 = 0.0 \quad (8.19)$$

$$Q(12) = (T33-200.0) \times R32 \times 2.47 - 494.0 = 0.0 \quad (8.20)$$

$$Q(13) = 494.0 - U.A1 \times ((480.0-T33) - (T32-200.0))/ \ln((480.0-T33)/(T32-200.0)) = 0.0 \quad (8.21)$$

$$Q(14) = (480.0-T31) \times R31 \times 3.15 - 230.5 = 0.0 \quad (8.22)$$

$$Q(15) = 230.5 - U.A2 \times (198.848-T31+168.823)/\ln(198.848/(T21-168.823)) = 0.0 \quad (8.23)$$

Hence for any choice of U.A1 and U.A2 all of these equality constraints must be satisfied:

$$Q = 0 \quad (8.24)$$

To ensure feasible operation at all three sets of conditions bounds can be put on all of the unknown variables to ensure that the network constraints are satisfied. For example, referring to Figure 8.13, T11 must be less than 480.0 to ensure a positive heat load in the exchanger. Also, T11 must be greater than 153.47 to ensure a positive temperature approach. To ensure that T11 > 153.47, R11 must be greater than 0.23.

By similar arguments:

Nominal Conditions

$$153.47 < T11 < 480.0 \quad (8.25)$$

$$200.0 < T12 < 480.0 \quad (8.26)$$

$$200.0 < T13 < 480.0 \quad (8.27)$$

$$0.23 < R11 < 0.43 \quad (8.28)$$

$$0.0 < R12 < 1.0 \quad (8.29)$$

Critical Point I

$$168.82 < T21 < 480.0 \quad (8.30)$$

$$200.0 < T22 < 480.0 \quad (8.31)$$

$$200.0 < T23 < 480.0 \quad (8.32)$$

$$0.27 < R21 < 0.46 \quad (8.33)$$

$$0.0 < R22 < 1.0 \quad (8.34)$$

Critical Point II

$$168.82 < T31 < 480.0 \quad (8.35)$$

$$200.0 < T32 < 480.0 \quad (8.36)$$

$$200.0 < T33 < 480.0 \quad (8.37)$$

$$0.24 < R31 < 0.43 \quad (8.38)$$

$$0.0 < R32 < 1.0 \quad (8.39)$$

Approximate bounds on the optimum values of U.A1 and U.A2 may be deduced from Table 8.2.

$$6.7 < U.A1 < 7.3 \quad (8.40)$$

$$1.6 < U.A2 < 2.3 \quad (8.41)$$

Therefore, to ensure feasible operation at all three sets of operating conditions, the unknown variables must be within the bounds set:

$$x^L < x < x^U \quad (8.42)$$

The problem of finding a minimum cost combination of U.A1 and U.A2 while maintaining feasible operation becomes:

$$\begin{aligned} &\min_{\text{w.r.t. } x} && ((U.A1)^{0.6} + (U.A2)^{0.6}) \\ &\text{s.t.} && Q = 0 \\ & && x^L < x < x^U \end{aligned} \quad (8.43)$$

This is a standard minimization problem and was solved in this case using NAG routine EØ4UAF, the solution being found in less than 5 c.p. sec. (CDC 6500).

The solution found was:

$$U.A1 = 6.971 \quad , \quad U.A2 = 2.157$$

The performance of this optimum cost network at the three sets of operating conditions considered is shown in Figure 8.14. Further investigation using the constraint function maximization program demonstrates that this network can operate over the full operating range. A comparison of the cost of this network with the cost of the best network not using a stream-split is given in Table 8.3. Even though the network with a stream-split is more expensive when designed for nominal operating conditions it requires less oversizing than the network without a stream-split to make it flexible over the given operating range and therefore produces a cheaper design.

8.2.2 General design method for a network with design flexibility

This example demonstrates a design method which can be applied to all networks with design flexibility:

1) For a given network designed for one point in the operating range use a constraint function maximization procedure to identify the critical operating points, that is the operating points producing the maximum constraint function values, some of which will be infeasible values if the original design is inadequate. In general there will be more than one critical point to consider, as in the above example.

2) Set up the equality and inequality constraints, in terms of the unknown U.A values and dependent variables, which must be satisfied to ensure feasible operation at each critical point. For pinched problems a utility use constraint can be used to ensure adequate heat recovery for each critical point.

3) Using an optimization procedure calculate the economically optimum exchanger areas which ensure feasibility for all the critical points.

4) Using the network constraint function maximization procedure check the resultant network for adequate operation over the full operating range. If a new critical point is found producing infeasible operation add it to the set of critical points and return to Step (2).

Step (4) is necessary because it is not clear whether network contraint functions are monotonic with respect to the exchanger areas chosen; in terms of equation (2.3) in Chapter 2, whether g' are monotonic with respect to d. The critical points found by considering a network sized for one set of operating conditions may not be the critical points when the areas of the exchangers are changed. However, it would seem likely that, in general, just as the vast majority of the examples considered in the present work have had constraint functions which are monotonic with respect to the uncertain (θ) and control (z) variables so most problems will also have constraint functions which are monotonic with respect to the design parameters (d).

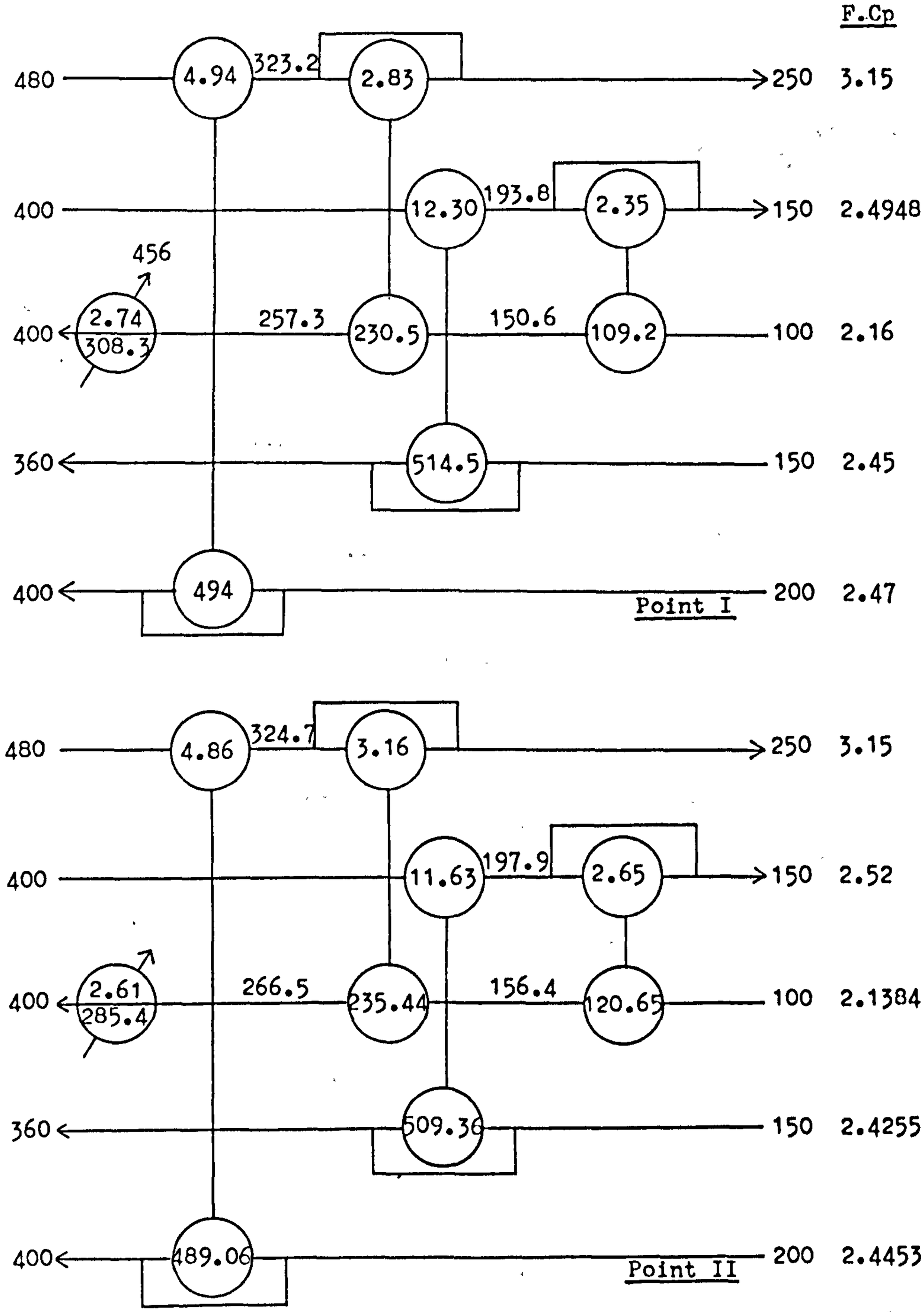
	Inlet F.Cp value of streams 1 to 5	U.A1	U.A2
Nominal Conditions	3.15, 2.52, 2.16, 2.45, 2.47	7.035	1.782
Critical Point I	3.15, 2.52, 2.052, 2.3275, 2.3465	6.789	2.257
Critical Point II	3.15, 2.52, 2.052, 2.3275, 2.47	7.211	1.971

Table 8.2 The optimum U.A1 and U.A2 values for three sets of operating conditions.

Cost ($)

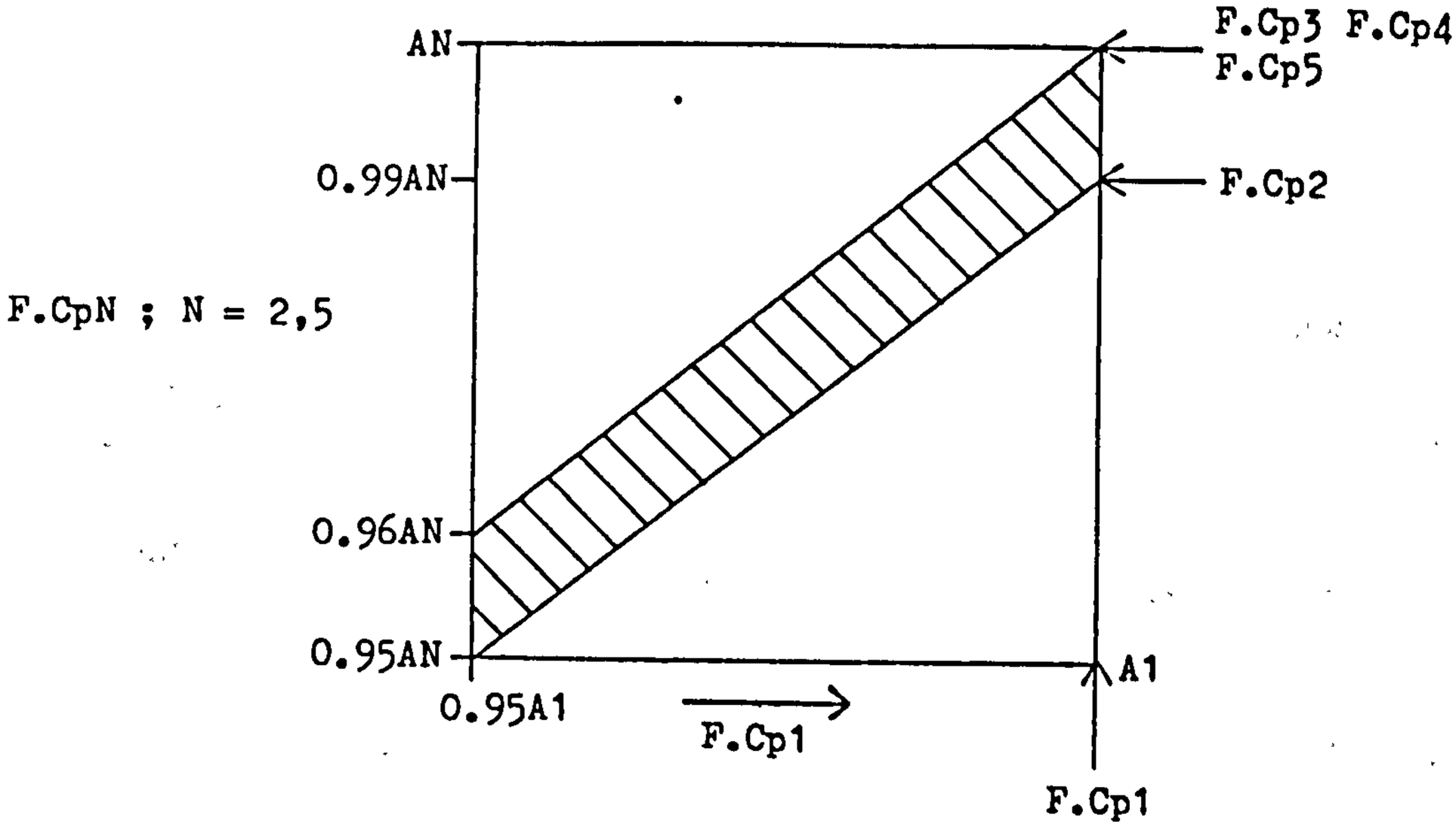
	Design for nominal operating conditions only	Design for an operating range of F.Cp values from 0.95 to 1.0 times nominal
The best network without a stream-split	53 260	59 330
The network with a stream-split	53 770	57 810

Table 8.3 A comparison of the costs of a network using a stream-split with the best network not using a stream-split

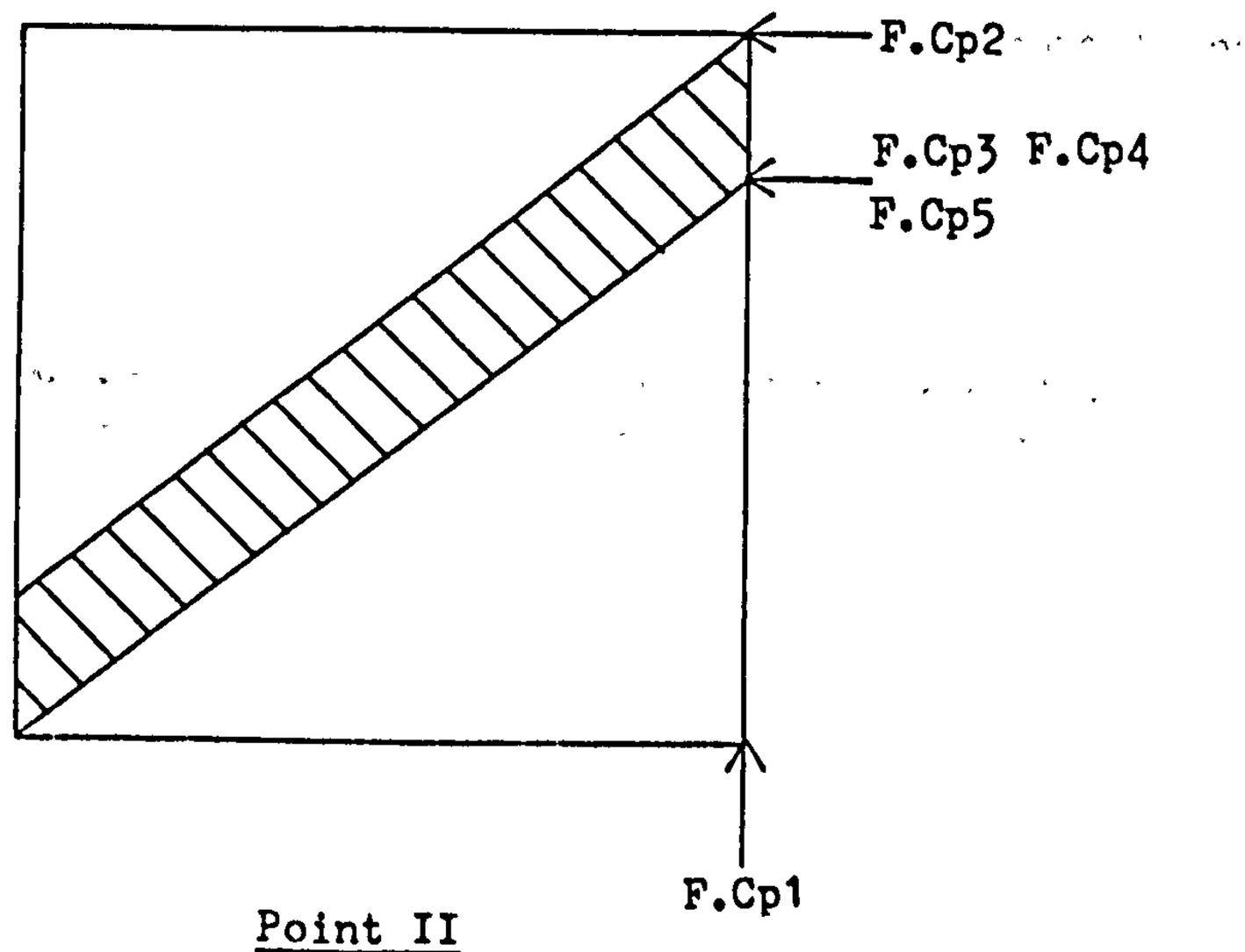


Maximum required U.A values : W = 2.74 , Ex.1 = 4.94 , Ex.2 = 3.16 ,
Ex.3 = 12.30 , Ex.4 = 2.65

Figure 8.4 The minimum required exchanger areas at operating points I and II.



Point I



Point II

Figure 8.5 The location of critical points I and II in the operating range as represented in Figure 8.3.

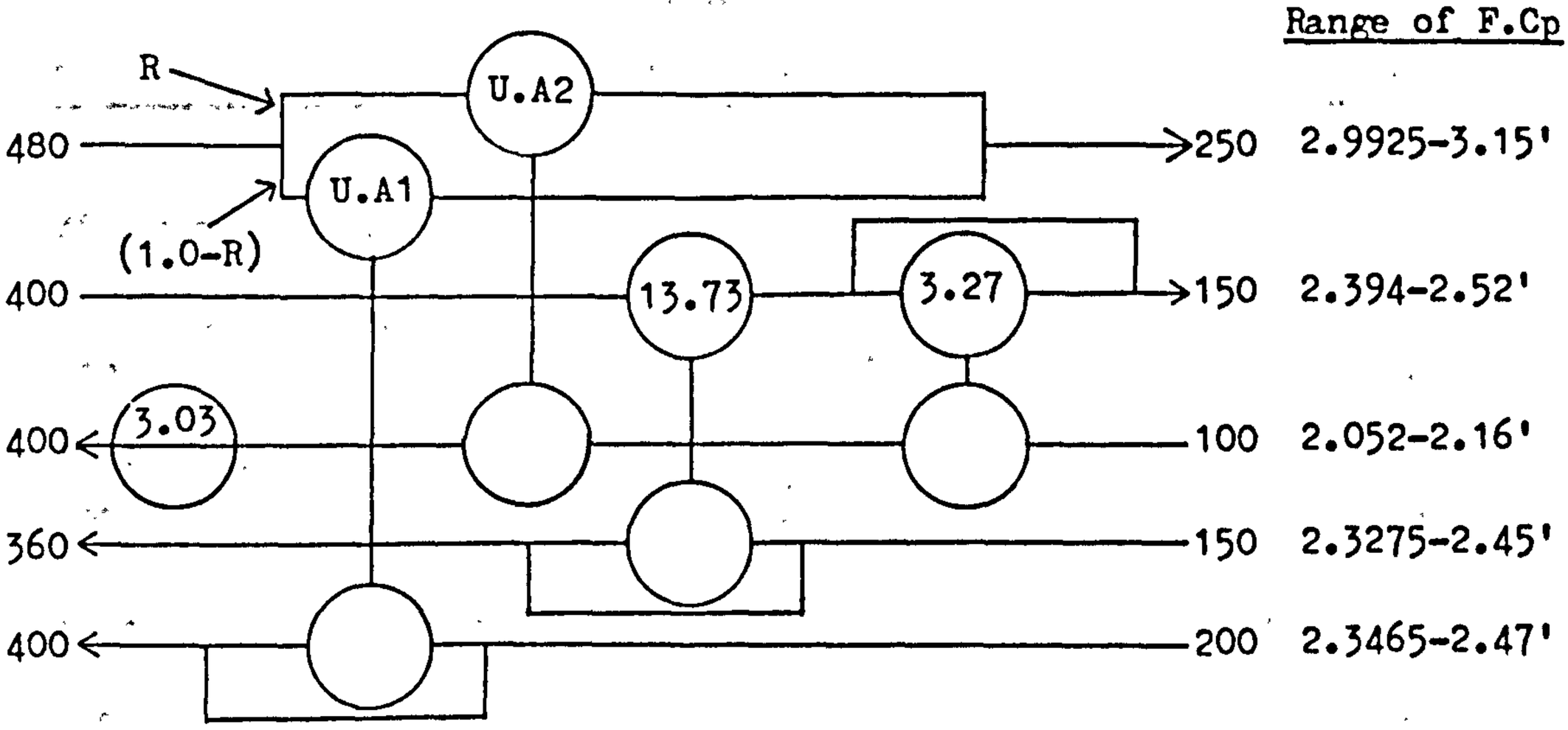


Figure 8.6 A possible network including a stream-split for problem 5SP1 with a range of F.Cp values from 0.95 to 1.0 times nominal.

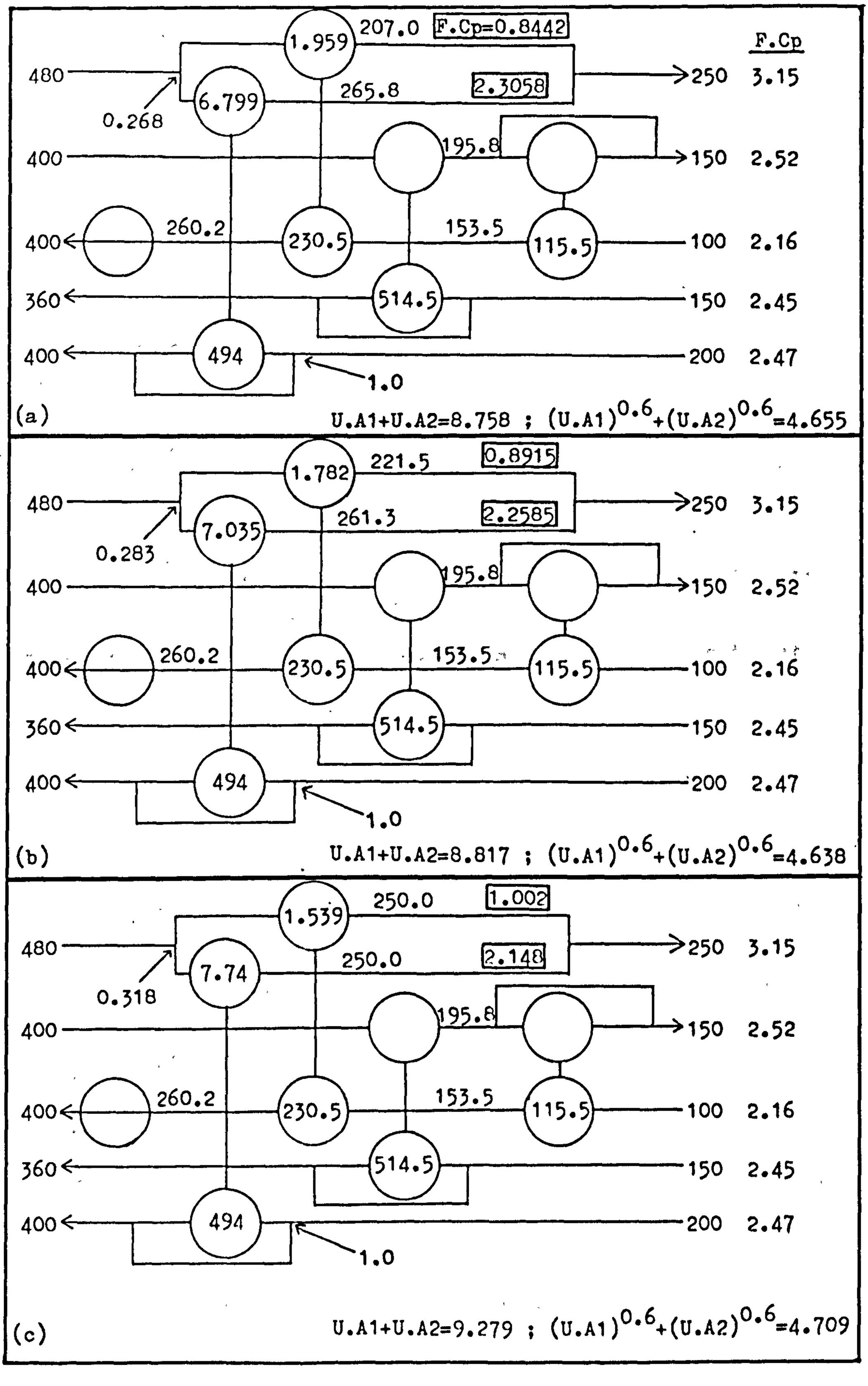


Figure 8.7 Three designs for nominal conditions.

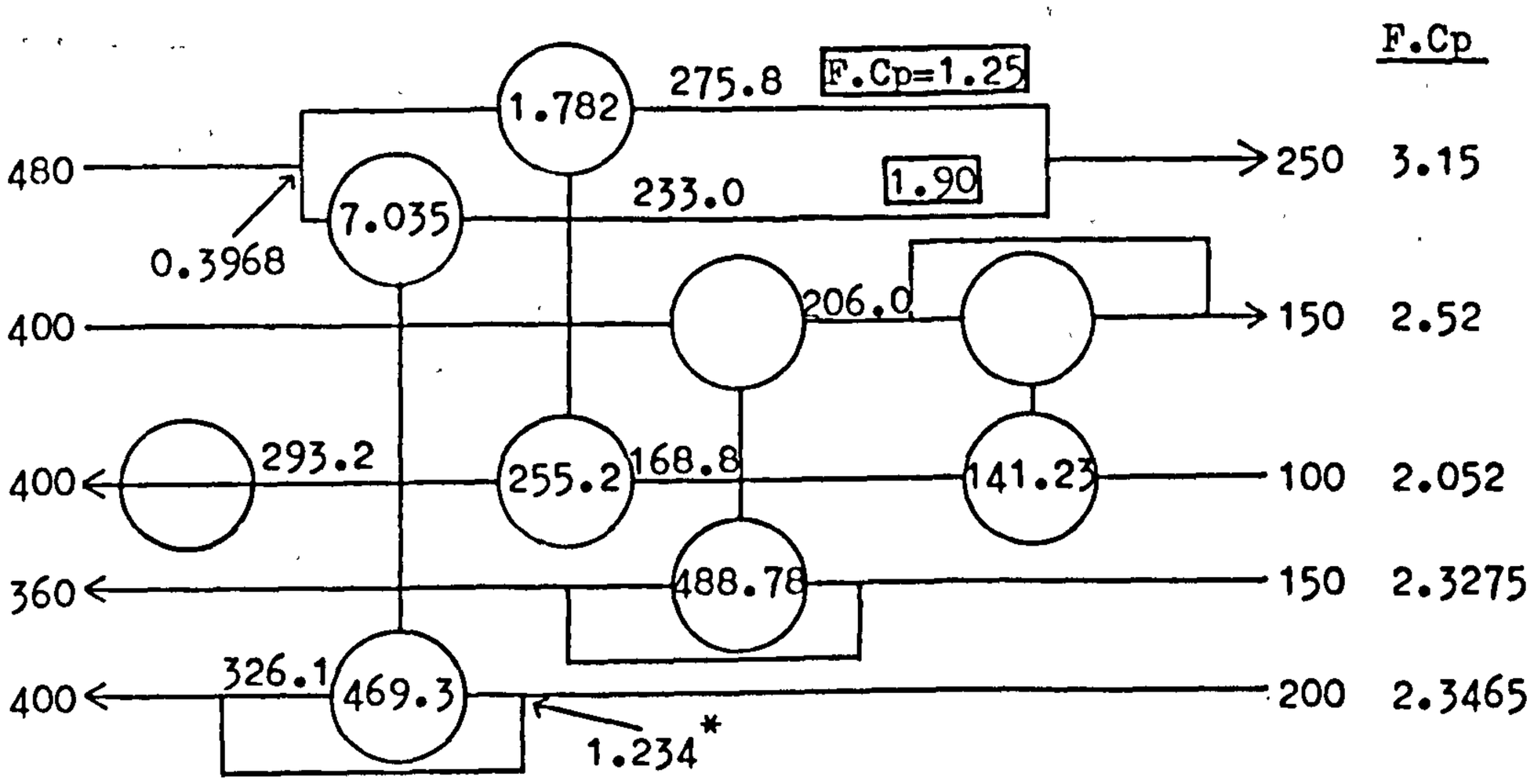


Figure 8.8 The performance of the optimum nominal case design at operating point I.

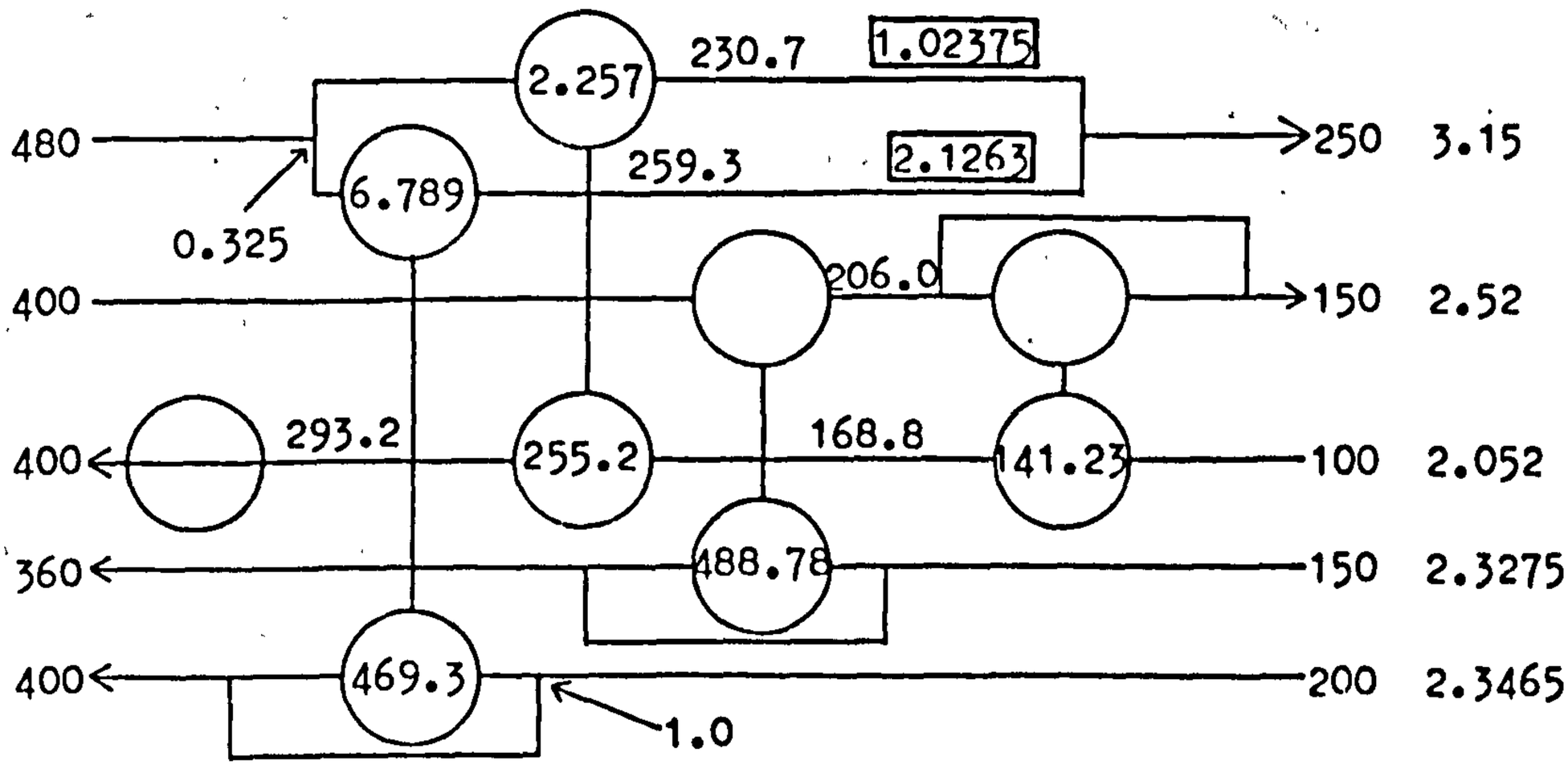


Figure 8.9 The optimum design for operating point I.

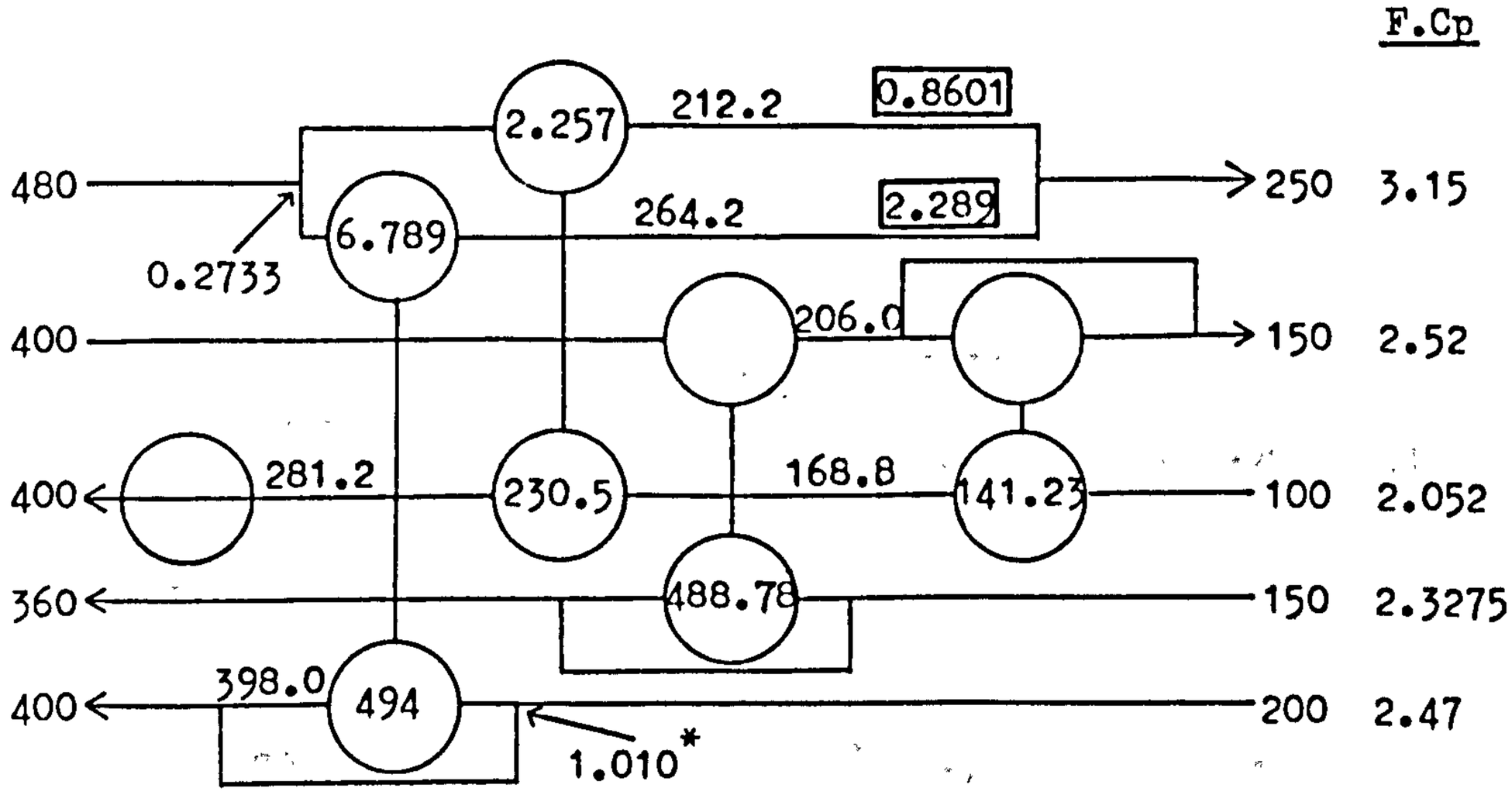


Figure 8.10 The performance of the optimum network designed for operating point I at operating point II.

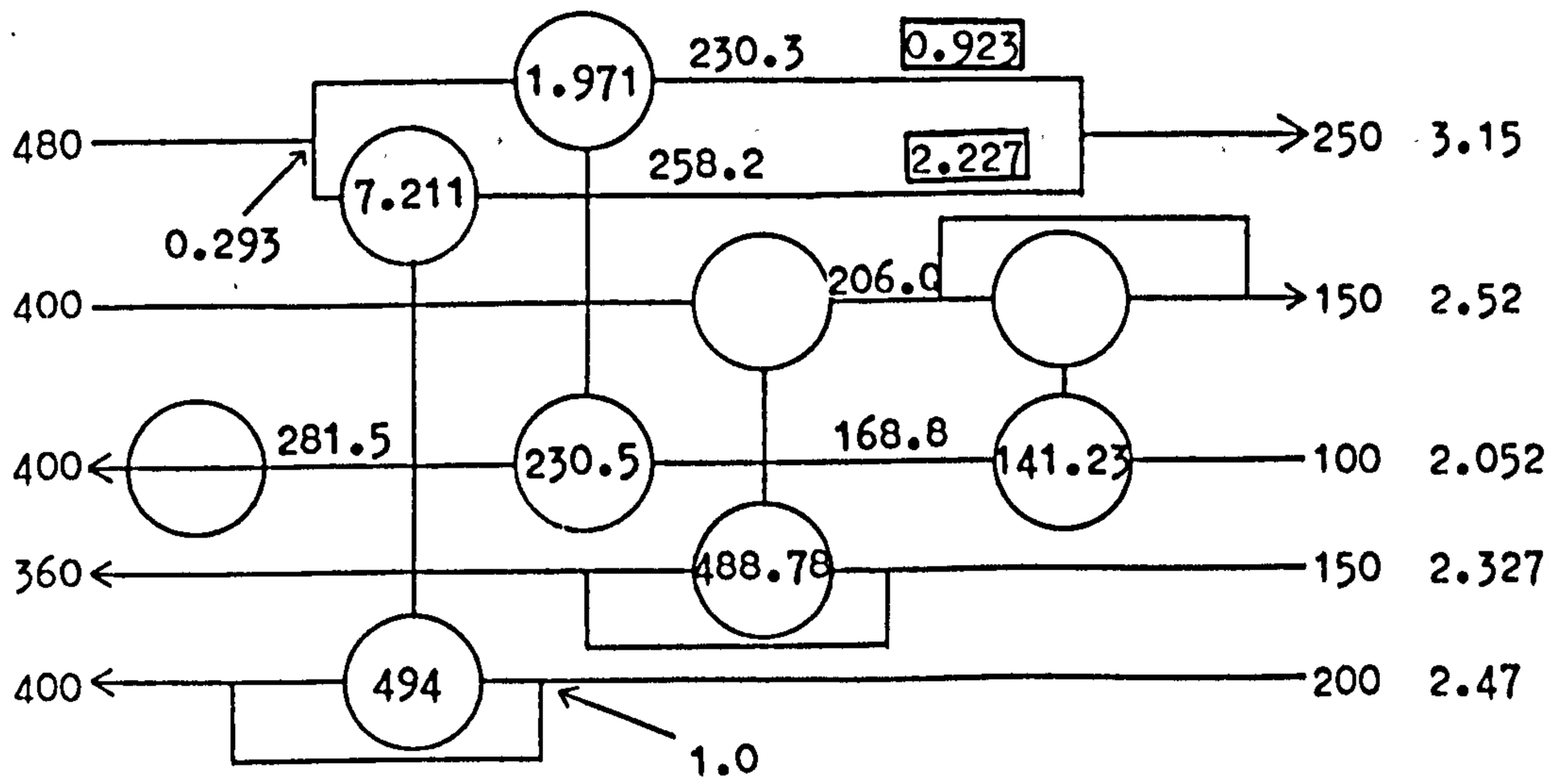


Figure 8.11 The optimum design for operating point II.

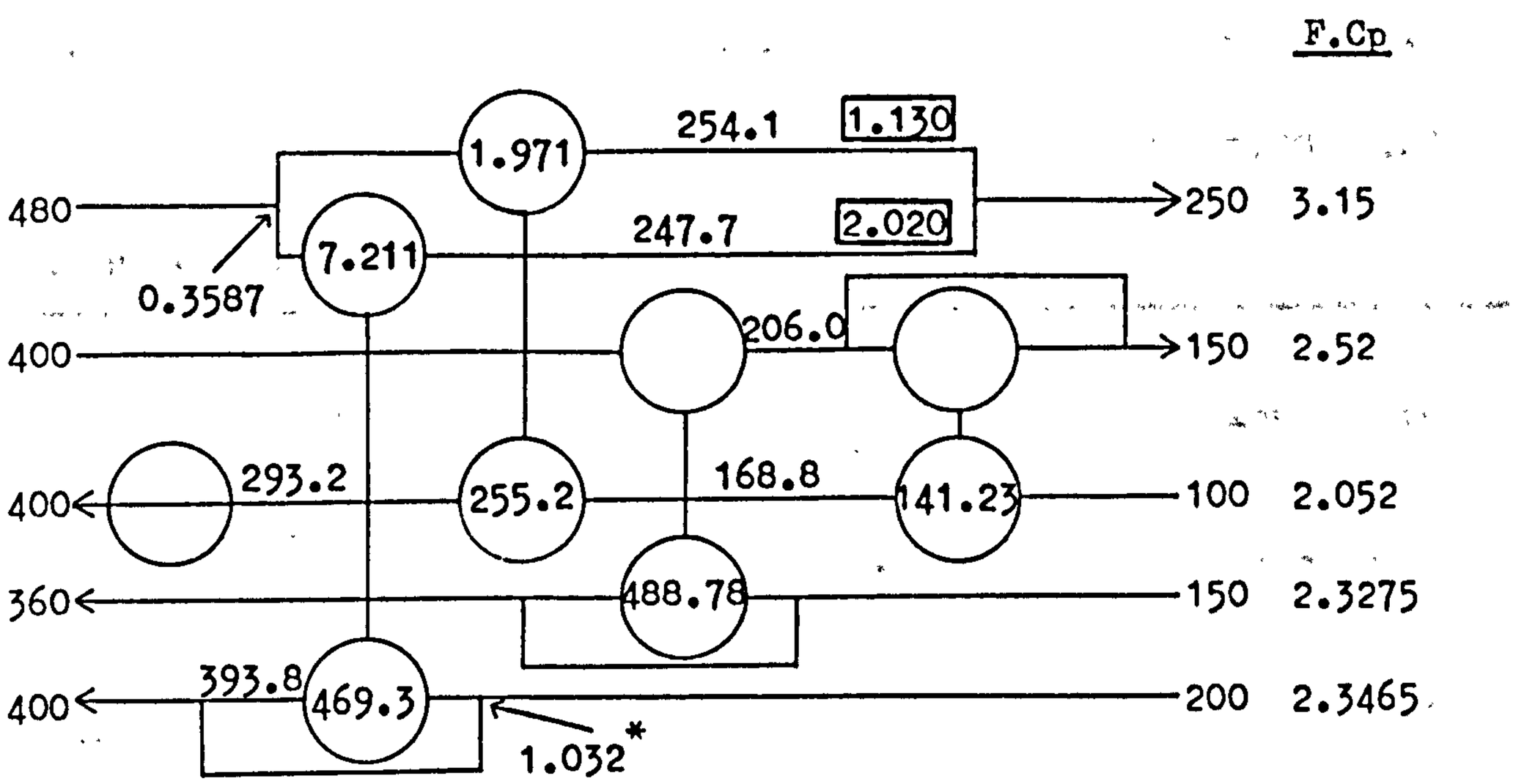


Figure 8.12 The performance of the optimum network designed for operating point II at operating point I.

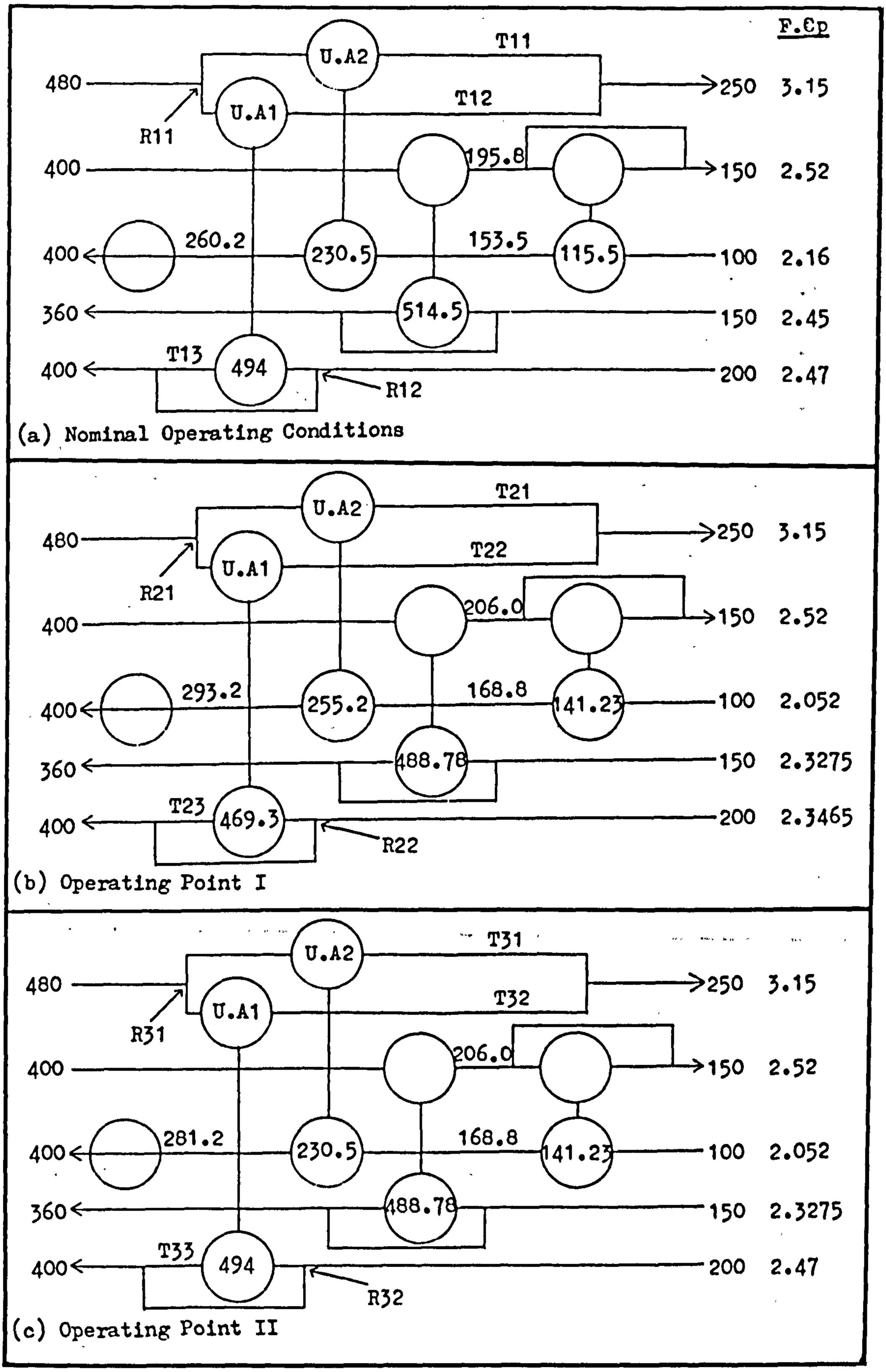


Figure 8.13 The performance of the network at three sets of operating conditions for variable U.A1 and U.A2 values.

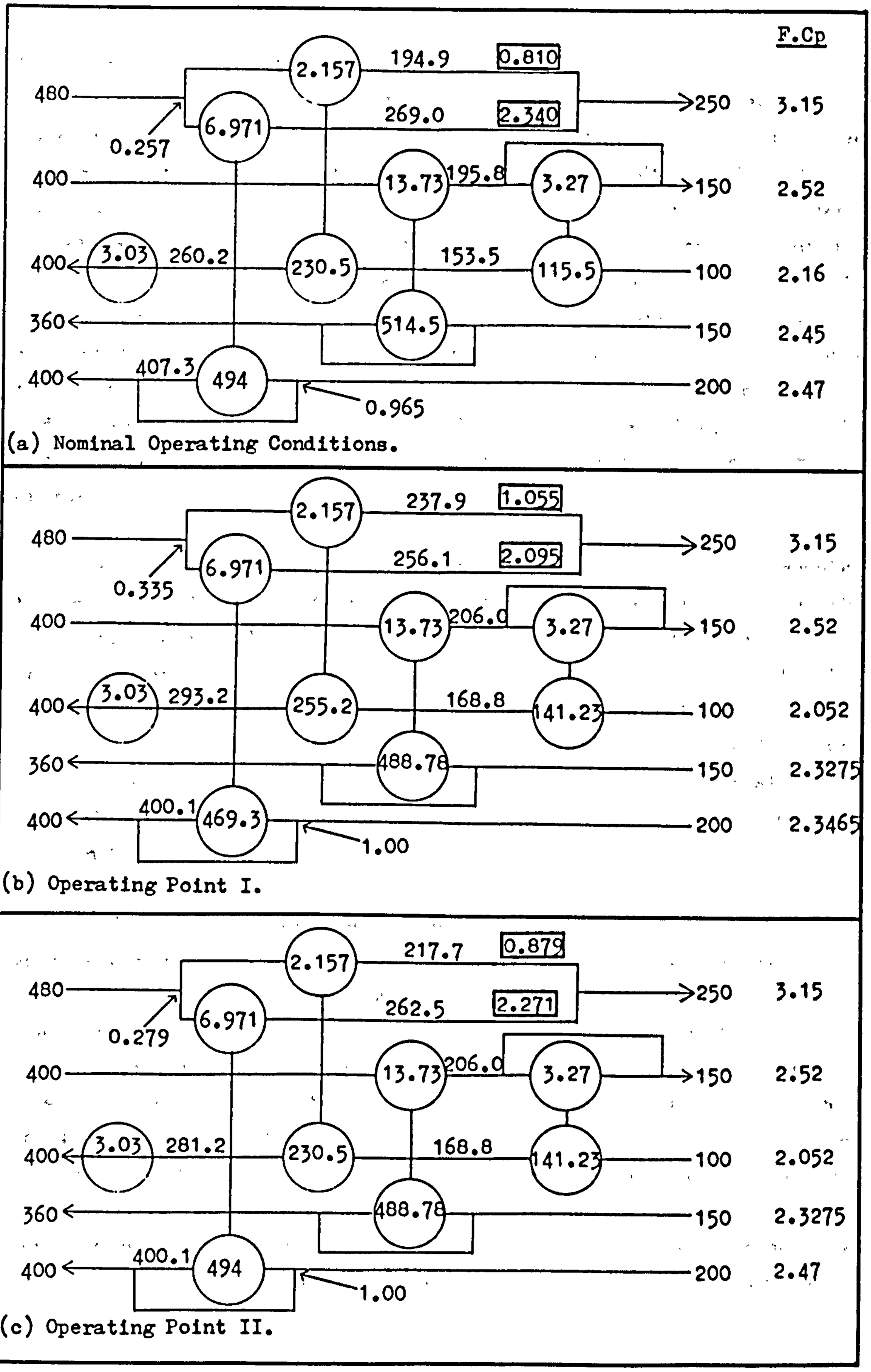


Figure 8.14 The performance of the optimum network at three sets of operating conditions.

8.3 Large Operating Ranges

In general if the operating range is relatively small, as are the operating ranges considered in Chapter 7, or if a network is required to operate at a number of discrete sets of inlet conditions in addition to a small continuous operating range, as in the combined network example in Chapter 7, then a network design which can operate for all the required inlet conditions with high heat recovery and simple control requirements can be produced. For large operating ranges this may not be the case. To ensure feasible operation of a network over the full operating range, which is essential, and to retain a simple and reliable control scheme, which is highly desirable, it may be necessary to sacrifice the target of high heat recovery for all operating conditions.

These general observations are borne out by the following example. The network given in Figure 8.15 (a) has been designed using the techniques developed in the present work and can operate over the full operating range given. The nominal conditions are as used in Pinched Problem 4 discussed in Chapter 4 but the operating range is considerably larger with hot stream F.Cp variations of 0.5 to 1.0 times nominal and cold stream variations of 1.0 to 1.5 times nominal. Also the inlet temperature of process stream 1 has been allowed to vary between 572 and 750.

The increased operating range includes sets of conditions for which the problem is unpinched. The performance of the network at one such set of conditions is shown in Figure 8.15 (b). For this set of conditions, the problem table indicates that the problem is unpinched with a minimum cold utility load of 77.8 needed. Using the network shown in Figure 8.15 (b) both heating and cooling is needed with a total utility use of 207.3. The usual requirement for unpinched problems is that the network should achieve the minimum possible utility use. The network structure used cannot accommmodate minimum possible utility use, as is shown in Figure 8.15 (c) with a negative temperature approach being required at the cold end of exchanger 1

independently of any chosen areas for the exchangers (when the two redundant heating utility exchangers are removed the network becomes a minimum number of units solution and the intermediate temperatures, therefore, are fixed).

If maximum heat recovery is to be achieved for this set of operating conditions, a different network structure must be used. No simple structure is possible using close to the minimum number of units. There is no guarantee that a network recovering the maximum amount of heat for this set of operating conditions would be able to operate over the full range, in which case a combined network possibly requiring optimizing control would be required. Certainly a novel control scheme would be required to isolate the redundant utility exchangers (see Appendix A) when the inlet conditions corresponded to an unpinched problem and to reorganize the regulatory control scheme. This would be more difficult to implement and potentially more unreliable than the simple regulatory control which is possible for the network given in Figure 8.15 (a).

Therefore, ensuring high heat recovery requires considerable extra capital cost due to extra units being needed and also poses control problems. It would seem likely that in practical situations the network given in Figure 8.15 (a) would be preferred, accepting that not all heat can be recovered for all unpinched parts of the operating range.

The set of operating conditions showing the worst utility use figures based on the difference between actual and minimum theoretical use (rather than on calculating the actual use as a percentage greater than theoretical minimum use as has usually been used) and the performance of the network at this set of conditions (note that this is not a cornerpoint) is shown in Figure 8.15 (d). For this set of conditions actual utility use is 231.5 compared with the theoretical minimum use of 55.50 (the problem is unpinched), a difference of 176.0. The designer must decide whether this is an acceptable level of utility use. In this case the utility use is probably acceptable since the

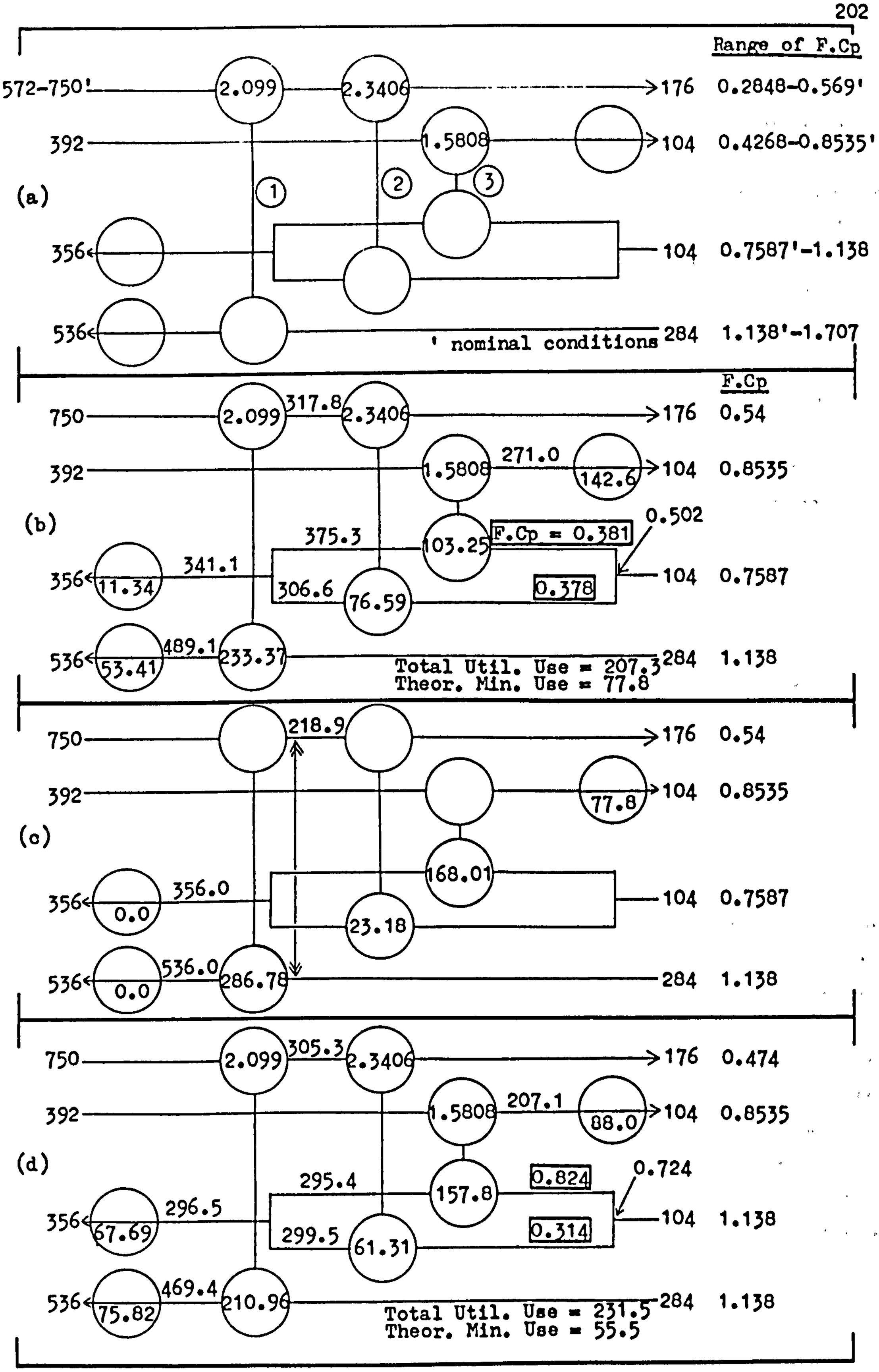


Figure 8.15 A network designed for a large operating range.

'excess' utility use of 176.0 is relatively small in comparison with the amount of heat transferred in the exchangers, 430.07, effectively 'saving' an extra utility load of 860.14 which would be required if no network were present. More heat (though not the maximum amount) can be recovered by increasing the exchanger areas, however, there will be limited scope for this as operation over the full operating range must be maintained.

Any network designed for a particular operating range may also have the ability to operate over a larger range (although possibly with higher utility use) and this may be a factor in choosing between alternative designs. In this situation a method of determining the size of the feasible space such as that proposed by Swaney and Grossmann (1982) is useful.

8.4 Additions and Improvements to the Design Method

8.4.1 Additional uncertain variables

In the present work the only uncertain variables considered are the process stream inlet temperatures and heat capacity flowrates as these are likely to be subject to the largest variations. The design method can however, accommodate other uncertain variables the most important of which are target temperatures and heat transfer coefficients.

The designer may not require all target temperatures to be met exacly and an acceptable range for each target temperature may be set. Referring to Chapter 3 it will be seen that for each target temperature which is not required to take a fixed value the degrees of freedom of the network are reduced by one and one less splitter or utility exchanger is required for regulatory control. If all the target temperatures are specified as ranges then in theory no bypasses are needed and the utility exchangers may be used for optimizing control, in practice the utility exchangers are used to bring hot streams to their highest acceptable target temperatures and cold streams

to their lowest acceptable target temperatures thus minimizing utility use.

To deal with variable target temperatures, a different form of computer modelling to that used in the present work is needed in which the variable target temperatures may be calculated for any particular set of inlet conditions i.e. a mode of operation similar to that used by standard flowsheeting packages. The bounds on the target temperatures may be treated as imposed constraints and optimizing split fractions may be manipulated to bring the target temperatures within their required ranges while minimizing utility use. A result of the change in the degrees of freedom analysis is that each subset of streams need not necessarily be connected to a utility exchanger. For the case in which not all target temperatures are specified a simple flexibility analysis is given by Linnhoff and Kotjabasakis (1984), although they do not give any method for determining exchanger areas.

Heat transfer coefficient variations may be dealt with in exactly the same way as inlet temperature and F.Cp variations, that is they may be manipulated to find the worst possible value of each network constraint function value in turn.

8.4.2 Heat exchanger models

In the present work, pure countercurrent single shell pass single tube pass exchangers are assumed and for feasible operation the standard log-mean temperature difference formulation (5.37) is applied. For more than a single shell pass or a single tube pass a correction factor is commonly applied to the log-mean temperature difference and this may easily be included in a computer modelling program. The use of a new model will have no effect on the conclusions reached in the present work; undersized exchangers will still give rise to infeasible split fraction values and the maximum heat will be transferred in an exchanger when no flow is bypassed round the exchanger. The only difference might be that the corrected log-mean temperature difference may have to be redefined for infeasible temperature approaches in a different way to that proposed in Appendix B.

8.4.3 Physical properties

In the present work it has been assumed that heat capacity values are independent of temperature. It would be relatively simple to include appropriate relationships for calculating heat capacities in the computer model if this were not a valid assumption.

Phase changes, that is boiling or condensing streams, could also be accommodated by appropriate reformulation of the network modelling equations.

8.4.4 A short-cut design method

For all of the design examples studied in the present work except for the specific examples cited in Chaper 6 all of the critical operating points identified were cornerpoints of the operating range. This implies that generally the network constraint functions are monotonic with respect to the uncertain inlet variables. If the designer is willing to risk missing the few cases in which this is not the case a short-cut design method based on the work of Grossmann (1977) can be used.

Starting from a computer model of a network at nominal operating conditions each uncertain inlet variable (θ) in turn may be perturbed slightly and the network remodelled. From this a numerical approximation to the partial derivative of each network constraint function (g) with respect to each uncertain variable can be determined.

If there are J network constraints and I uncertain variables a numerical approximation to each

$$\left(\frac{\partial g_j}{\partial \theta_i}\right) \qquad j\epsilon J,\ i\epsilon I \qquad (8.44)$$

can be found.

Three possible outcomes are possible: $\partial g_j/\partial \theta_i$ may be found to be positive, zero or negative. As the constraint functions are assumed to

be monotonic it can be assumed that each $\partial g_j/\partial\theta_i$ retains the same sign or remains zero over the entire range of θ (i.e. this is the definition of monotonicity). As each constraint value must be < 0 for feasibility the lower or upper bound on the inlet variable giving rise to the worst constraint value (i.e. that which is most infeasible or closest to infeasibility) can be immediately determined:

$$\begin{aligned}
&\text{If } \partial g_j/\partial\theta_i \text{ is +ve set } \theta_i \text{ to } \theta_i^U \\
&\partial g_i/\partial\theta_i \text{ is } 0 \quad g_i \text{ is independent of } \theta_i \text{ and } \theta_i \text{ can be set to } \theta_i^U \text{ or } \theta_i^L \\
&\partial g_i/\partial\theta_i \text{ is -ve set } \theta_i \text{ to } \theta_i^L
\end{aligned} \tag{8.45}$$

Thus for each constraint fuction a cornerpoint has been found which maximizes the constraint value. Some of the cornerpoints found are likely to be the same for different constraints and Grossmann (1977) gives an algorithm for merging all the critical cornerpoints into the minimum number of points.

The design may be made feasible for these critical points and, assuming that the constraints are monotonic with respect to the design variables and uncertain variables, a fully feasible network will result.

The advantage of this method is that it requires network modelling for only (I+1) sets of operating conditions, the disadvantage is that it misses the rare case in which the constraints are not monotonic. In general the extra costs of using an optimization routine to maximize the constraint functions will be negligible compared to the potential cost penalty of a network which cannot operate over the full operating range.

CHAPTER 9

CONCLUSIONS

At the end of Chapter 2 a number of requirements were listed which if satisfied by a new method for the design of flexible heat exchanger networks would ensure that the method was an improvement on existing techniques. It is now possible to review these requirement and determine whether they have been satisfied by the new design method presented in this work.

1) Ability to deal with both the sizing and structural problem.

As discussed in Chapter 6, the computer based method developed is able to distinguish between sizing and structural infeasibilities.

Techniques are developed in Chapter 6 and in Chapter 8 (for networks with design flexiblity) which can be used to oversize exchangers in a network just sufficiently to handle a required operating range and therefore the present work provides a comprehensive method for detecting and eliminating sizing infeasibilities.

For structural infeasibilities the situation is less clear cut. Structural infeasibilities can be detected using the methods developed in the present work but no universally applicable rules for their elimination can be given. Often, by considering a large number of base networks, alternatives which show only sizing infeasibilities can be retained and the networks showing structural infeasibilities eliminated, as in the unpinched design example based on problem 5SP1 presented in Chapter 7. In other cases, a combined network can be formed to eliminate the structural infeasibility while retaining only regulatory control. In some cases, however, possibly when designing for a large operating range, it seems probable that a network with switching or optimizing control may be needed. This class of problems have

not been addressed in the present work and would provide a fruitful area for future work.

2) Optimal balancing of capital and operating cost.

The networks designed using the new method do not feature (and are not intended to) optimal balancing of capital and operating cost. This objective implies a single 'best' network which is not an appropriate objective for the present work. The new method can be used to design a number of alternative flexible networks from which the best network may be selected by applying not only economic but also dynamic operability criteria. A balance between capital and operating cost is achieved by ensuring that all alternatives recover a high proportion of the recoverable heat for all operating conditions.

3) Correct assignment of bypasses to the network.

The degrees of freedom analysis in Chapter 3 gives information about how many bypasses need to be placed in a network for control purposes and identifies cases in which incorrect bypass placement makes adequate control impossible. As is emphasized by the examples presented in Chapter 7 not all exchangers in a network need necessarily be bypassed and the choice of which exchangers to bypass is crucial in designing for flexibility.

In many cases flexible designs can be produced which have zero degrees of freedom and therefore require regulatory control only with no utility use penalty resulting from leaving exchangers unbypassed.

The choice between a number of networks featuring alternative possible bypass placements cannot be made a priori and the design method based on computer modelling presented in the present work is the only current method which can select the best alternative.

4) Ensuring that the network can handle the full operating range, not just cornerpoints.

As discussed in Chapter 6 and demonstrated in Appendix D the computer program developed can detect infeasible operation at

points other than cornerpoints and is the only current method able to do this.

Thus it can be seen that with some minor reservations all of the requirements set out in Chapter 2 have been satisfied and therefore the design method developed in the present work does appear to be an advance on other existing methods.

A number of topics related to the main design problem have also been investigated and important results found.

1) The design method has been extended to deal with cases in which the uncertain variables are non-independent thus providing a method of ensuring that a network is not designed to handle impossible (or highly unlikely) operating conditions. The economic advantages of this approach are demonstrated in Chapter 8.

2) The design method is applicable to networks which have design flexibility (that is more than the minimum number of units or stream-splits) and this distinguishes it from some other current methods.

3) A new method for modifying the log-mean temperature difference formulation has been developed (Appendix B) and this has proved very successful when used in the network modelling program. A detailed investigation of the properties of the new function has not been carried out although it would seem to be better than other corrections which are currently used. It would be interesting to try the new formulation in other applications and compare its performance with other methods.

APPENDIX A

Redundant Utility Exchangers

The network shown in Figure A.1(a) has zero degrees of freedom as it stands and therefore the exchanger, apparently, need not be bypassed.

For the particular set of operating conditions shown, the problem table indicates that the problem is unpinched with only heating required, although for other operating points cooling is required. The exchanger U.A value has been calculated for this operating point and therefore the performance of the network at this point is as shown in Figure A.1(b) with a cooling load of 0.0 and a heating load of 600.0, which are the values indicated by the problem table.

Another point in a possible operating range at which the inlet temperature of the hot stream has been decreased by 10^{o} is shown in Figure A.1(c). The problem table indicates for this set of conditions that only heating is required with a minimum heat load of 610.0. Figure A.1(d) shows the performance of the network, using the same U.A value as previously, at this set of conditions. Although the total utility use is 610.0 (606.66 + 3.34) which corresponds to the minimum possible, it can be seen that a heating load of 3.34 is required in the cooling utility exchanger. This is a consequence of the heat exchanger being oversized for this particular duty. For the maximum possible heat to be recovered using only the heating utility exchanger a solution must be found at which the cooling heat load, C, is equal to zero. The cooling utility exchanger must be treated as if it were not present in the network and this effectively reduces the degrees of freedom by 1. To compensate for this a bypass must be introduced on the exchanger. The required operation of the network is shown in Figure A.1(e).

Thus, the following addition to the degrees of freedom analysis can be made:

If a given network has both heating and cooling utilities but the problem table indicates that for a particular part of the operating range the problem is unpinched and the designer wishes to recover the maximum amount of heat, then for this sub-range either the heating or the cooling utility exchangers are not needed (are redundant). In assigning stream-splits and bypasses to the network it must be ensured that the remaining network, when all the redundant utility exchangers have been removed from consideration, has zero degrees of freedom.

For example, if the original network has zero degrees of freedom and it is found that for some sets of operating conditions only heating is required then the degrees of freedom must be increased by the number of cooling utility exchangers. If within the operating range two points can be found one of which requires only heating and the other only cooling utilities, then the degrees of freedom must be increased by Max (No. cooling utility exchangers, No. heating utility exchangers).

For an unpinched problem where, in the interests of the number of exchangers used or the simplicity of the network, all of the available heat is not recovered both cooling and heating utilities are needed. For this case total utility use will be greater than the theoretical minimum and may be treated as for pinched problems.

In the computer programs developed in the present work it is possible to specify that whenever a set of operating conditions is encountered which gives rise to an unpinched problem, as detected by the problem table analysis, then the maximum amount of heat must be recovered and the redundant utility exchangers removed from consideration. Otherwise unpinched and pinched problems are treated in the same way.

Returning to Figure A.1(d) it can be seen that if the outlet temperature of the hot stream is not critical, advantage may be taken of the oversized exchanger. If the hot stream outlet temperature of 196.66 is acceptable (although the specified outlet is 200.0) then the

heating utility required to bring the cold stream to 500^{o} is reduced from 610.0 to 606.66. Inevitably in a flexible network exchangers will be oversized for many sets of operating conditions and it may be possible to relax some targets to achieve greater utility savings.

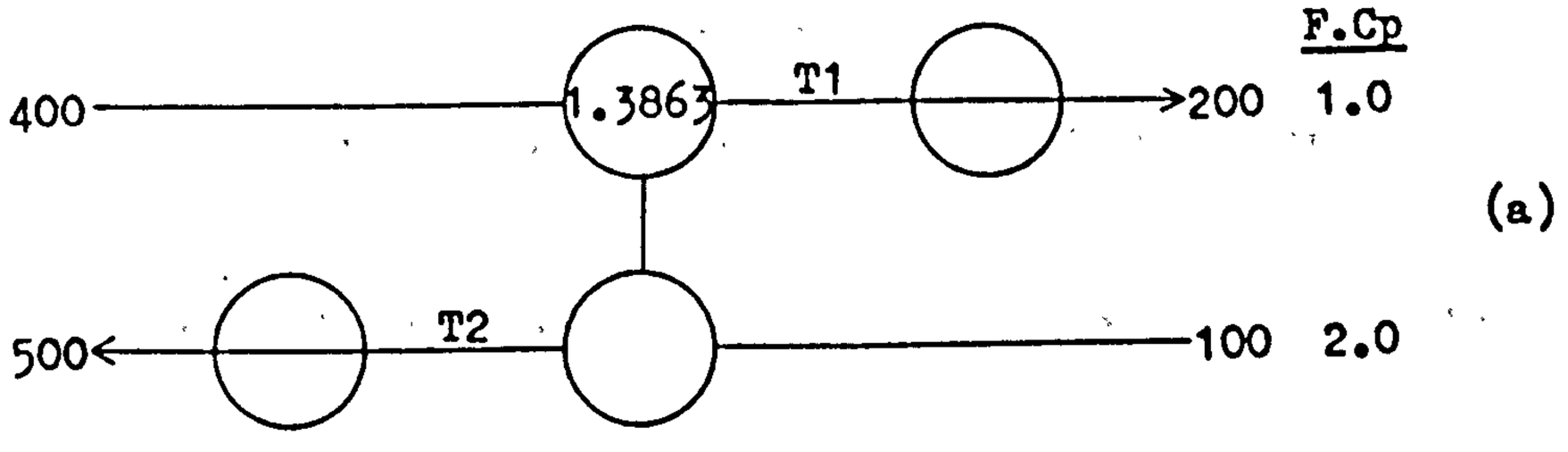
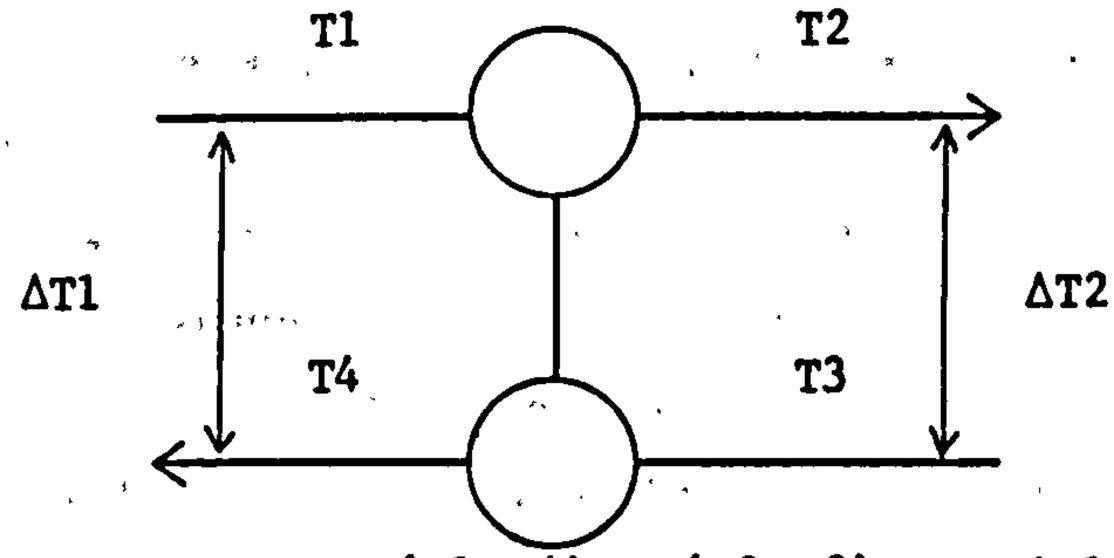


Figure A.1 A network demonstrating redundant utility exchangers.

APPENDIX B

Log-Mean Temperature Difference Formulation

The standard log-mean temperature difference formulation is as follows:



$$\Delta T_{LM} = \frac{(T1-T4) - (T2-T3)}{\ln\left(\frac{(T1-T4)}{(T2-T3)}\right)} = \frac{\Delta T1 - \Delta T2}{\ln\left(\frac{\Delta T1}{\Delta T2}\right)} \qquad (B.1)$$

In practice during the iterations to find a solution to a network's describing equations or, indeed, at the solution the temperatures accociated with an exchanger, T1, T2, T3 and T4 above, may take values for which the ΔT_{LM} formulation (B.1) cannot return a value.

For example if $(\Delta T1/\Delta T2) < 0$ then $\ln(\Delta T1/\Delta T2)$ cannot be calculated. Grossmann (1977) gives a modified formulation which allows for negative temperature approaches. The redefinition used is:

$$\Delta T_{LM} = \frac{\Delta T1 - \Delta T2}{Z} \quad \text{where } Z = \begin{cases} \ln(\Delta T1/\Delta T2) & \text{when } \Delta T1/\Delta T2 > 1-\varepsilon \\ \frac{\Delta T1/\Delta T2}{1-\varepsilon} + \ln(1-\varepsilon) - 1 & \text{when } \Delta T1/\Delta T2 < 1-\varepsilon \end{cases} \qquad (B.2)$$

This formulation produces a continuous function for ΔT_{LM}. Figure B.1 shows the variation of ΔT_{LM} with $\Delta T1$ for a fixed $\Delta T2$ value of 10 for the two ΔT_{LM} formulations (B.1) and (B.2). The modified formulation (B.2) has been calculated assuming $\varepsilon = 0.95$. It can be seen that above a $\Delta T1$ value of 0.5 the two functions are identical but for $\Delta T1$ values of < 0.5 Grossmann's formulation (B.2) departs from the standard formulation (B.1) and allows negative $\Delta T1$ values to be accommodated, whereas the formulation (B.1) is undefined for negative $\Delta T1$ values. As $\Delta T1$ becomes more negative the ΔT_{LM} value produced by

(B.2) asymptotically approaches 0.5 (in the general case as $\Delta T1 \rightarrow \infty$, $\Delta T_{LM} \rightarrow \Delta T2\ (1-\varepsilon)$).

Grossmann's formulation (B.2) is useful as it will return a value of ΔT_{LM} for any combination of any values (negative or positive) of $\Delta T1$ and $\Delta T2$. However this formulation is not ideal for the present work. One anomalous property of the function is that the value of ΔT_{LM} returned is dependent on the ordering of the temperature approaches $\Delta T1$ and $\Delta T2$. For example $\Delta T1 = 10$, $\Delta T2 = 0.1$ returns a different ΔT_{LM} value to $\Delta T1 = 0.1$, $\Delta T2 = 10$. This property may well lead to numerical difficulties. Another problem is that the ΔT_{LM} value has a lower bound. In the example given in Figure B.1 with $\Delta T2$ fixed at 10, if $\Delta T1$ is being controlled by a bypass and the required ΔT_{LM} value for a particular set of stream data is 0.4 then it can be demonstrated that there is no value of $\Delta T1$ which will give this value (the lowest ΔT_{LM} possible is 0.5). The numerical procedures would attempt to keep reducing $\Delta T1$ and hence ΔT_{LM} but this would soon lead to a very large negative $\Delta T1$ and failure because of overflow. To remedy this inadequacy it would be necessary to enable any ΔT_{LM} value to be achieved by a suitable choice of $\Delta T1$.

For positive $\Delta T1$ and $\Delta T2$ values the ΔT_{LM} value returned must be as the standard function B.1 but for negative values any suitable value may be returned for ΔT_{LM}, given that for all solutions found checks are made directly on $\Delta T1$ and $\Delta T2$ to determine whether they take feasible values. In the present work it is not sufficient to know that no feasible solution to a particular problem exists, the degree of infeasibility should vary in a regular manner so that the most infeasible solution can be found.

Taking the above factors into consideration a new ΔT_{LM} formulation has been developed.

A New Formulation

ΔT_{LM} can be found from a composite function the general form of which is shown in Figure B.2(a). The main features of this function are as follows:

1) If $\Delta T1 = \Delta T2$ then $[\ln(\Delta T1/\Delta T2)]^{-1}$ in (B.1) is undefined. This case is detected and when $\Delta T1/\Delta T2 \simeq 1.0$ ΔT_{LM} is set equal to $(\Delta T1 + \Delta T2)/2.0$, which, for $\Delta T1 = \Delta T2$, is the value to which ΔT_{LM} calculated from (B.1) tends.

2) For positive $\Delta T1$ and $\Delta T2$ values greater than a small value e ($\sim 10^{-5}$) the ΔT_{LM} value returned is as in the standard formulation (B.1).

3) The convention used is that if $\Delta T1$ is positive and $\Delta T2$ is negative, or vice versa, then the ΔT_{LM} value returned is equal to minus the ΔT_{LM} value returned by (B.1) using the modulus of $\Delta T1$ and $\Delta T2$ i.e. a negative ΔT_{LM} value. If both $\Delta T1$ and $\Delta T2$ are negative then the ΔT_{LM} value returned is equal to the ΔT_{LM} value returned by (B.1) using the modulus of $\Delta T1$ and $\Delta T2$ i.e. a positive value.

Referring to Figure B.2(a) (1), (2) and (3) above define the function for $\Delta T1 > e$ and $\Delta T1 < \varepsilon$. For $\Delta T1 = 0$, $\ln(\Delta T1/\Delta T2)$ is undefined therefore (B.1) cannot be applied. Therefore:

4) A linear section is included around $\Delta T_{LM} = 0.0$ bridging the $\Delta T_{LM} = 0.0$ axis and making the function continuous in this region.

Expanding on Figure B.2(a), if the value of $\Delta T2$ is allowed to vary a family of curves is produced as indicated in Figure B.2(b).

This new formulation has the advantages that:

(a) A value of ΔT_{LM} will be returned for any combination of any values of $\Delta T1$ and $\Delta T2$.

(b) The value of ΔT_{LM} returned is independent of the order in which $\Delta T1$ and $\Delta T2$ are specified.

i.e. The value returned for $\Delta T1 = x$, $\Delta T2 = y$ is the same as the value returned for $\Delta T1 = y$, $\Delta T2 = x$ for any x and y.

(c) For a fixed value of $\Delta T1$ the value of ΔT_{LM} is unbounded, that

is by an appropriate choice of $\Delta T2$ <u>any</u> value of ΔT_{LM} can be produced.

(d) ΔT_{LM} varies monotonically with $\Delta T1$ and $\Delta T2$ thereby enabling the 'most' infeasible solution for a particular problem to be found.

The new formulation, which is used in the heat exchanger network modelling program, has been found to be very reliable and has made the equation solving routine very robust.

In very rare cases duplicate solutions can be produced and so it is important to have a sensible initial guess of the solution to a problem.

A subroutine written in FORTRAN77 which will return a ΔT_{LM} value based on this new formulation follows.

```
      SUBROUTINE DELTA(DELT1,DELT2,DELTAX)
C
C*    SUBROUTINE CALCULATES LOG-MEAN TEMPERATURE DIFFERENCE WITH
C*    CORRECTIONS FOR DISCONTINUITIES,SINGULARITIES AND UNDEFINED
C*    REGIONS
C*
C*
C*    ARGUMENTS:
C*          IN: DELT1,DELT2: THE EXCHANGER 'HOT' AND 'COLD' END
C*              TEMPERATURE APPROACHES. THE 'HOT' STREAM TEMPERATURE
C*              MINUS THE 'COLD' STREAM TEMPERATURE IN EACH CASE.
C*
C*          OUT: DELTAX: THE MODIFIED LOG-MEAN TEMPERATURE DIFFERENCE
C*               VALUE.
C*
C*
C
      PDELTA(XX,YY)=(XX-YY)/ALOG(XX/YY)
      VDELTX(XX,YY)=-(ABS(XX)-YY)/ALOG(ABS(XX)/YY)
      VDELTY(XX,YY)=-(XX-ABS(YY))/ALOG(XX/ABS(YY))
      E=1.0E-05
      IF(DELT1.LT.E.AND.DELT1.GT.-E.AND.DELT2.LT.E.AND.DELT2.GT.-E)THEN
      DELTAX=0.0
      ELSEIF(DELT1.LT.E.AND.DELT1.GT.-E)THEN
      IF(DELT2.GT.0)THEN
      DELTAX=((PDELTA(E,DELT2)-VDELTX(-E,DELT2))*(DELT1+E)/
     +(2.0*E))+VDELTX(-E,DELT2)
      ELSE
      DELTAX=((VDELTY(E,DELT2)+PDELTA(-E,DELT2))*(DELT1+E)/
     +(2.0*E))-PDELTA(-E,DELT2)
      ENDIF
      ELSEIF(DELT2.LT.E.AND.DELT2.GT.-E)THEN
      IF(DELT1.GT.0)THEN
      DELTAX=((PDELTA(DELT1,E)-VDELTY(DELT1,-E))*(DELT2+E)/
     +(2.0*E))+VDELTY(DELT1,-E)
      ELSE
      DELTAX=((VDELTX(DELT1,E)+PDELTA(DELT1,-E))*(DELT2+E)/
     +(2.0*E))-PDELTA(DELT1,-E)
      ENDIF
      ELSEIF(ABS(DELT1/DELT2).LT.(1.0+E).AND.ABS(DELT1/DELT2).GT.(1.0-E
     +))THEN
      IF(DELT1.GT.0.AND.DELT2.LT.0.OR.DELT1.LT.0.AND.DELT2.GT.0)THEN
      DELTAX=-(ABS(DELT1)+ABS(DELT2))/2.0
      ELSE
      DELTAX=ABS(DELT1+DELT2)/2.0
      ENDIF
      ELSEIF(DELT1.GT.0.AND.DELT2.LT.0)THEN
      DELTAX=VDELTY(DELT1,DELT2)
      ELSEIF(DELT1.LT.0.AND.DELT2.GT.0)THEN
      DELTAX=VDELTX(DELT1,DELT2)
      ELSEIF(DELT1.LT.0.AND.DELT2.LT.0)THEN
      DELTAX=(-PDELTA(DELT1,DELT2))
      ELSE
      DELTAX=PDELTA(DELT1,DELT2)
      ENDIF
      RETURN
      END
```

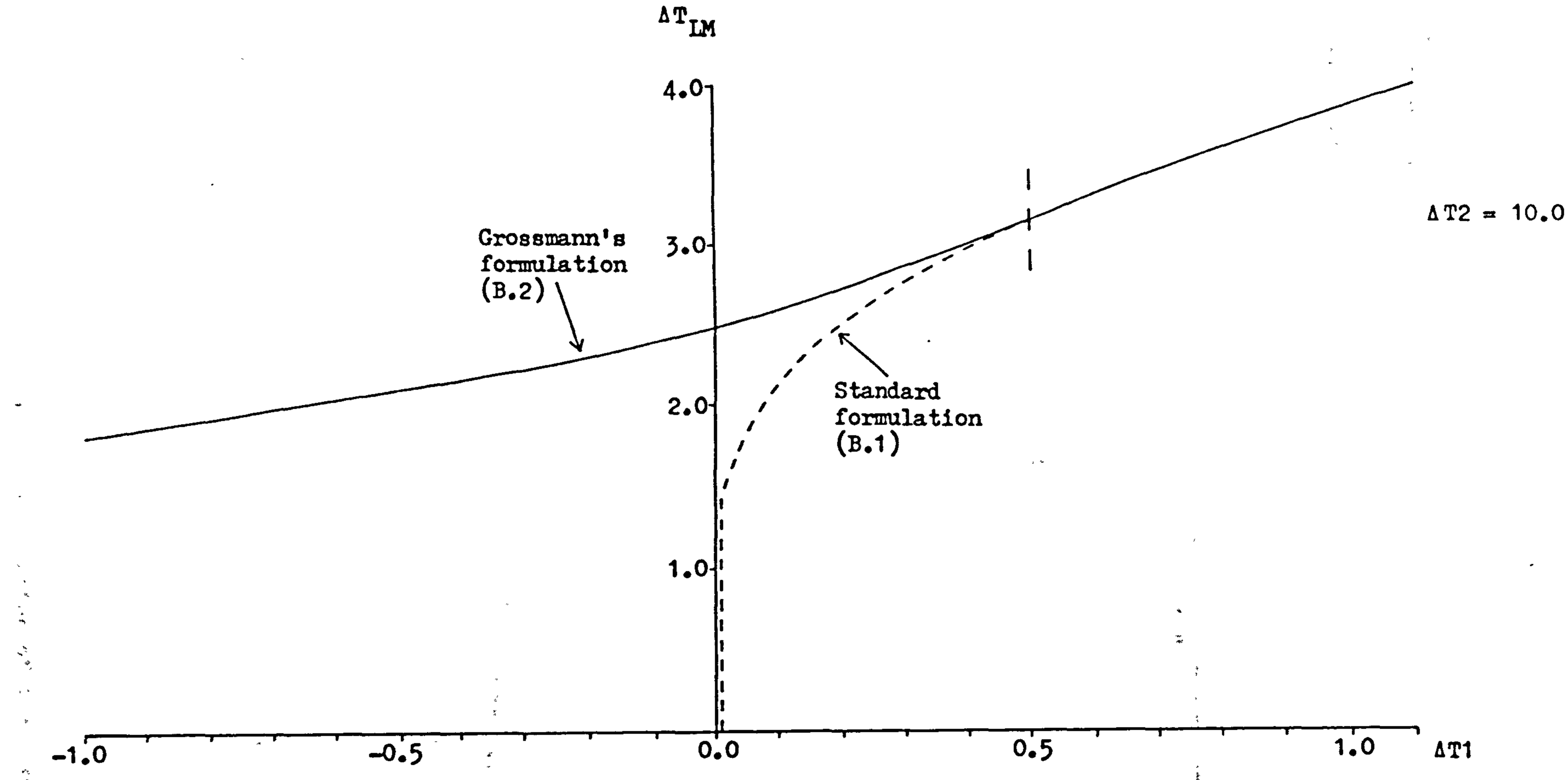
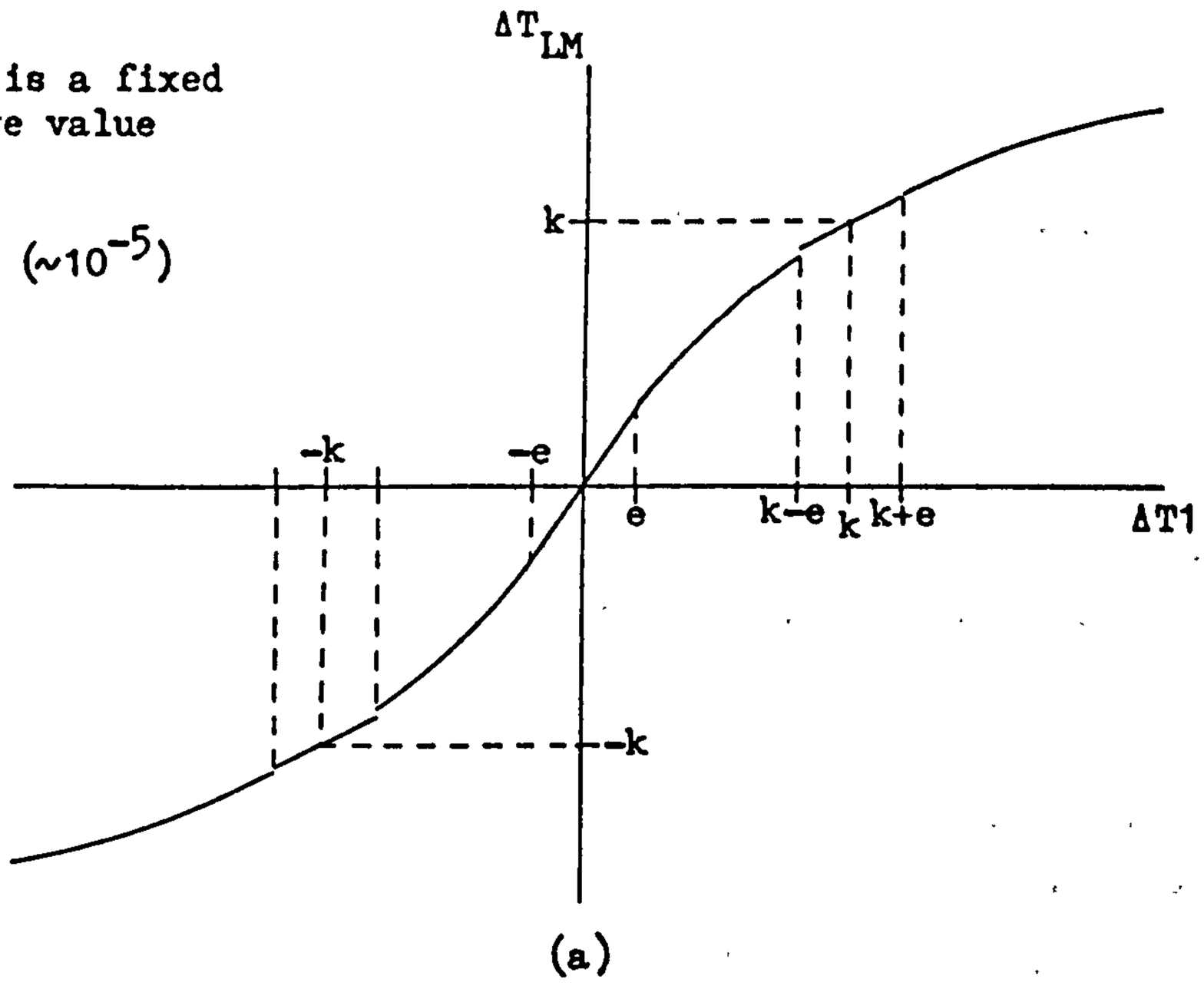


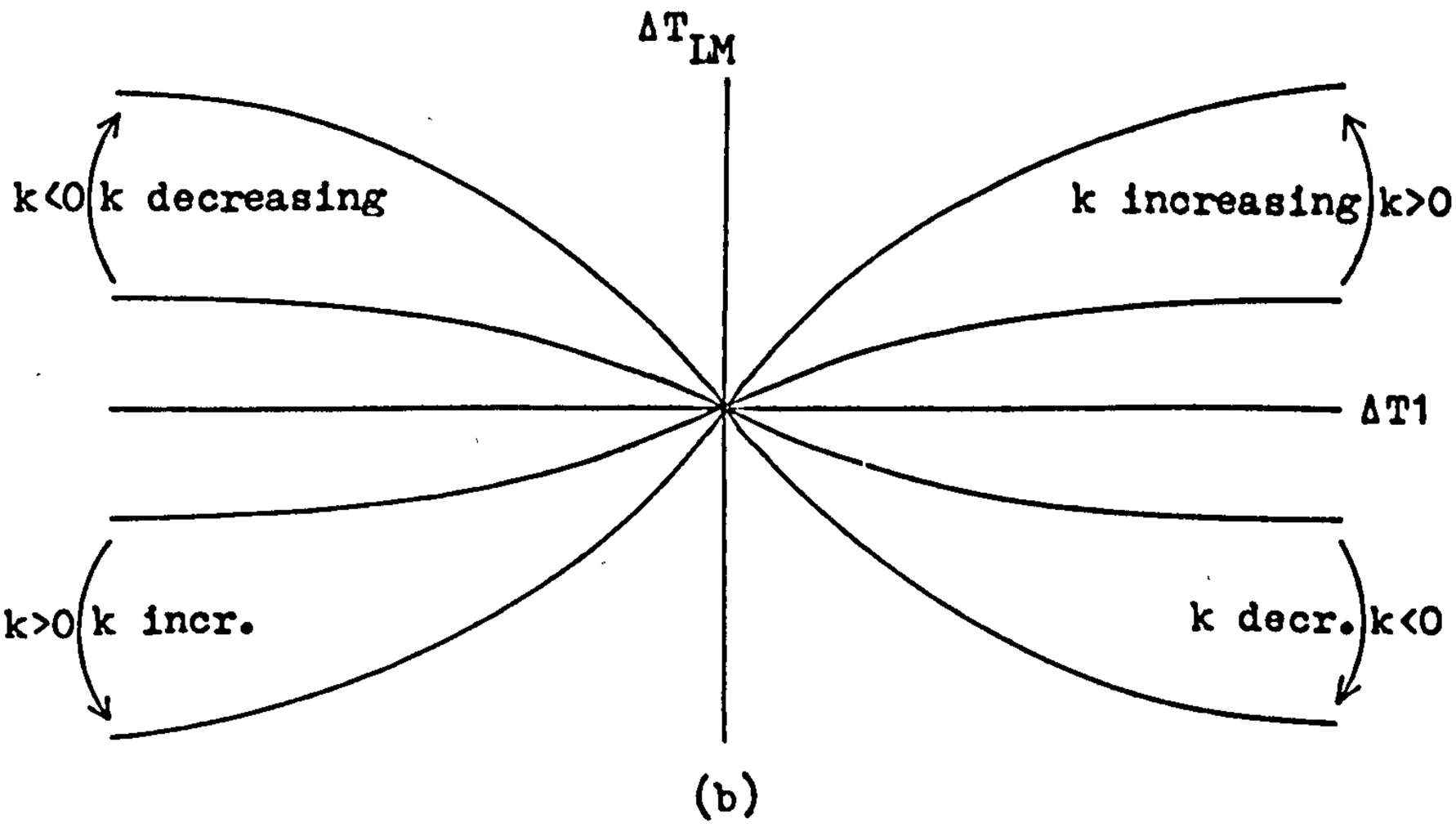
Figure B.1 A comparison of the standard formulation of the log-mean temperature difference with Grossmann's formulation.

$\Delta T2=k$, k is a fixed positive value

e small ($\sim 10^{-5}$)



(a)



(b)

Figure B.2 A new formulation for the calculation of the log-mean temperature difference.

APPENDIX C

Use of Program HERON for Cornerpoint Checking

The network used to demonstrate the program is that given in Figure 3.6 which was discussed in Chapter 3 as one in which the choice of which split fraction to use for optimizing control was not immediately apparent. The purpose of this example is to demonstrate the way in which the program chooses the correct optimizing controller, models the operation of the network at a particular cornerpoint (which includes modelling of optimizing control), and then checks other cornerpoints in turn to try to find a cornerpoint for which operation is infeasible.

The network used is given in Figure C.1 together with the input data required by the program. For details on the form of the input data see Chapter 5 and the listing of Heron in Appendix F. Points to note are:

1) The required operating range consists of both inlet temperature and F.Cp variations.

2) If a temperature approach of less than 2.5 (ΔT_{min}) is required for any set of operating conditions this constitutes infeasible operation.

3) If for any set of operating conditions the actual total utility use is more than 15% (P) greater than the theoretical minimum use calculated for ΔT_{min} this constitutes infeasible operation.

The output from the program for this problem follows with points to note indicated (n). These are:

(1) The particular combination of inlet conditions corresponding

to cornerpoint 1, as defined internally by the program is selected.

(2) A routine based on Linnhoff's problem table approach is used to calculate minimum utility use requirements directly from the process stream data.

(3) The degrees of freedom of the network are greater than zero and a search is being made to select a split fraction for use as an optimizing control. The split fractions of units number 1 and 3 (refer to Figure C.1) are found to produce non-independent describing equations as indicated by a singular initial Jacobian matrix. This confirms the conclusions given in Chapter 3 where these split fractions were found to be inappropriate for use as optimizing controllers.

(4)-(5) The equation solving routine has converged to the required accuracy. The substream data and data relating to each unit are given. Note that a negative utility heat load indicates that cooling is required and a positive load indicates that heating is required.

To determine whether the solution is feasible the constraint function values are given. As EØ4UAF is a minimization routine, the form of the constraint values is that they must be greater than 0.0 unlike the mathematical development of Chapter 4 in which maximization was assumed. Constraints 9 to 11 are the split fraction values.

The split fraction of unit 5 has been found to be acceptable for use as optimizing control (as in Chapter 3) and it takes its initial guessed value of 0.5.

All the constraints are feasible except for the utility use which is 15.6% > minimum whereas it must be no greater than 15.0% > minimum.

(6) The optimization routine EØ4UAF is called to manipulate the

optimizing split fraction in an attempt to find a feasible operating point.

(6)-(7) A feasible point is found. All the constraints are feasible including that for utility use which is just on the limit of 15% > minimum. The split fraction of unit 5 has been changed from 0.5 to 0.9155. The program does not attempt to find the minimum possible utility use for this set of inlet conditions: as soon as a feasible point is found the optimization stops.

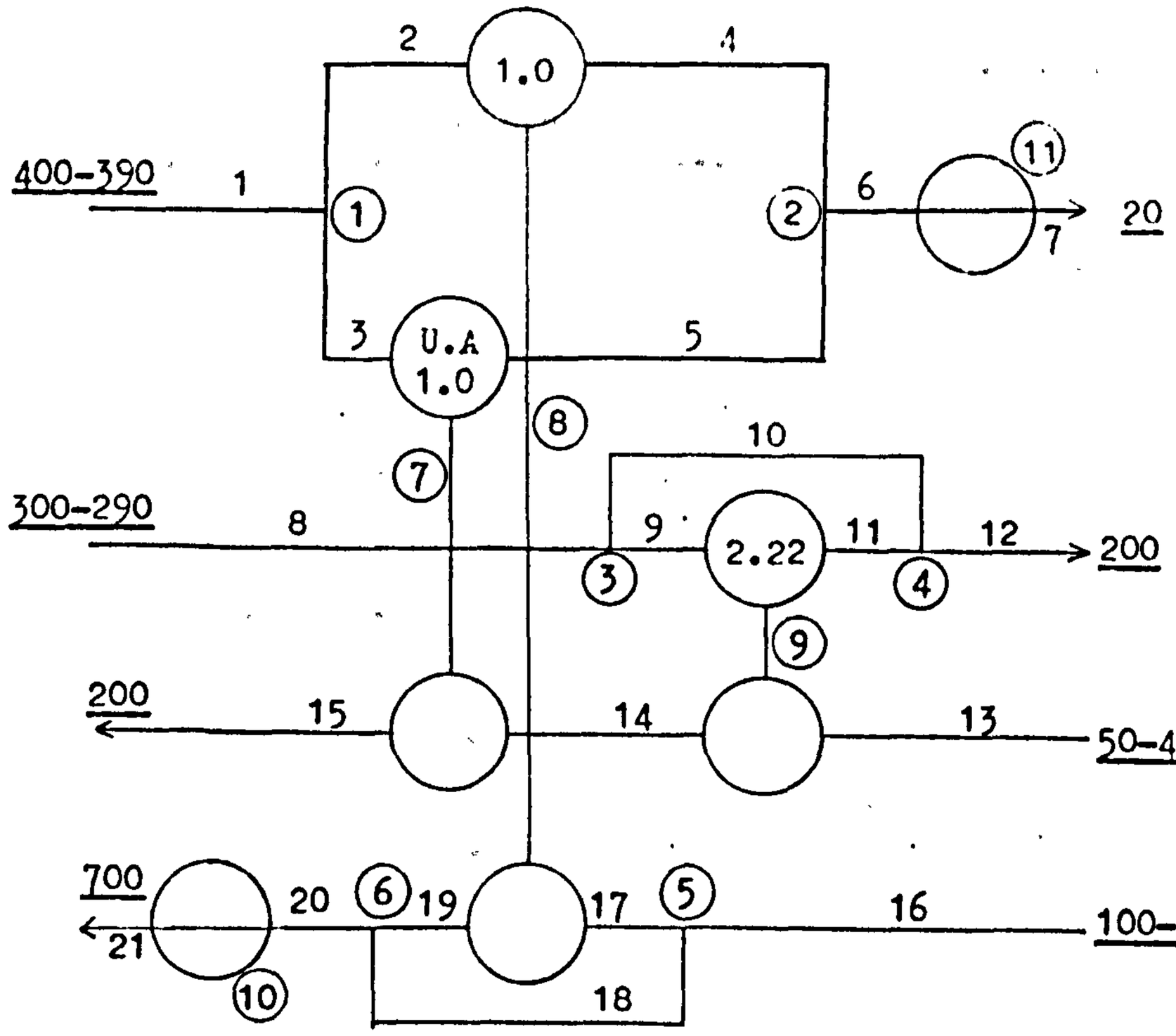
(8) The inlet conditions are changed to those corresponding to cornerpoint 2. These are identical to those for cornerpoint 1 except that the inlet temperature of process stream 1 is now 390 rather than 400.

(8)-(9) The network is modelled for these inlet conditions and operation is found to be feasible.

(10)-(11) The networkis unable to handle cornerpoint 3 with the utility use constraint being violated.

(11)-(12) No feasible point can be found by the optimizing routine. The minimum possible utility use is 15.3% > minimum which exceeds the specified maximum value of 15%. The program stops at this cornerpoint as the designer must decide whether to redifine the utility use constraint or alter the network.

The minimum possible utility use for this operating point is achieved for an optimizing split fraction of 1.0 and, in fact, for all operating conditions this is the case and the optimizing split fraction (the cold stream bypass on the exchanger unit number 8) need not therefore be present, as was the case for the examples discussed in Chapter 7.



Range of F.Cp

1.0-1.01

2.0-2.01

2.0-2.01

3.0-3.01

T - Temperatures ; n - Substream numbers ; (m) - Unit numbers

11,21	Number of units , Number of substreams
1,1,0,2,3	
2,4,5,6,0	
3,8,0,9,10	
4,10,11,12,0	
5,16,0,17,18	
6,18,19,20,0	Process matrix
7,3,14,5,15	
8,2,17,4,19	
9,9,13,11,14	
10,20,0,21,0	
11,6,0,7,0	
4	Number of process streams
1,1,1.01,400,390	
8,2,2.01,300,290	Process stream inlet data
13,2,2.01,50,45	
16,3,3.01,100,90	
20,200,200,700	Process stream target temperatures
2.5,15	ΔT_{min} , P
1,1,1	U for each exchanger
1,1,2.22	A
1,1000	Starting cornerpoint , Finishing cornerpoint

Figure C.1 The network used in the example and the input data required by the program.

****
*
* CORNERPOINT NUMBER 1
*
****

(1)

FOR A MIN. TEMP. APPROACH OF 2.50
(COLD STREAM) PINCH TEMP. IS 50.00
MINIMUM HOT UTILITY USAGE IS 1552.50
MINIMUM COLD UTILITY USAGE IS 32.50
MINIMUM TOTAL UTILITY USAGE IS 1585.00

(2)

XXXXX EQUATIONS MAY NOT BE INDEPENDENT XXXXX
FOR DEGREES OF FREEDOM

SPLIT FRACTION UNIT NUMBER
1 1

(3)

XXXXX EQUATIONS MAY NOT BE INDEPENDENT XXXXX
FOR DEGREES OF FREEDOM

SPLIT FRACTION UNIT NUMBER
2 3

BROYDN HAS CONVERGED TO REQUIRED ACCURACY (4)

EXIT FROM BROYDN AFTER 27 ITERATIONS

STREAM NO.	TEMPERATURE	FCP
1	400.00	1.00
2	400.00	.52
3	400.00	.48
4	162.39	.52
5	190.64	.48
6	175.88	1.00
7	20.00	1.00
8	300.00	2.00
9	300.00	.99
10	300.00	1.01
11	98.76	.99
12	200.00	2.00
13	50.00	2.00
14	150.00	2.00
15	200.00	2.00
16	100.00	3.00
17	100.00	1.50
18	100.00	1.50
19	182.75	1.50
20	141.37	3.00
21	700.00	3.00

HEAT EXCHANGER	UNIT NO.	HEAT LOAD	U	AREA
1	7	100.00	1.00	1.00
2	8	124.12	1.00	1.00
3	9	200.00	1.00	2.22

UTILITY EXCHANGER	UNIT NO.	HEAT LOAD
1	10	1675.88
2	11	-155.88

TOTAL UTILITY USAGE IS 1831.76

SPLITTER	UNIT NO.	SPLIT FRACTION
1	1	.522
2	3	.497
3	5	.500

```
CONSTRAINT VALUES (C) (SHOULD ALL BE +VE
EXCEPT CC(12) WHICH SHOULD BE IN RANGE 0.000 TO    .150)

XXXXX    NEXT VALUE IS INFEASIBLE  XXXXX
NO.12   (UTILITY USAGE)                C =  -.568E-02  (  15.6 P.C.> MIN  )
NO. 1   (HEAT LOAD,UNIT NO.   7 )      C =   .100E+01  (  HT.LOAD =  100.0)
NO. 2   (HEAT LOAD,UNIT NO.   8 )      C =   .124E+01  (  HT.LOAD =  124.1)
NO. 3   (HEAT LOAD,UNIT NO.   9 )      C =   .200E+01  (  HT.LOAD =  200.0)
NO. 4   (UNIT   7   HOT END DELT)      C =   .198E+01  (  200.0    DEGREES  )
NO. 5   (UNIT   7   COLD END DELT)     C =   .381E+00  (   40.6    DEGREES  )
NO. 6   (UNIT   8   HOT END DELT)      C =   .215E+01  (  217.3    DEGREES  )
NO. 7   (UNIT   8   COLD END DELT)     C =   .599E+00  (   62.4    DEGREES  )
NO. 8   (UNIT   9   HOT END DELT)      C =   .148E+01  (  150.0    DEGREES  )
NO. 9   (UNIT   9   COLD END DELT)     C =   .463E+00  (   48.8    DEGREES  )

SPLIT FRACTIONS USED AS DEGREES OF FREEDOM

     UNIT NO.      SPLIT FRACTION

          5              .5000

CALCULATED SPLIT FRACTIONS

     UNIT NO.      SPLIT FRACTION

          1              .5224
          3              .4969

** CURRENT VALUES ARE INFEASIBLE   (5)

CALLING OPTIMIZATION ROUTINE E04UAF (FROM NAG8F)

FEASIBLE POINT FOUND                                              (6)

==========================================
=              FINAL     RESULTS         =
==========================================
BROYDN HAS CONVERGED TO REQUIRED ACCURACY

EXIT FROM BROYDN AFTER    14 ITERATIONS

  STREAM NO.     TEMPERATURE        FCP
       1            400.00          1.00
       2            400.00           .52
       3            400.00           .48
       4            153.72           .52
       5            190.64           .48
       6            171.35          1.00
       7             20.00          1.00
       8            300.00          2.00
       9            300.00           .99
      10            300.00          1.01
      11             98.76           .99
      12            200.00          2.00
      13             50.00          2.00
      14            150.00          2.00
      15            200.00          2.00
      16            100.00          3.00
      17            100.00          2.75
      18            100.00           .25
      19            146.84          2.75
      20            142.88          3.00
      21            700.00          3.00

HEAT EXCHANGER  UNIT NO.    HEAT LOAD        U        AREA
       1            7          100.00        1.00      1.00
       2            8          128.65        1.00      1.00
       3            9          200.00        1.00      2.22

UTILITY EXCHANGER  UNIT NO.    HEAT LOAD
       1             10          1671.35
       2             11          -151.35

TOTAL UTILITY USAGE IS             1822.70

SPLITTER   UNIT NO.   SPLIT FRACTION
```

1	1	.522
2	3	.497
3	5	.915

CONSTRAINT VALUES (C) (SHOULD ALL BE +VE
EXCEPT CC(12) WHICH SHOULD BE IN RANGE 0.000 TO .150)

```
NO.12  (UTILITY USAGE)              C =  .292E-04 ( 15.0 P.C.> MIN )
NO. 1  (HEAT LOAD,UNIT NO.  7 )     C =  .100E+01 ( HT.LOAD =  100.0)
NO. 2  (HEAT LOAD,UNIT NO.  8 )     C =  .129E+01 ( HT.LOAD =  128.6)
NO. 3  (HEAT LOAD,UNIT NO.  9 )     C =  .200E+01 ( HT.LOAD =  200.0)
NO. 4  (UNIT  7  HOT END DELT)      C =  .198E+01 ( 200.0   DEGREES )
NO. 5  (UNIT  7  COLD END DELT)     C =  .381E+00 (  40.6   DEGREES )
NO. 6  (UNIT  8  HOT END DELT)      C =  .251E+01 ( 253.2   DEGREES )
NO. 7  (UNIT  8  COLD END DELT)     C =  .512E+00 (  53.7   DEGREES )
NO. 8  (UNIT  9  HOT END DELT)      C =  .148E+01 ( 150.0   DEGREES )
NO. 9  (UNIT  9  COLD END DELT)     C =  .463E+00 (  48.8   DEGREES )
```

SPLIT FRACTIONS USED AS DEGREES OF FREEDOM

UNIT NO.	SPLIT FRACTION
5	.9155

CALCULATED SPLIT FRACTIONS

UNIT NO.	SPLIT FRACTION
1	.5224
3	.4969

** ALL CONSTRAINTS ARE FEASIBLE (7)

```
**************************************************
*                                                *
*     CORNERPOINT NUMBER     2                   *   (8)
*                                                *
**************************************************
```

FOR A MIN. TEMP. APPROACH OF	2.50
(COLD STREAM) PINCH TEMP. IS	50.00
MINIMUM HOT UTILITY USAGE IS	1562.50
MINIMUM COLD UTILITY USAGE IS	32.50
MINIMUM TOTAL UTILITY USAGE IS	1595.00

BROYDN HAS CONVERGED TO REQUIRED ACCURACY

EXIT FROM BROYDN AFTER 15 ITERATIONS

STREAM NO.	TEMPERATURE	FCP
1	390.00	1.00
2	390.00	.49
3	390.00	.51
4	145.95	.49
5	194.22	.51
6	170.61	1.00
7	20.00	1.00
8	300.00	2.00
9	300.00	.99
10	300.00	1.01
11	98.76	.99
12	200.00	2.00
13	50.00	2.00
14	150.00	2.00
15	200.00	2.00
16	100.00	3.00
17	100.00	2.75
18	100.00	.25
19	143.47	2.75
20	139.80	3.00
21	700.00	3.00

HEAT EXCHANGER	UNIT NO.	HEAT LOAD	U	AREA

1	7	100.00	1.00	1.00
2	8	119.39	1.00	1.00
3	9	200.00	1.03	2.22

UTILITY EXCHANGER	UNIT NO.	HEAT LOAD
1	10	1680.61
2	11	-150.61

TOTAL UTILITY USAGE IS 1831.21

SPLITTER	UNIT NO.	SPLIT FRACTION
1	1	.489
2	3	.497
3	5	.915

CONSTRAINT VALUES (C) (SHOULD ALL BE +VE
EXCEPT CC(12) WHICH SHOULD BE IN RANGE 0.000 TO .150)

```
NO.12  (UTILITY USAGE)                  C =  .190E-02 ( 14.8 P.C.> MIN )
NO. 1  (HEAT LOAD,UNIT NO.  7 )         C =  .100E+01 ( HT.LOAD = 100.0)
NO. 2  (HEAT LOAD,UNIT NO.  8 )         C =  .119E+01 ( HT.LOAD = 119.4)
NO. 3  (HEAT LOAD,UNIT NO.  9 )         C =  .200E+01 ( HT.LOAD = 200.0)
NO. 4  (UNIT  7  HOT END DELT)          C =  .188E+01 ( 190.0   DEGREES )
NO. 5  (UNIT  7  COLD END DELT)         C =  .417E+00 (  44.2   DEGREES )
NO. 6  (UNIT  8  HOT END DELT)          C =  .244E+01 ( 246.5   DEGREES )
NO. 7  (UNIT  8  COLD END DELT)         C =  .434E+00 (  45.9   DEGREES )
NO. 8  (UNIT  9  HOT END DELT)          C =  .148E+01 ( 150.0   DEGREES )
NO. 9  (UNIT  9  COLD END DELT)         C =  .463E+00 (  48.8   DEGREES )
```

SPLIT FRACTIONS USED AS DEGREES OF FREEDOM

UNIT NO.	SPLIT FRACTION
5	.9155

CALCULATED SPLIT FRACTIONS

UNIT NO.	SPLIT FRACTION
1	.4892
3	.4969

** ALL CONSTRAINTS ARE FEASIBLE (9)

```
**************************************************
*                                                *
*     CORNERPOINT NUMBER     3                   *   (10)
*                                                *
**************************************************
```

```
FOR A MIN. TEMP. APPROACH OF              2.50
(COLD STREAM) PINCH TEMP. IS             50.00
MINIMUM HOT UTILITY USAGE IS           1572.50
MINIMUM COLD UTILITY USAGE IS            32.50
MINIMUM TOTAL UTILITY USAGE IS         1605.00
```

BROYDN HAS CONVERGED TO REQUIRED ACCURACY

EXIT FROM BROYDN AFTER 18 ITERATIONS

STREAM NO.	TEMPERATURE	FCP
1	400.00	1.00
2	400.00	.39
3	400.00	.61
4	128.08	.39
5	204.85	.61
6	175.29	1.00
7	20.00	1.00
8	290.00	2.00

9	290.00	.89
10	290.00	1.11
11	87.42	.89
12	200.00	2.00
13	50.00	2.00
14	140.00	2.00
15	200.00	2.00
16	100.00	3.00
17	100.00	2.75
18	100.00	.25
19	138.13	2.75
20	134.90	3.00
21	700.00	3.00

HEAT EXCHANGER	UNIT NO.	HEAT LOAD	U	AREA
1	7	120.00	1.00	1.00
2	8	104.71	1.00	1.00
3	9	180.00	1.00	2.22

UTILITY EXCHANGER	UNIT NO.	HEAT LOAD
1	10	1695.29
2	11	-155.29

TOTAL UTILITY USAGE IS 1850.57

SPLITTER	UNIT NO.	SPLIT FRACTION
1	1	.385
2	3	.444
3	5	.915

CONSTRAINT VALUES (C) (SHOULD ALL BE +VE
EXCEPT CC(12) WHICH SHOULD BE IN RANGE 0.000 TO .150)

```
XXXXX   NEXT VALUE IS INFEASIBLE  XXXXX
NO.12   (UTILITY USAGE)              C = -.301E-02 (  15.3 P.C.> MIN )
NO. 1   (HEAT LOAD,UNIT NO.  7 )     C =  .120E+01 (  HT.LOAD =  120.0)
NO. 2   (HEAT LOAD,UNIT NO.  8 )     C =  .105E+01 (  HT.LOAD =  104.7)
NO. 3   (HEAT LOAD,UNIT NO.  9 )     C =  .180E+01 (  HT.LOAD =  180.0)
NO. 4   (UNIT  7  HOT END DELT)      C =  .198E+01 (  200.0   DEGREES  )
NO. 5   (UNIT  7  COLD END DELT)     C =  .623E+00 (   64.8   DEGREES  )
NO. 6   (UNIT  8  HOT END DELT)      C =  .259E+01 (  261.9   DEGREES  )
NO. 7   (UNIT  8  COLD END DELT)     C =  .256E+00 (   28.1   DEGREES  )
NO. 8   (UNIT  9  HOT END DELT)      C =  .148E+01 (  150.0   DEGREES  )
NO. 9   (UNIT  9  COLD END DELT)     C =  .349E+00 (   37.4   DEGREES  )
```

SPLIT FRACTIONS USED AS DEGREES OF FREEDOM

UNIT NO.	SPLIT FRACTION
5	.9155

CALCULATED SPLIT FRACTIONS

UNIT NO.	SPLIT FRACTION
1	.3851
3	.4443

** CURRENT VALUES ARE INFEASIBLE (11)

CALLING OPTIMIZATION ROUTINE E04UAF (FROM NAG8F)

EXITING FROM E04UAF AFTER 10 CYCLES

```
==========================================
=              FINAL     RESULTS         *
==========================================
```

BROYDN HAS CONVERGED TO REQUIRED ACCURACY

EXIT FROM BROYDN AFTER 12 ITERATIONS

STREAM NO.	TEMPERATURE	FCP
1	400.00	1.00

2	400.00	.39
3	400.00	.61
4	127.56	.39
5	204.85	.61
6	175.09	1.00
7	20.00	1.00
8	290.00	2.00
9	290.00	.89
10	290.00	1.11
11	87.42	.89
12	200.00	2.00
13	50.00	2.00
14	140.00	2.00
15	200.00	2.00
16	100.00	3.00
17	100.00	3.00
18	100.00	.00
19	134.97	3.00
20	134.97	3.00
21	700.00	3.00

HEAT EXCHANGER	UNIT NO.	HEAT LOAD	U	AREA
1	7	120.00	1.00	1.00
2	8	104.91	1.00	1.00
3	9	180.00	1.00	2.22

UTILITY EXCHANGER	UNIT NO.	HEAT LOAD
1	10	1695.09
2	11	-155.09

TOTAL UTILITY USAGE IS 1850.17

SPLITTER	UNIT NO.	SPLIT FRACTION
1	1	.385
2	3	.444
3	5	1.000

CONSTRAINT VALUES (C) (SHOULD ALL BE +VE
EXCEPT CC(12) WHICH SHOULD BE IN RANGE 0.000 TO .150)

```
XXXXX   NEXT VALUE IS INFEASIBLE  XXXXX
NO.12  (UTILITY USAGE)                  C = -.275E-02 ( 15.3 P.C.> MIN )
NO. 1  (HEAT LOAD,UNIT NO.  7 )         C =  .120E+01 ( HT.LOAD = 120.0)
NO. 2  (HEAT LOAD,UNIT NO.  8 )         C =  .105E+01 ( HT.LOAD = 104.9)
NO. 3  (HEAT LOAD,UNIT NO.  9 )         C =  .130E+01 ( HT.LOAD = 180.0)
NO. 4  (UNIT  7  HOT END DELT)          C =  .198E+01 ( 200.0  DEGREES )
NO. 5  (UNIT  7  COLD END DELT)         C =  .623E+00 (  64.8  DEGREES )
NO. 6  (UNIT  8  HOT END DELT)          C =  .263E+01 ( 265.0  DEGREES )
NO. 7  (UNIT  8  COLD END DELT)         C =  .251E+00 (  27.6  DEGREES )
NO. 8  (UNIT  9  HOT END DELT)          C =  .148E+01 ( 150.0  DEGREES )
NO. 9  (UNIT  9  COLD END DELT)         C =  .349E+00 (  37.4  DEGREES )
```

SPLIT FRACTIONS USED AS DEGREES OF FREEDOM

UNIT NO.	SPLIT FRACTION
5	1.0000

CALCULATED SPLIT FRACTIONS

UNIT NO.	SPLIT FRACTION
1	.3851
3	.4443

** CURRENT VALUES ARE INFEASIBLE

XXXXX CURRENT POINT INFEASIBLE XXXXX

PROGRAM ENDS AT CORNERPOINT NO. 3 (12)

APPENDIX D

Use of Program HERON1 for Constraint Maximization

To demonstrate the use of program HERON1 for constraint maximization the network used is that given in Figure 6.14 for which it was found that infeasible operation occurred not at a cornerpoint but at an intermediate point in the operating range. The purpose of this example is to demonstrate the way in which the program detects the set of intermediate operating conditions which produce the most infeasible operation.

The network used is given in Figure D.1 together with the input data required by the program. For details on the form of the input data see Chapter 5 and the listing of HERON1 in Appendix G.

The output from the program for this problem follows with points to note indicated (n). These are:

(1) The program has been run in interactive mode and the starting point for the optimization has been chosen as the mid-point between the lower bound and the upper bound of the input variables (in this case only one inlet is variable).

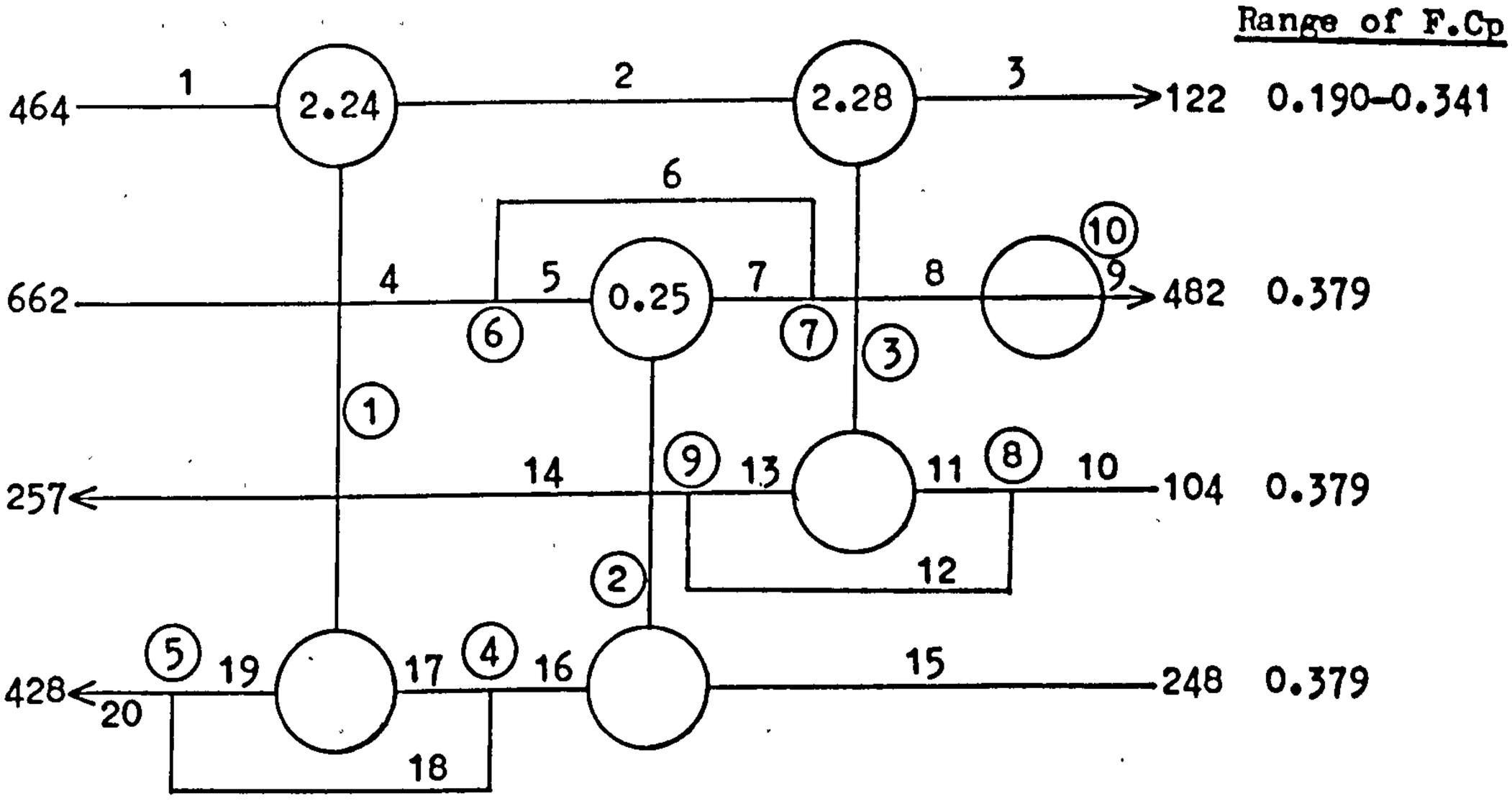
(2) A low level of output has been selected in which only the constraint function value being studied and infeasible constraint function values are reported. It can be seen immediately that although the constraint function value investigated, the heat load of the exchanger unit number 1, is feasible the temperature approaches associated with exchanger 1 are negative, indicating a structural infeasibility.

(3) Routine EØ4JAF is used to minimize the constraint function relating to the heat load of exchanger 1 (as in Appendix C, in practice a minimization is used although the mathematical development in Chapter 4 assumed maximization). The heat load must always be greater than zero.

(4) The routine finds that the minimum heat load is 7.0 therefore the heat load takes positive values throughout the operating range. The inlet conditions which produce the minimum heat load correspond to the F.Cp value of process stream 1 taking its lower limit.
The utility used is indicated as being infeasible because of rounding errors.

(5) Each network constraint function must be investigated and the next run investigates constraint number 4, the cold end temperature approach of exchanger 1. Again the mid-point of the operating range is used as a starting point.

(6) The minimum (most infeasible) value of the cold end temperature approach of exchanger 1 found is -1.5, which indicates a structural infeasibility. The value of the F.Cp of stream 1 producing this value is between the upper and lower limits. Referring to Figure 6.15, it can be seen that the program has correctly identified the inlet conditions which give rise to the most infeasible value of the constraint function relating to the cold end temperature approach of exchanger 1.




```
10,20
1,1,17,2,19
2,5,15,7,16
3,2,11,3,13
4,16,0,17,18
5,18,19,20,0
6,4,0,5,6
7,6,7,8,0
8,10,0,11,12
9,12,13,14,0
10,8,0,9,0
4
1,.190,.341,464,464
4,.379,.379,662,662
10,.379,.379,104,104
15,.379,.379,248,248
122,482,257,428
0,0
1,1,1
2.24,0.25,2.28
1,1
```

Figure D.1 The network used in the example and the input data required by the program.

```
RUN NUMBER    1 STARTS
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CURRENT INLET CONDITIONS

INLET NO.  VARIABLE  LOWER BOUND  UPPER BOUND  ACTUAL VALUE
     1        FCP         .19          .34          .27
     1         T       464.00       464.00       464.00
     2        FCP         .38          .38          .38
     2         T       662.00       662.00       662.00
     3        FCP         .38          .38          .38
     3         T       104.00       104.00       104.00
     4        FCP         .38          .38          .38
     4         T       248.00       248.00       248.00

MINIMIZED AND INFEASIBLE CONSTRAINTS
CONSTRAINT BEING MINIMIZED IS NO.  2
NO. 1  (UTILITY USAGE)                C =  .112E-09 (  -.0 P.C.> MIN )
NO. 2  (HEAT LOAD,UNIT NO.  1 )       C =  .328E+00 ( HT.LOAD =   32.8)
NO. 3  (UNIT  1  HOT END DELT)        C = -.611E+00 ( -61.1  DEGREES )
NO. 4  (UNIT  1  COLD END DELT)       C = -.101E-01 (  -1.0  DEGREES )

***** INFEASIBLE POINT FOUND *****
CALLING OPTIMIZING ROUTINE E04JAF FROM NAG8F
EXITING FROM OPTIMIZING ROUTINE

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

A MINIMUM HAS BEEN FOUND

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

CURRENT INLET CONDITIONS

INLET NO.  VARIABLE  LOWER BOUND  UPPER BOUND  ACTUAL VALUE
     1        FCP         .19          .34          .19
     1         T       464.00       464.00       464.00
     2        FCP         .38          .38          .38
     2         T       662.00       662.00       662.00
     3        FCP         .38          .38          .38
     3         T       104.00       104.00       104.00
     4        FCP         .38          .38          .38
     4         T       248.00       248.00       248.00

MINIMIZED AND INFEASIBLE CONSTRAINTS
CONSTRAINT BEING MINIMIZED IS NO.  2
NO. 1  (UTILITY USAGE)                C =  .244E-12 (  -.0 P.C.> MIN )
NO. 2  (HEAT LOAD,UNIT NO.  1 )       C =  .699E-01 ( HT.LOAD =    7.0)

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
RUN NUMBER    2 STARTS
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CURRENT INLET CONDITIONS

INLET NO.  VARIABLE  LOWER BOUND  UPPER BOUND  ACTUAL VALUE
     1        FCP         .19          .34          .27
     1         T       464.00       464.00       464.00
     2        FCP         .38          .38          .38
     2         T       662.00       662.00       662.00
     3        FCP         .38          .38          .38
     3         T       104.00       104.00       104.00
     4        FCP         .38          .36          .38
     4         T       248.00       248.00       248.00

MINIMIZED AND INFEASIBLE CONSTRAINTS
CONSTRAINT BEING MINIMIZED IS NO.  4
NO. 1  (UTILITY USAGE)                C =  .223E-08 (  -.0 P.C.> MIN )
NO. 3  (UNIT  1  HOT END DELT)        C = -.611E+00 ( -61.1  DEGREES )
NO. 4  (UNIT  1  COLD END DELT)       C = -.101E-01 (  -1.0  DEGREES )

CALLING OPTIMIZING ROUTINE E04JAF FROM NAG8F
EXITING FROM OPTIMIZING ROUTINE
```

(1) (2) (3) (4) (5)

++

PROBABLE THAT A MINIMUM HAS BEEN FOUND

++

CURRENT INLET CONDITIONS

INLET NO.	VARIABLE	LOWER BOUND	UPPER BOUND	ACTUAL VALUE
1	FCP	.19	.34	.25
1	T	464.00	464.00	464.00
2	FCP	.38	.38	.38
2	T	662.00	662.00	662.00
3	FCP	.38	.38	.38
3	T	104.00	104.00	104.00
4	FCP	.38	.38	.38
4	T	248.00	248.00	248.00

(6)

MINIMIZED AND INFEASIBLE CONSTRAINTS
CONSTRAINT BEING MINIMIZED IS NO. 4
NO. 1 (UTILITY USAGE) C = -.398E-11 (.0 P.C.> MIN)
NO. 3 (UNIT 1 HOT END DELT) C = -.455E+00 (-45.5 DEGREES)
NO. 4 (UNIT 1 COLD END DELT) C = -.150E-01 (-1.5 DEGREES)

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

PROGRAM ENDS

APPENDIX E

A reprint of the paper "The Design of Flexible Heat Exchanger Networks" based on a summary of part of the work presented in this thesis follows.

The paper was presented at the first UK National Heat Transfer Conference held in Leeds on July 3-5, 1984, The proceedings of which appear in I.Chem.E. Symposium Series No. 86.

THE DESIGN OF FLEXIBLE HEAT EXCHANGER NETWORKS

A.C. Beautyman* and A.R.H. Cornish*

A method is presented for the design of heat exchanger networks which enables a range of operating conditions to be accommodated by the network with high energy recovery. The method, which is based on computer simulation and optimization techniques, is applied to a number of pinched and unpinched network designs. The importance of correct placement of bypasses in order to ensure good control characteristics and high energy recovery is demonstrated. Finally, an extension to the design method is presented which enables it to include cases in which some of the operational variables are interdependent.

INTRODUCTION

In recent years rising energy costs have forced plant designers to strive for greater recovery of waste heat. Linnhoff and coworkers (8,11,12,13) have provided design methods for heat exchanger networks which for fixed operating conditions strike an economic balance between high energy recovery and capital cost. Increased integration, however, can severely restrict the flexibility of the integrated plant; that is its ability to handle a range of operating conditions in an economical way.

It is relatively easy to introduce some flexibility into a network by arbitrarily oversizing heat exchangers, for example by applying a fixed overdesign factor of 10%, and indeed this seems to be how flexible designs were achieved in the past. In the current restrictive economic climate more exact design methods are needed for quantitative overdesign leading to low capital and operating costs. This work provides a practical approach to the solution of this problem.

PROBLEM STATEMENT

The problem is formulated as the design of a heat exchanger network which is to be able to handle a range of inlet temperatures and flowrates for which only upper and lower limits are known. This represents a practical way of treating unpredictable inlet variations which arise from feedstock or product changes or from the need to operate at less than maximum throughput. These variables would be impossible to quantify accurately in advance except in the most general way. The design method presented can be easily modified to handle other uncertainties such as target temperature or heat transfer coefficient variations.

*Department of Chemical Engineering and Chemical Technology, Imperial College, London

The following example illustrates a flexible design problem statement:

Stream	F.Cp	Ti	Tt	Type
1	1.2 - 1.4	500 - 450	100	hot
2	1.6 - 1.9	450 - 300	150	hot
3	0.4 - 0.9	100 - 110	400	cold

where F.Cp = The product of flowrate and specific heat capacity (kW/K)
Ti = Inlet temperature (°C)
Tt = Target temperature (°C)

e.g. The F.Cp value of stream 1 can take any value between 1.2 and 1.4.

The combination of the ranges of F.Cp and Ti defines the 'disturbance range' of uncertain inlet conditions which contains all possible sets of inlet conditions. Extreme points of the disturbance range are termed 'cornerpoints'.

The objective of this work is to design a network which can handle the entire disturbance range in an economical way using as a starting point any given method for the design of a network of sized exchangers for any single point in the disturbance range (the standard 'non-flexible' synthesis problem).

DESIGN APPROACH

Feasible Operation

The final designed network must be able to operate at any possible combination of inlet conditions. As there are an infinite number of such sets of conditions direct checking is not possible. Marselle and coworkers (1,2,3,4) suggested that if a design is feasible for four selected cornerpoints of the disturbance range then that design would probably be feasible for the entire range. However, this has been found to be rarely the case and Saboo and Morari (9) provided examples where feasibility of all the cornerpoints does not guarantee operation at intermediate points.

A more rigorous approach is needed.

If the following vectors are defined:

d = equipment design variables (e.g. exchanger areas)
x = state variables (e.g. temperatures, regulatory split fractions used for meeting target temperatures)
z = control variables (e.g. optimizing split fractions used for meeting energy and operating targets)
θ = uncertain variables (inlet temperatures, heat capacity flowrates)

D is the disturbance range of all uncertain inlet variables (θ) which contains all possible sets of values of θ.

The equations governing a network are:

$$h'(d,z,x,\theta) = 0$$

To ensure that in practice a network can operate feasibly certain constraints must be satisfied:

i) All split fractions must be less than 1.0, greater than 0.0.
ii) All exchanger temperature approaches must be greater than a small positive value.
iii) All heat exchanger heat loads must be positive.

If g' is the set of all (J) such constraints then for feasible operation

$$g'(d,z,x,\theta) \leqslant 0$$

or as $x = x(d,z,\theta)$ the state variables can be eliminated.

~~$h(d,z,\theta) = 0$~~

$$\text{s.t. } g(d,z,\theta) \leqslant 0$$

To ensure feasibility throughout the disturbance range the following must apply

$$\max_{\theta \in D} \min_{z} \max_{j \in J} g_j(d,z,\theta) \leqslant 0$$

In general terms the "worst" value of all the individual constraint values which can be found in the disturbance range must be feasible. The control variables z are used to try to maintain feasibility.

For networks with zero degrees of freedom, DF = 0, no optimizing control variables z are present. The problem then reduces to:

$$\max_{\theta \in D} \max_{j \in J} g_j(d,\theta) \leqslant 0$$

So that, as shown by Grossmann and coworkers (6,7), the constraints may be treated individually.

$$\max_{\theta^J \in D} g_j(d,\theta^j) \leqslant 0 \text{ for all } j \in J$$

In terms of a search procedure:

For each constraint in turn (j = 1,J)

a) Check that the "worst" possible value in the disturbance range is feasible.

$$\text{Is } \max_{\theta^J \in D} g_j(d,\theta^j) \leqslant 0?$$

b) As a secondary, but not essential, check ensure that all other constraints at this point are feasible.

$$\text{Are } g_i(d,\theta^j) \leqslant 0 \text{ for all i except } i = j$$

If additional degrees of freedom are required, DF > 0, the procedure is much more difficult to carry out as it is not possible to check the constraints individually, see Grossmann and coworkers (6,7).

In the work of Grossmann and coworkers (5,6,7) the design variables d are manipulated in an optimization problem which is subject to these feasibility constraints, in order to find the minimum of a cost function. This can lead to severe computational problems in any implementation. However, such a strict optimum is not necessary for network design.

Energy Targets

The techniques of Grossmann could be applied to find the cost optimum design if the computational difficulties could be overcome. To be effective Grossmann's approach would require accurate future economic data and because of the highly uncertain nature of the problem would probably lead to needlessly overexact solutions. The work of Linnhoff and coworkers does, however, provide a useful method for setting economic targets. Linnhoff and coworkers (8,11,12, 13) reported that because of the relatively high cost of energy compared with that of equipment, a network which recovers a high proportion of the recoverable energy will be very close to the 'theoretical' minimum cost network. This led to the development of a successful non-flexible synthesis method which does not guarantee an optimal solution but does produce a good solution.

As energy prices are likely to remain high in the future this goal of 'high energy recovery' is a good one to adapt for flexible design purposes. To implement this an extra constraint can be imposed on the network to ensure high energy recovery throughout the disturbance range.

For an unpinched problem this may be the requirement that all possible energy must be recovered for all points in the disturbance range.

For a pinched problem the requirement could take the form that the amount of energy used at any point in the disturbance range should be less than some specified multiple of the minimum theoretical energy use, Umin, at the same point in the disturbance range calculated for the minimum physically reasonable value of the allowable approach temperature ΔT_{min}.

The energy constraint could be:

$$\frac{U_{actual}}{U_{min}} < 1.2 \text{ applied throughout the range.}$$

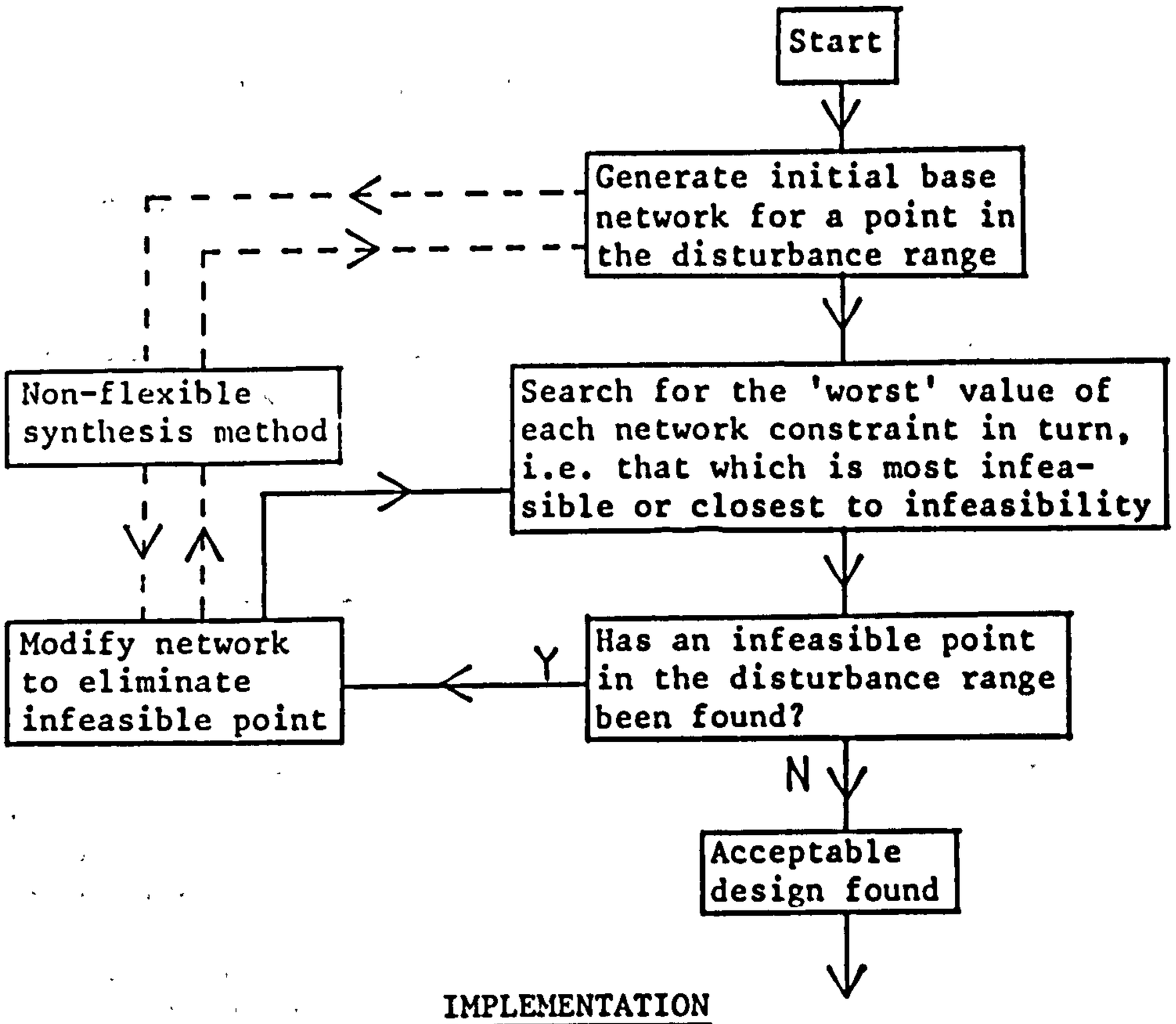
There are many possible ways of formulating an energy target constraint.

This constraint can then be included with and treated in the same way as the other feasibility constraints.

Design Scheme

The following outline design scheme can now be applied. Note that a range of initial networks can be considered and this may lead to a number of acceptable fully feasible designs for the same problem. The final design can then be chosen on the basis of capital cost, control and dynamic behaviour, etc.

This method has the immediate advantage of ensuring feasible operation and high energy recovery for <u>all</u> points in the disturbance range, not just cornerpoints.



IMPLEMENTATION

This design method has been implemented in a computer program for the case in which the number of degrees of freedom is zero, DF = 0.

A simulation program models the behaviour of any given arrangement of streams, sized heat exchangers, utility exchangers, bypasses, stream splits and mixers. Pure counter-current flow, single shell pass and single tube pass exchangers are assumed although the method will allow for any exchanger model. The equation solving routine used is based on a modified form of Broyden's method.

Inlet flowrate and temperature ranges are supplied and these become the manipulated variables in minimization routine. The routine used is the NAG Library algorithm EØ4JBF which is a quasi-Newton procedure for minimizing a function subject to fixed lower and upper bounds on the variables. This is used to maximize each network constraint in turn as described previously and thus values of inlet flowrate and temperature which cannot be handled feasibly by the network under consideration can be found.

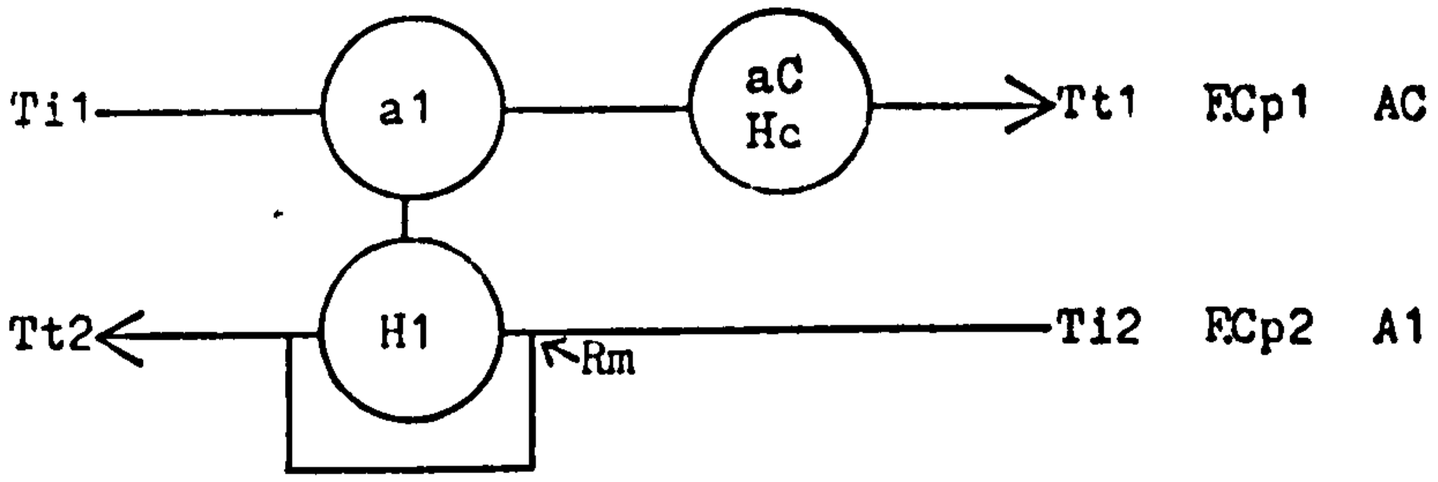
To ensure efficient equation solving and the detection of infeasible points the expression ΔTlogmean(ΔT_{LM}) has been modified to allow for negative exchanger temperature approaches. This is obviously not possible in a real network but is necessary to enable a solution to the describing equations to be found at infeasible points in the disturbance range. Similarly split fractions outside the feasible range of 0.0 to 1.0 are allowed.

The way in which this program can be used to detect sizing and structural infeasibilities is best and most simply illustrated by considering a flexible design example.

EXAMPLE PROBLEM 1

As the problem is unpinched the energy target set is that maximum energy must be recovered for all points in the disturbance range.

The notation is as follows:



Tin - Inlet temperature (°C) of stream n
Ttn - Target temperature (°C) of stream n
FCpn - Flow (F) x Specific heat capacity (Cp) of stream n (kW/K)
Hm - Heat load of exchanger m (numbered from left to right) (kW)
am - The overall heat transfer coefficient, U, times the required area, A, of exchanger m (kW/K) calculated
 i) At the set of inlet conditions currently being considered
 ii) Assuming all exchanger split fractions, Rm, are 1.0 to ensure the minimum required area is calculated.
Am - The largest required value of U.A for exchanger m which has been found in the search so far.
Rm - The exchanger split fraction, i.e. the fraction of the total flow which passes through exchanger m.
Subscript C refers to cooling utility exchangers (H for hot).

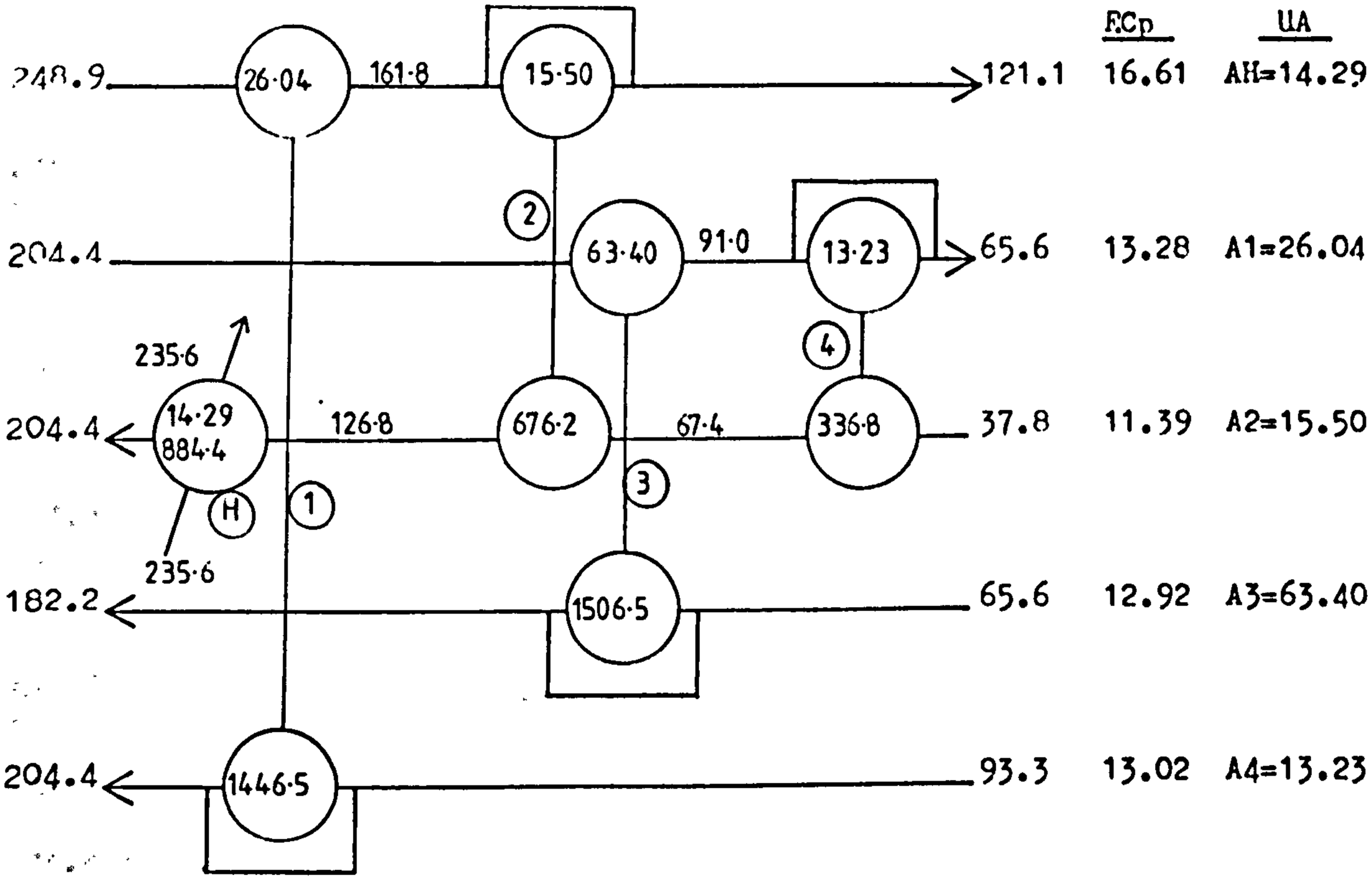
It is assumed that the overall heat transfer coefficient, U, is constant and therefore a larger value of am (and hence Am) corresponds to a larger exchanger area being required.

Heat capacity values are also assumed to be constant and therefore a range of inlet F.Cp values is produced by inlet flow variations only.

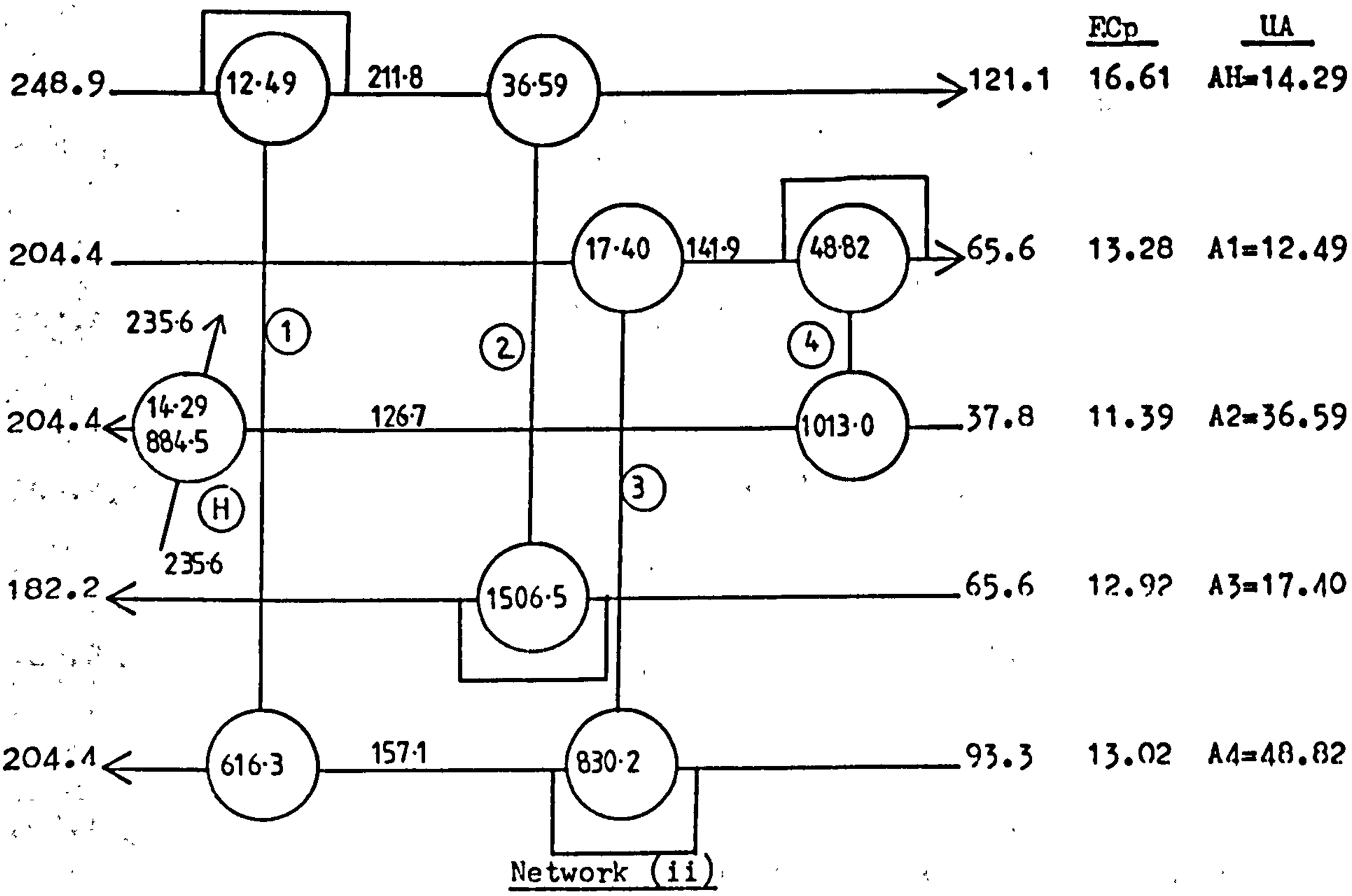
The networks in Fig.1 are two solutions to the literature problem 5SP1 given by Linnhoff (8,12). Bypasses have been assigned according to rules developed in a previous report (10) in order to ensure that the number of degrees of freedom is zero, which is an essential condition for regulatory control (the meeting of target temperatures). Briefly, one control in the form of a utility exchanger or bypass must be provided for each of the five target temperatures.

A disturbance range is formed by allowing all F.Cp values to vary independently by up to -10% of the nominal values indicated.

The constraint maximization procedure is applied to the two networks and a critical point in the disturbance range is found corresponding in network (i) to the most infeasible value for the split fraction on exchanger no. 2. Network (ii) is also checked at this point. The performance of the networks at this point is shown in Fig.2 .

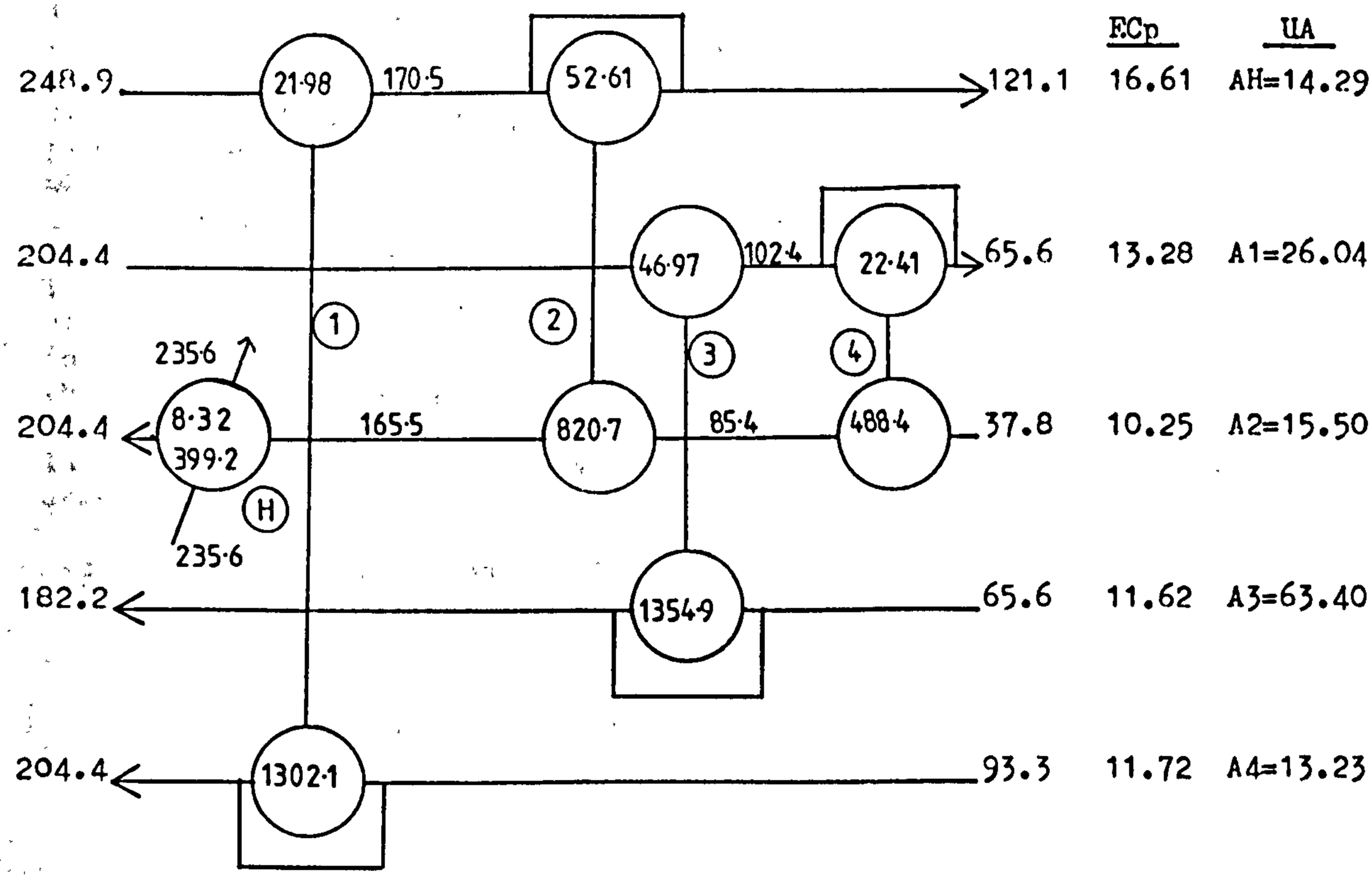


Network (i)

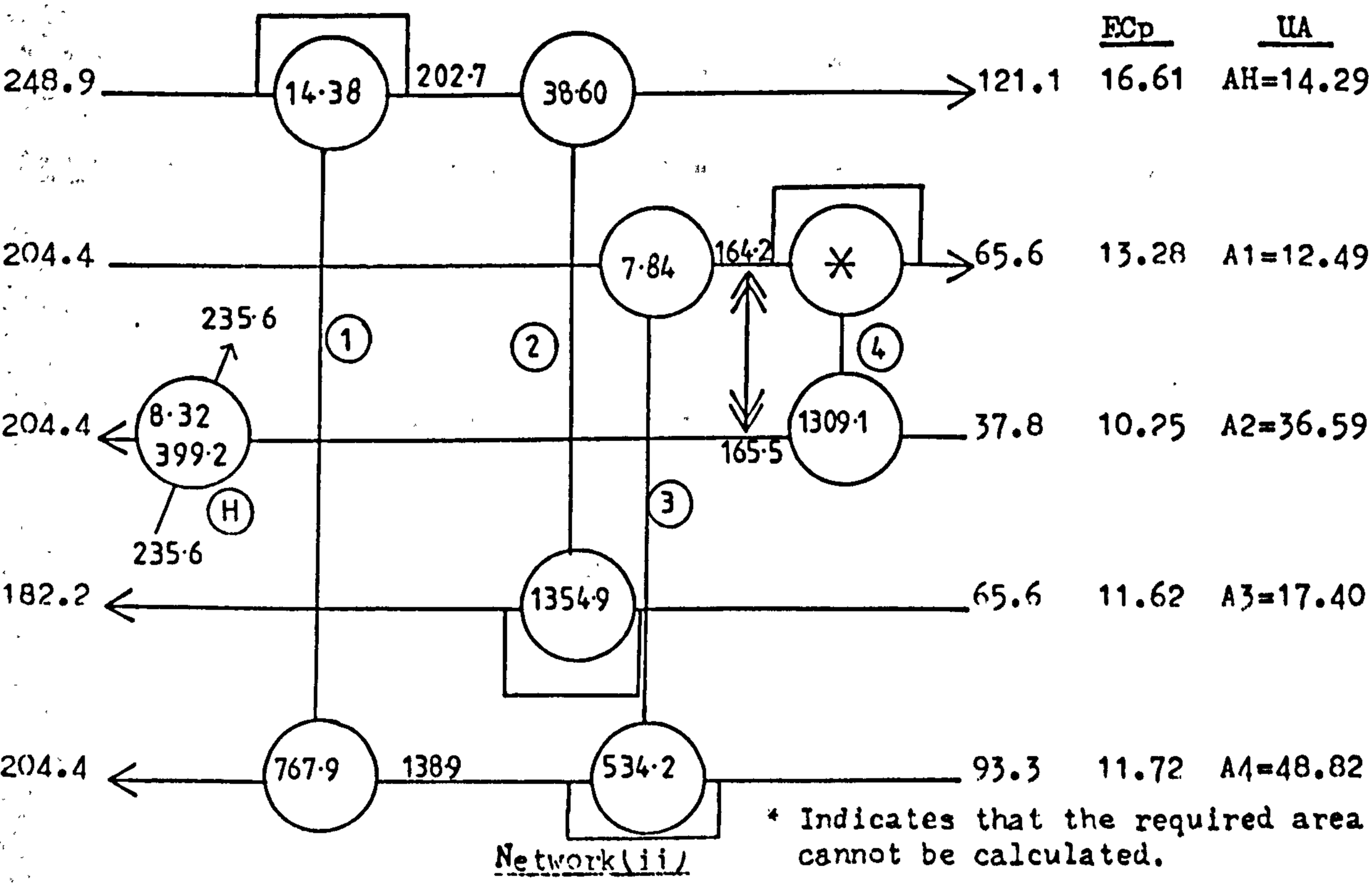


Network (ii)

Fig.1 Two nominal condition solutions for problem 5SP1



Network (i)



* Indicates that the required area cannot be calculated.

Network (ii)

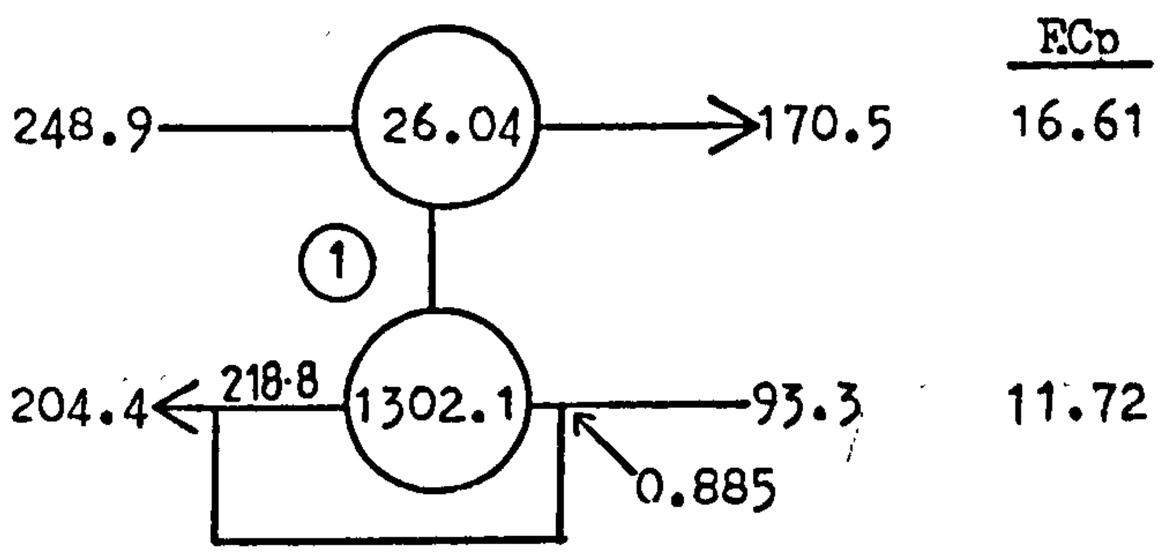
Fig.2 The performance of the networks at an infeasible point

In network (i) it can be seen that the required U.A for exchanger 1 (21.98) is less than the installed U.A (26.04). In practice and in the simulation this results in an exchanger split fraction on exchanger 1 of <1.0. Exchanger 1 can handle this particular set of inlet conditions feasibly.

In the case of exchanger 2, however, the installed area calculated from the previous set of 'nominal' inlet conditions resulted in a U.A value of 15.50, whereas the required U.A for this set of inlet values is 52.61. The exchanger is undersized for this duty and the area must be increased to give U.A = 52.61 in order to allow exchanger 2 to handle this set of inlet conditions. Similarly, it can be seen that the area of exchanger 4 must also be increased. This set of inlet conditions has revealed sizing infeasibilities for exchangers 2 and 4 in network (i). These can be removed by increasing the relevant U.A values in the network model and resuming feasibility checking.

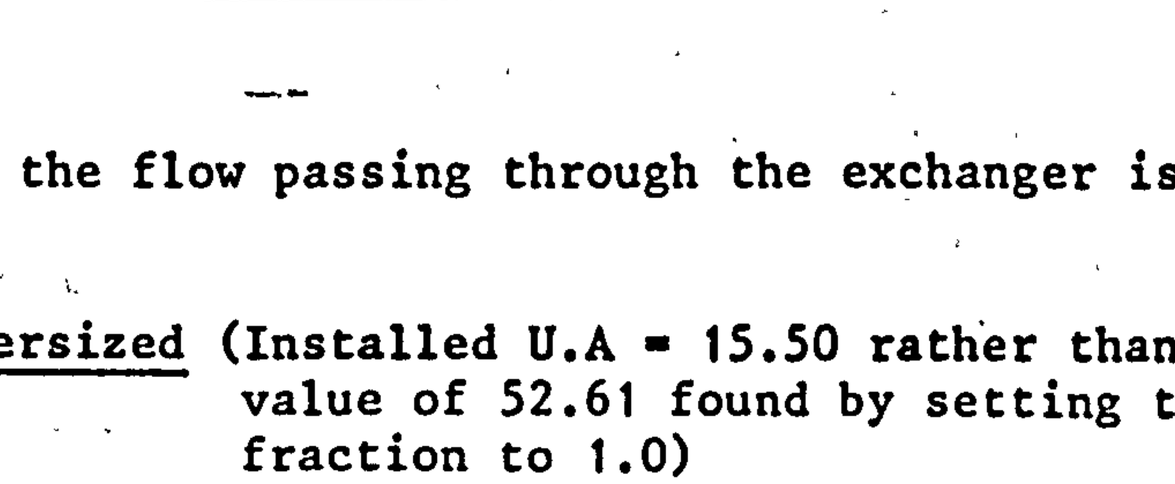
The results in the simulation of oversized and undersized exchangers can be demonstrated as follows.

Exchanger 1 oversized (Installed U.A = 26.04 rather than the minimum required value of 21.98)



The fraction of the flow passing through the exchanger is 0.885, a feasible value.

Exchanger 2 undersized (Installed U.A = 15.50 rather than the minimum required value of 52.61 found by setting the exchanger split fraction to 1.0)



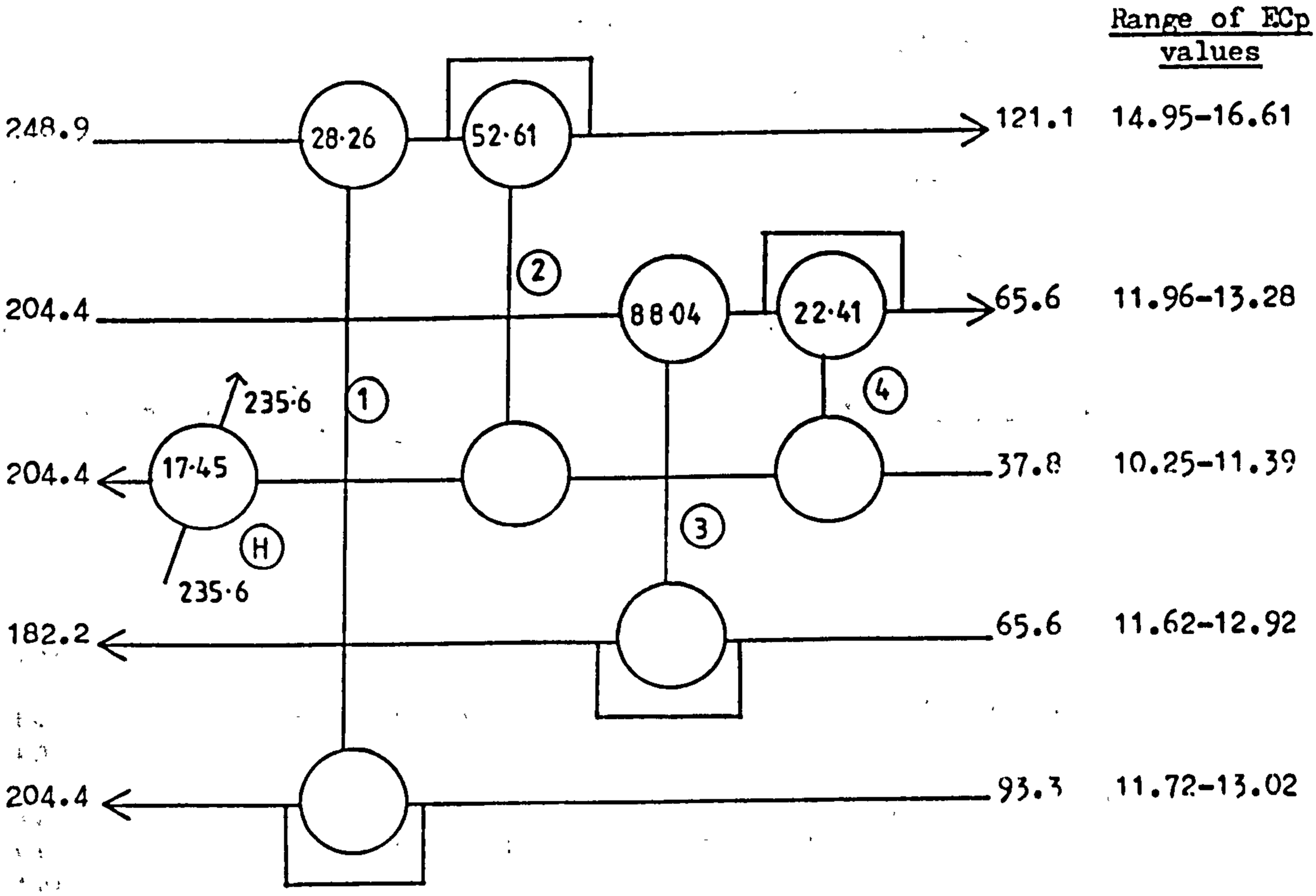
The required fraction of the flow passing through the exchanger is -0.428, obviously not a feasible value although mathematically all the energy balances are correct. The split fraction value of -0.428 is the most infeasible value of this constraint which can be found in the disturbance range.

For the same set of inlet conditions network (ii) shows a structural infeasibility. Exchanger 4 cannot be sized so as to handle this particular set of conditions due to the negative ΔT value indicated. The simulation program found a solution for this point using the modified ΔT_{LM} formulation but it is not a physically possible solution. In practice this would not occur and the infeasibility would show as a number of target temperatures not being met. To remove this infeasibility the structure of the network would have to be changed by adding an extra exchanger. This would mean:

1) Optimizing control may be required
2) More than the minimum number of units being used therefore increasing the cost of the final network.

For these reasons network (ii) is eliminated from consideration until it is clear whether network (i) can be made fully feasible simply by oversizing.

In fact this is the case and the following is a fully feasible version of network (i)



All the exchangers have had to be oversized from the nominal network.

In a full investigation of this problem five alternative networks were studied for five different disturbance ranges of inlet temperature and F.Cp.

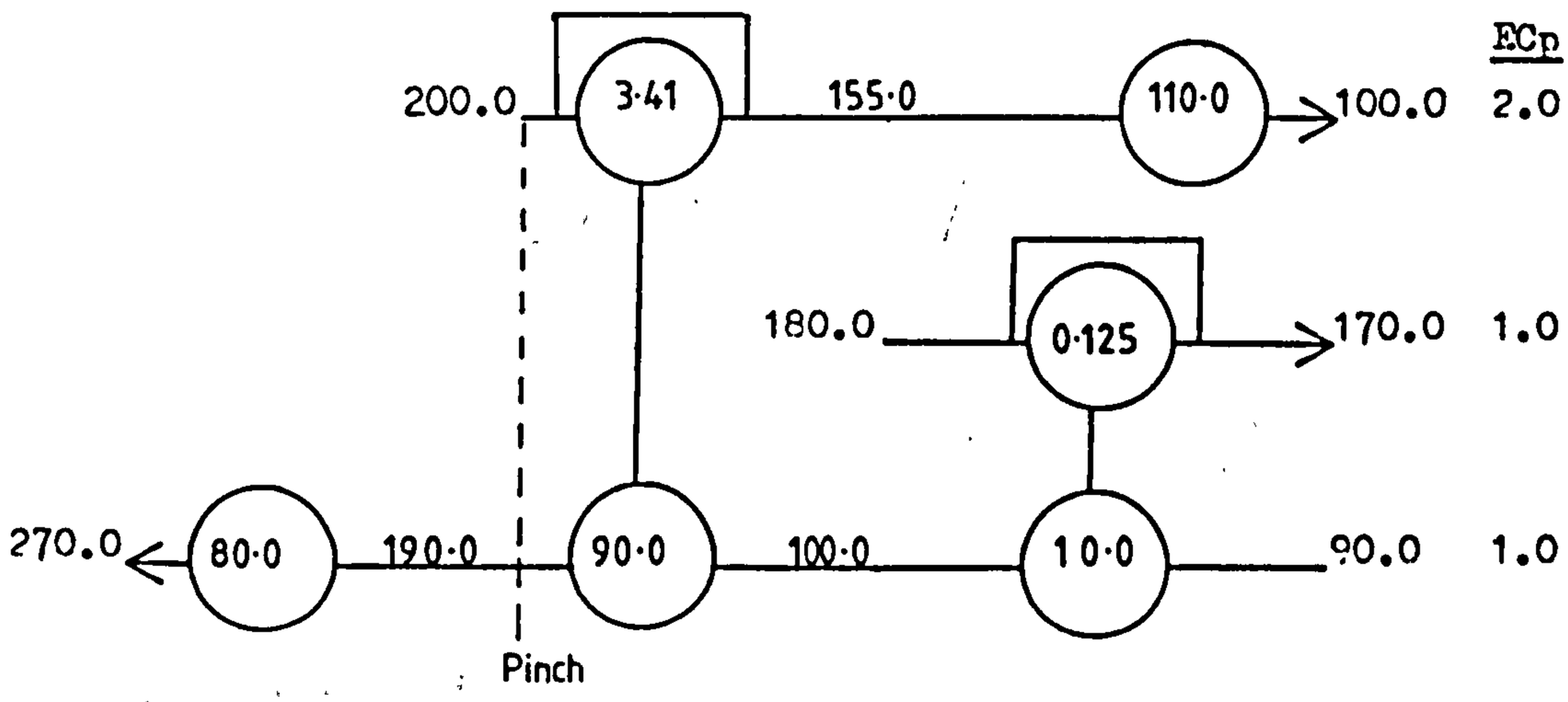
In some cases a number of alternative fully feasible networks (with oversizing only) were found and since the energy use is the same for all alternatives it was possible to select between them on the basis of capital cost only.

The cost of the final network was found to be very sensitive to the amount of flexibility required. Decisions about whether to attempt to lessen inlet disturbances by tighter design of the external process can be based on the relative costs of providing different amounts of flexibility.

For the case where both inlet temperatures and flowrates were allowed to vary it was found that, in terms of the computational effort involved, checking all 1024 cornerpoints of the disturbance range took more than twice the time taken to maximize all 17 network constraints, even though all the most infeasible points found were cornerpoints.

PINCHED PROBLEMS

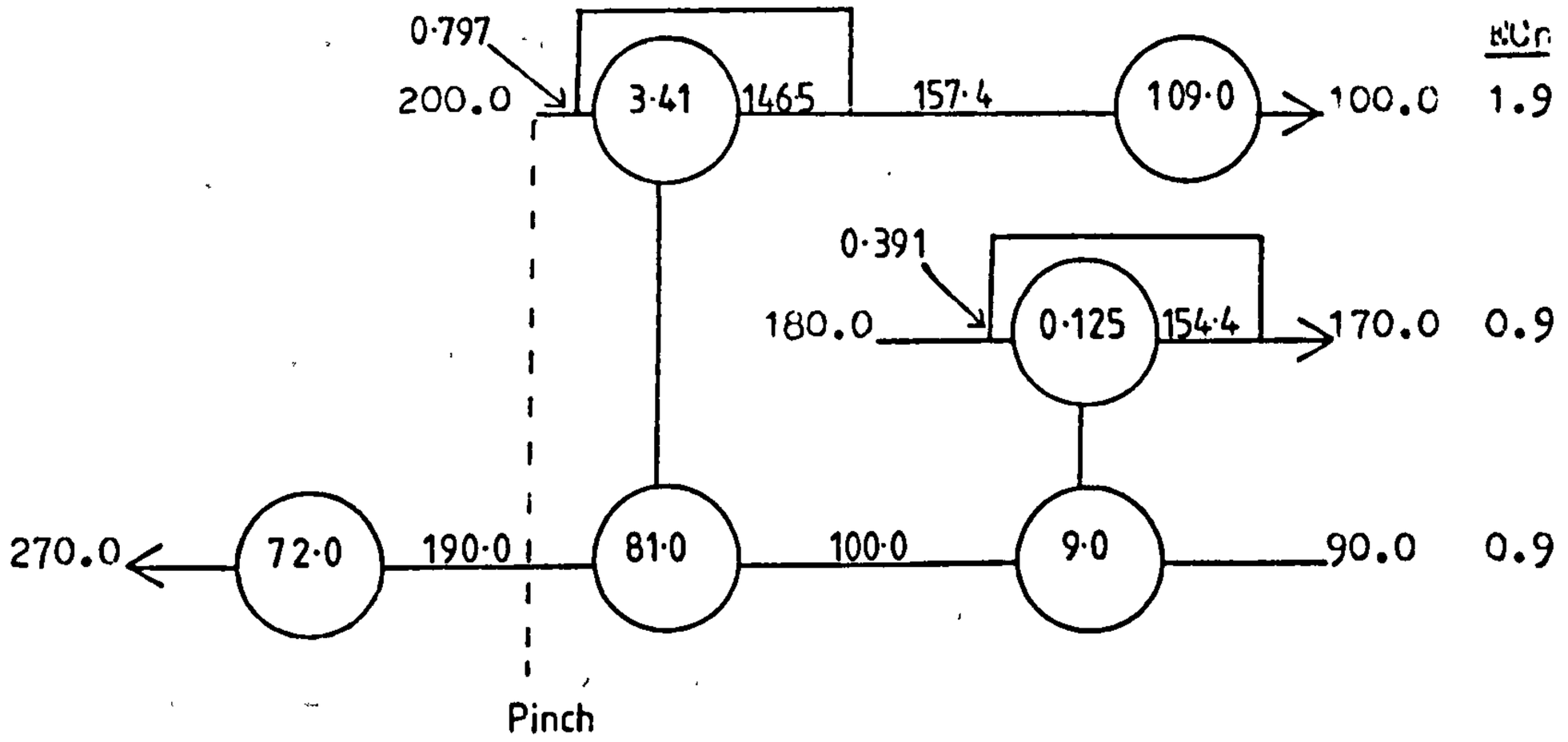
Marselle and Morari (1,2,3,4), in designing a feasible network structure, maintain a fixed ΔT_{min} throughout the disturbance range. This is a misleading energy recovery objective for flexible network design as can be demonstrated in the following example.



This network is designed for ΔT_{min} = 10 K.

Here DF = 1 therefore one bypass must be used for regulatory control, one for optimizing control.

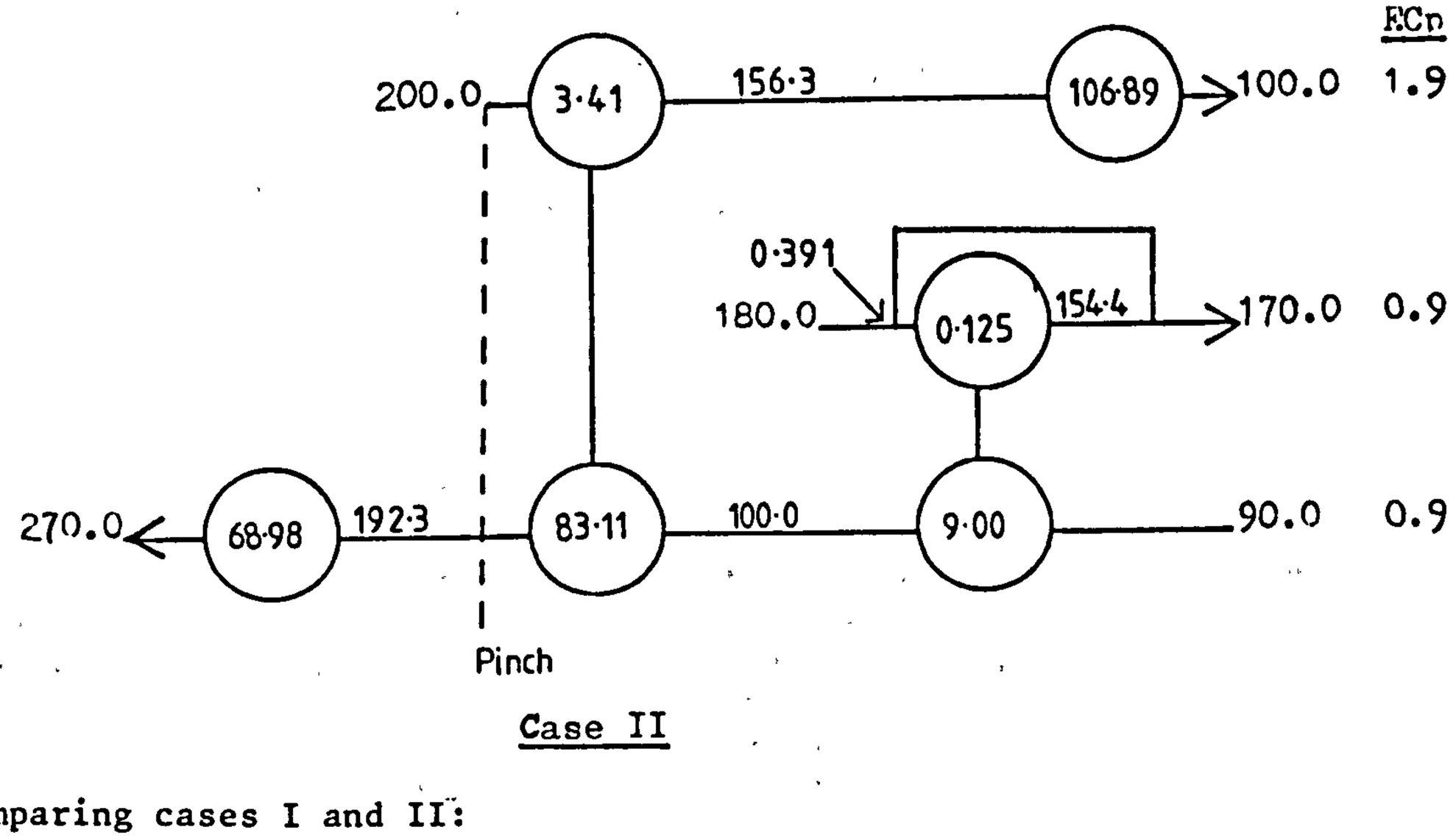
If a disturbance range is formed by allowing F.Cp values to vary by up to -0.1 of the nominal values above, then at another point in the disturbance range the following can apply.



Case I

ΔT_{min} = 10 K is maintained by setting both split fractions to values <1.0.

For exactly the same network and conditions the following is also possible:



Case II

Comparing cases I and II:

a) The use of optimizing control in case I to maintain ΔT_{min} = 10 K results in a total utility requirement of 181.0 kW. In case II no optimizing control is needed and utility use is <u>less</u>, 175.9 kW.

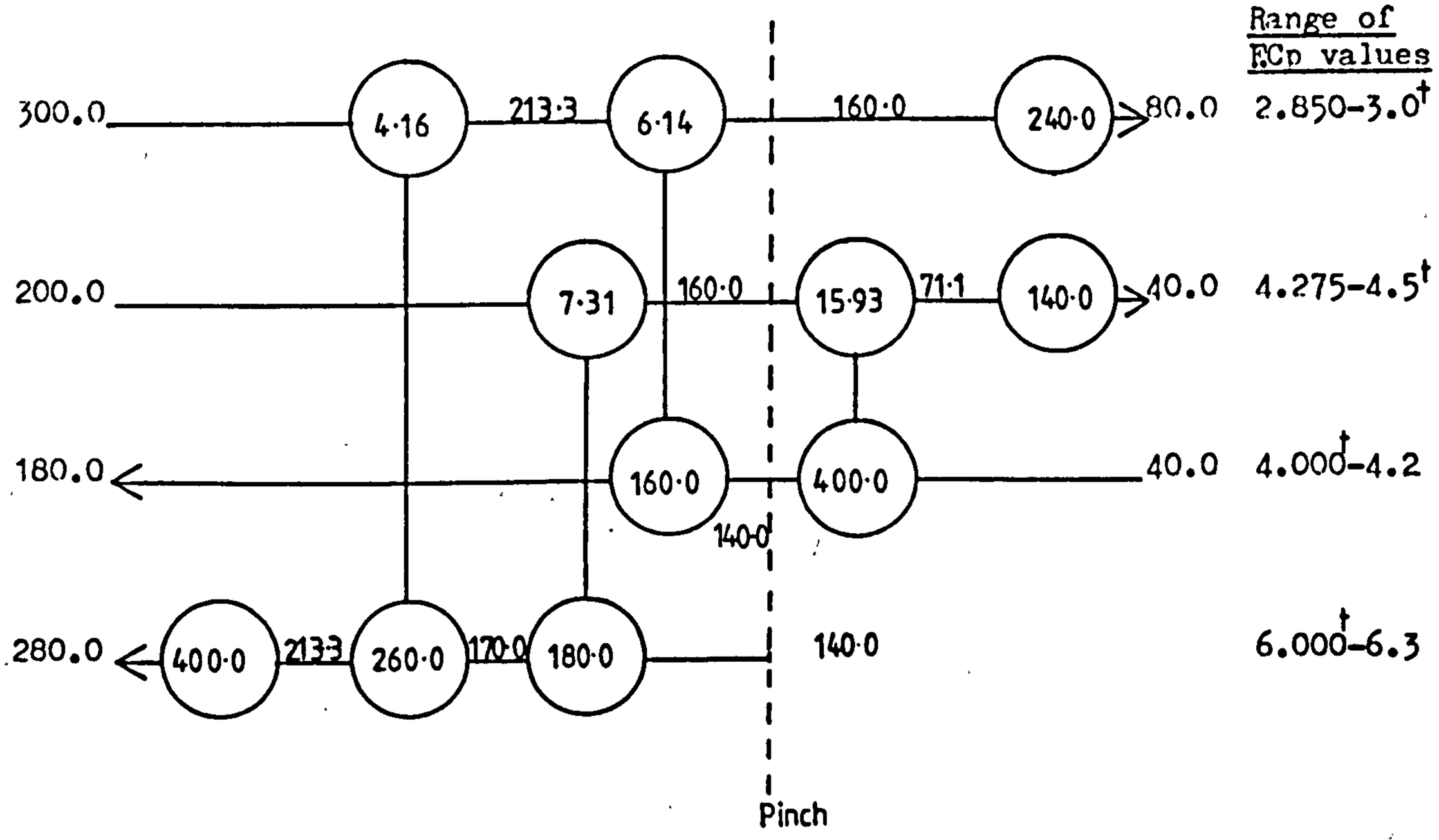
b) The ΔT_{min} value in case II is 7.7 K compared to 10K in case I. This does not mean that larger exchangers are required in case II, rather that advantage is being taken of exchangers which are already oversized for that particular duty.

A fixed ΔT_{min}, although useful for non-flexible designs, is not a satisfactory energy target for flexible designs. Marselle and Morari's implicit assumption that all exchangers must be bypassed is not valid, neither, therefore, is their neglect of actual exchanger areas.

These points are demonstrated in the design of a flexible pinched network.

EXAMPLE PROBLEM 2

The following network has been designed using Linnhoff and coworkers' procedures (8,11,12,13) assuming ΔT_{min}=20K. A disturbance range has been formed by allowing the F.Cp values to vary as indicated.

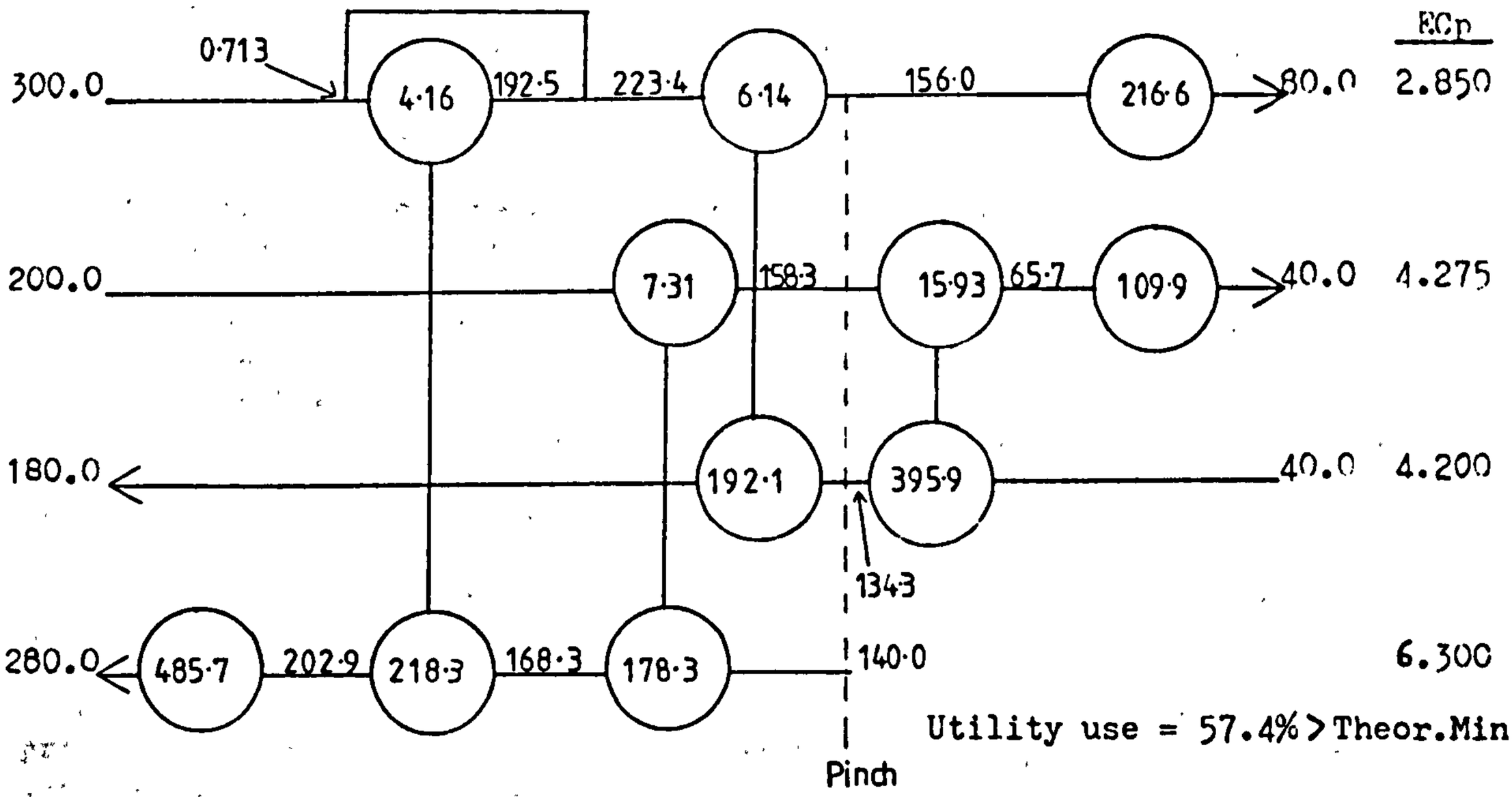


† = Nominal conditions for which this design applies.

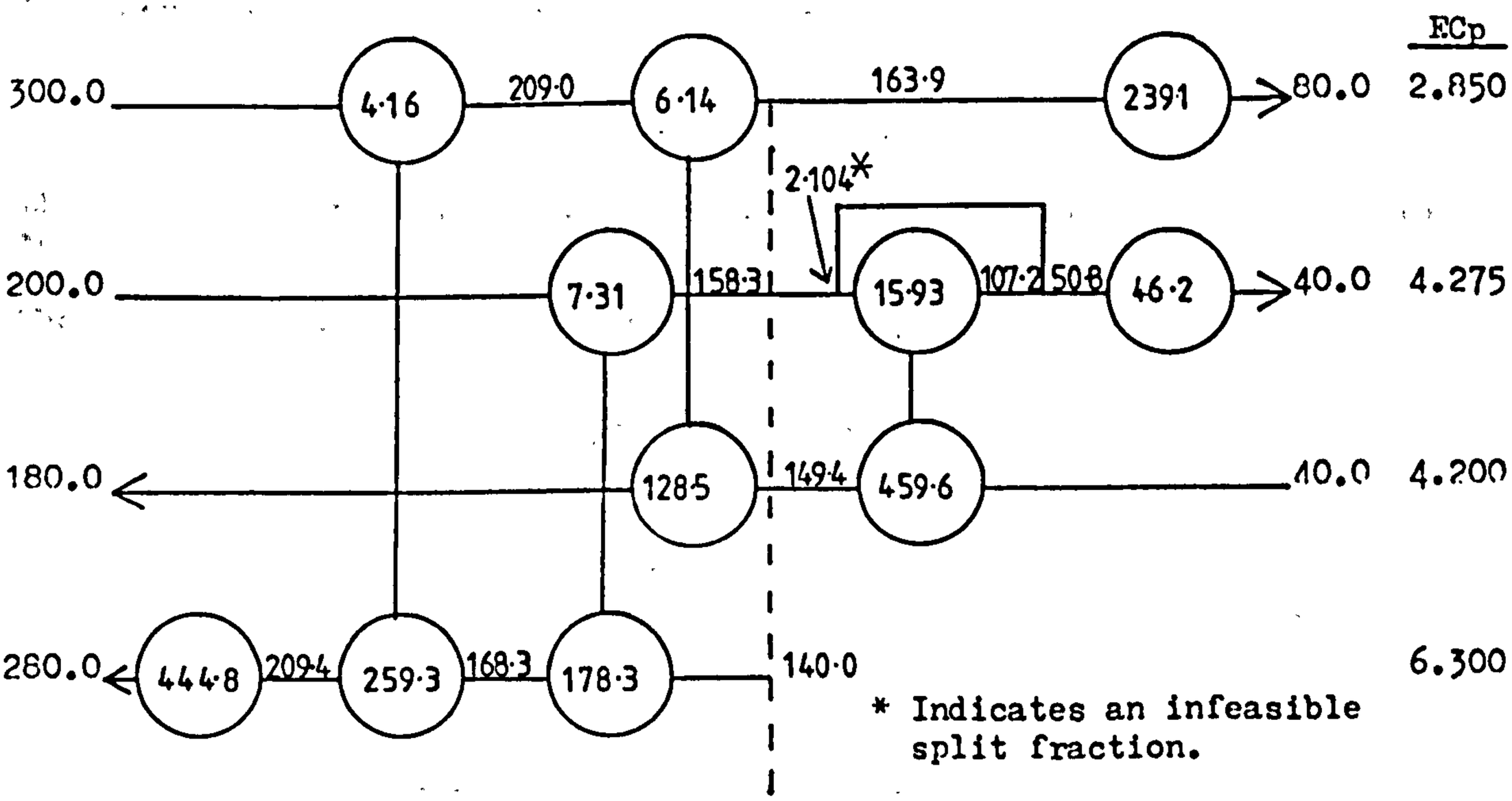
This network has an energy requirement that is 62.5% greater than theoretical minimum use calculated at ΔT_{min} = 0 K. Since this is an economical value for the nominal design (Linnhoff and coworkers(8,11,12,13)) this would seem to be a good energy target to set for the entire disturbance range: all points in the disturbance range must achieve an energy use of 62.5%> Theoretical Minimum or better. This energy target does not depend on any fixed ΔT_{min} value.

In the above design DF = -1 at present so at least one bypass must be assigned to allow for regulatory control. If all exchangers were bypassed three optimizing controls would result and obviously, therefore, there is a great incentive to provide regulatory control only.

There are four alternative bypass placements, two of which perform as follows at another point in the disturbance range. (Fig.3)



Network (a)



Network (b)

Fig.3 The performance of two otherwise identical networks with different bypass placements for the same inlet conditions.

Configuration (a) deals adequately with this set of operating conditions with a better percentage energy recovery than the nominal case. Configuration (b), however, which is identical to (a) apart from the bypass placement shows that if this configuration is used exchanger 4 is undersized for the required duty. Obviously, then, network (a) is preferred.

The other two alternative bypass placements, not shown here, are also infeasible at this point, one showing a size infeasibility, one a structural infeasibility.

Further investigation reveals that network (a), excactly as shown above, is feasible over the full disturbance range and, furthermore, it recovers the maximum possible energy for that particular set of exchanger areas for all points in the disturbance range (i.e. even if all exchangers are bypassed and optimizing control implemented with the objective of recovering maximum possible energy, the energy recovery figures of network (a) cannot be bettered).

This example clearly demonstrates the need to place bypasses correctly in a network to ensure flexible operation.

It can be shown that the choice of which exchangers to bypass cannot be made a priori by studying the network configuration or the pinch position, the disturbance range must be known.

NON-INDEPENDENT INLET VARIATIONS

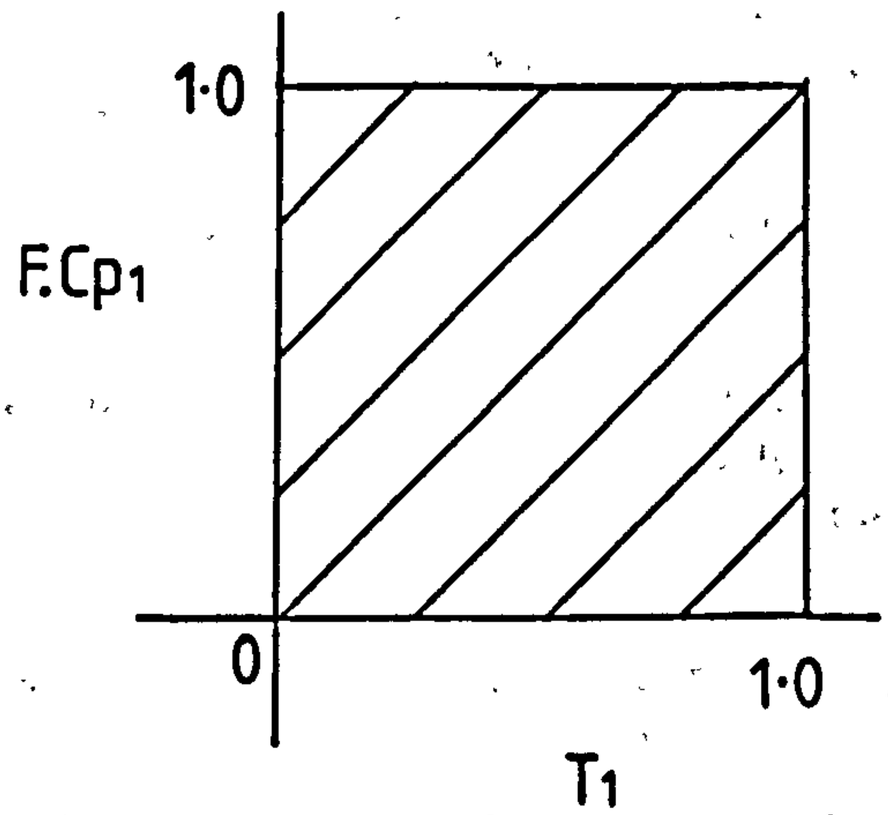
Consider a network which has only one variable inlet. The disturbance range is

$$0 \leqslant T1 \leqslant 1$$

$$0 \leqslant F.Cp1 \leqslant 1$$

where T1 and F.Cp1 are scaled values of the F.Cp and temperature inlet values for the variable stream.

The feasible space is as follows:

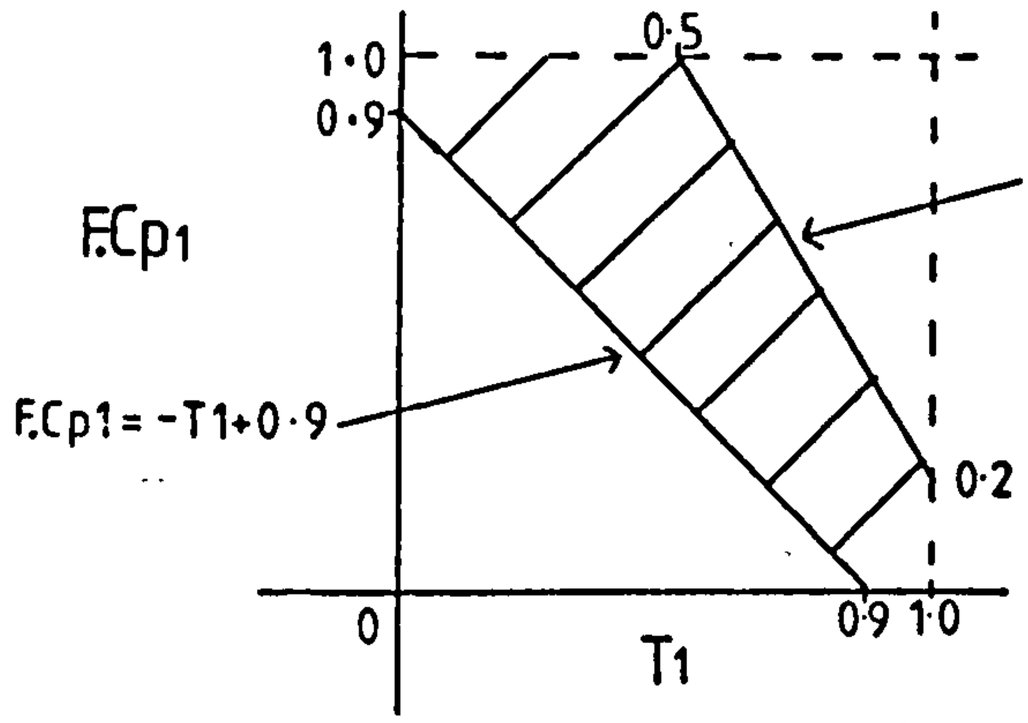


$0 \leqslant F.Cp1 \leqslant 1$

$0 \leqslant T1 \leqslant 1$

T1 and F.Cp1 vary independently with any combination being possible.

However, it may be possible to link T1 and F.Cp1 in the following way, for example as a result of a previous process involving this stream a high F.Cp value may occur with a low temperature and vice versa giving the following (shaded) feasible space.



$0 \leqslant F.Cp1 \leqslant 1$

$0 \leqslant T1 \leqslant 1$

$0.9 - F.Cp1 - T1 \leqslant 0$

$F.Cp1 + 1.6T1 - 1.8 \leqslant 0$

In terms of a feasibility checking procedure (DF = 0)

Max g_j for each j must be $\leqslant 0$
w.r.t. T1 and F.Cp1
s.t. $0 \leqslant F.Cp1 \leqslant 1$
$0 \leqslant T1 \leqslant 1$
$0.9 - F.Cp1 - T1 \leqslant 0$
$F.Cp1 + 1.6\ T1 - 1.8 \leqslant 0$

This would be a more time consuming optimization than that involving only simple bounds on F.Cp1 and T1 as inequality constraints involving both T1 and F.Cp1 must also be satisfied.

As the resulting network would probably be cheaper it would always be useful to specify such interactions arising from mass balance and external process considerations.

An example of the advantages of such an approach follows.

EXAMPLE PROBLEM 3

Of the five network structures given by Linnhoff (8) as solutions to problem 5SP1 the solution in Fig.4 results in the cheapest network able to handle the range of F.Cp values indicated. The flexible design procedure developed has been used to give fully feasible U.A values for all the exchangers.

In this case all flowrates have been allowed to vary independently by up to -5% (i.e. full 5% range).

Assume now that because of overall mass balance considerations the F.Cp values are linked as shown in Fig.5 , thereby reducing the feasible range. The F.Cp values of streams 2 to 5 are plotted against the F.Cp value of stream 1. ni is the nominal F.Cp value of stream number i.

It can be seen here that all F.Cp values F.Cp1 to F.Cp5 cover the range 95% - 100% of nominal conditions (as before) but that they are non-independent.

Low values of F.Cp1 will occur with low values of the other F.Cp values (2 to 5).

e.g. If F.Cp1 = 15.78 (95% of nominal) then F.Cp2, F.Cp3, F.Cp4, F.Cp5 will all be in the range 95% to 96% of their respective nominal values (ni).

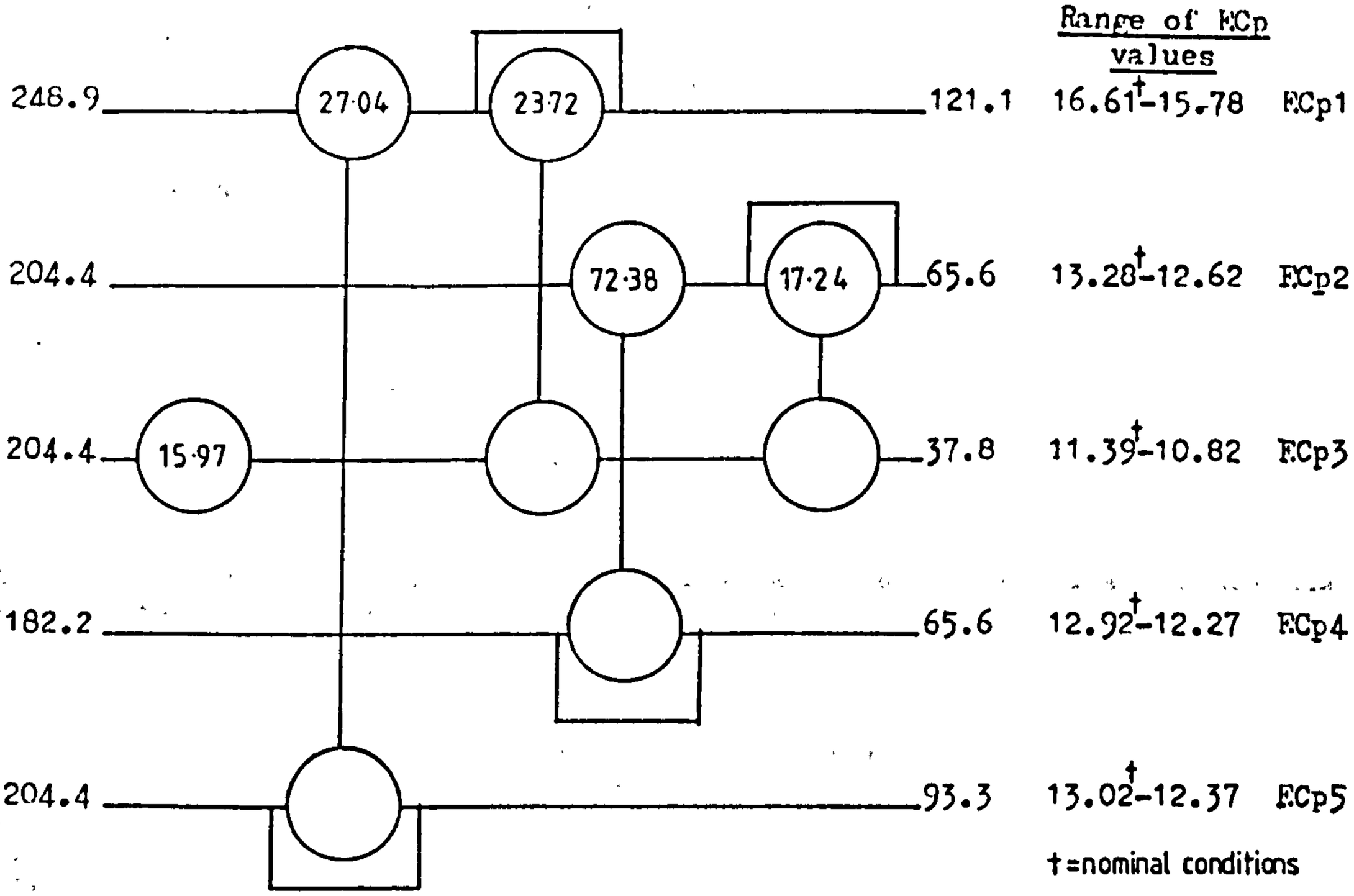


Fig.4 A fully feasible network for the problem 5SP1 with the range of F.Cp values shown.

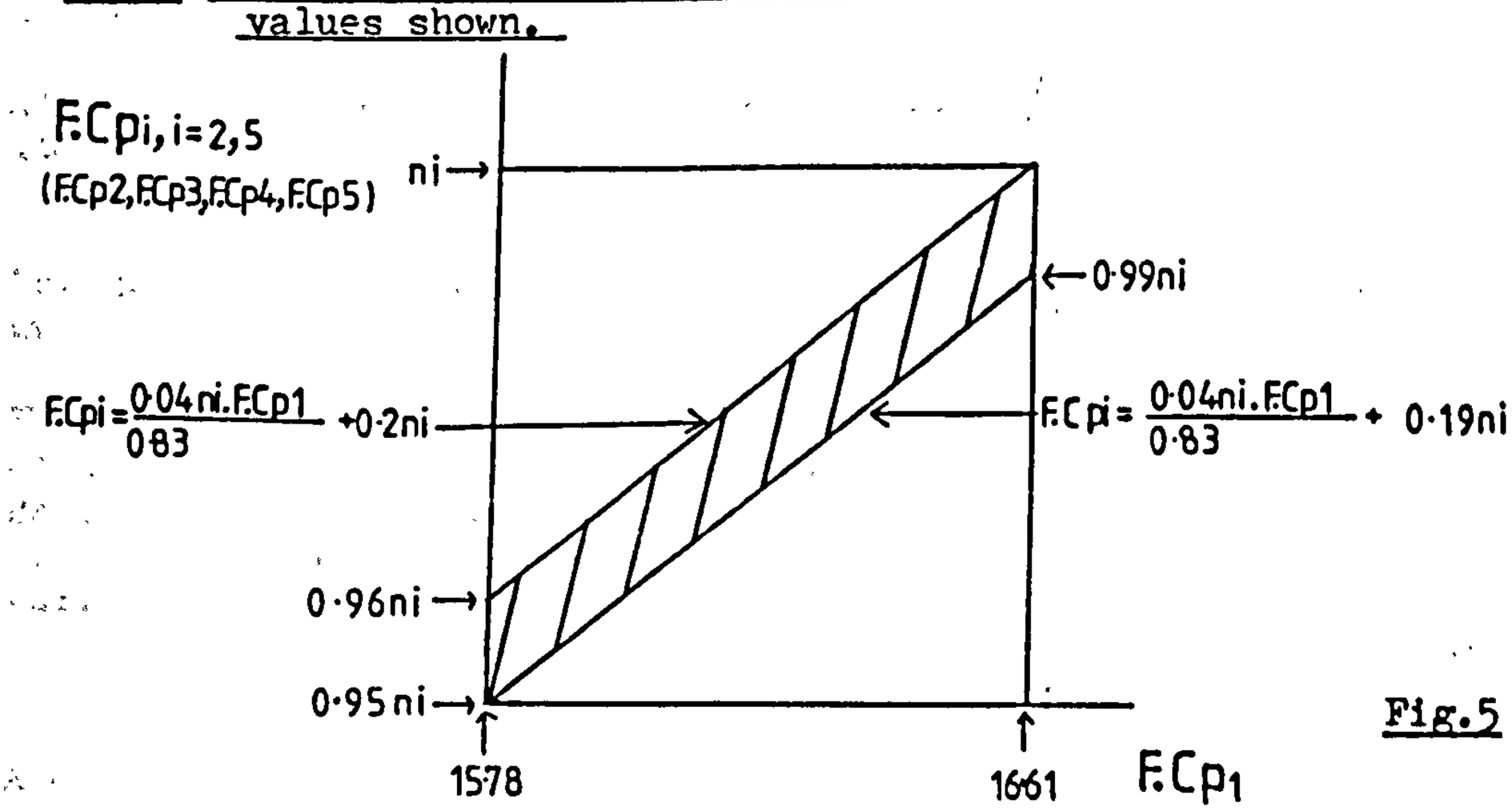


Fig.5

Thus it can be seen that in addition to the simple bounds on the F.Cp values

$$15.78 \leqslant F.Cp1 \leqslant 16.61$$
$$12.62 \leqslant F.Cp2 \leqslant 13.28$$
$$10.82 \leqslant F.Cp3 \leqslant 11.39$$
$$12.27 \leqslant F.Cp4 \leqslant 12.92$$
$$12.37 \leqslant F.Cp5 \leqslant 13.02$$

there are additional constraints of the form

$$\frac{0.04\ ni\ F.Cp1}{0.83} + 0.19\ ni \leqslant F.Cpi \leqslant \frac{0.04\ ni\ F.Cp1}{0.83} + 0.2\ ni \qquad \text{for } i = 2,3,4 \text{ \& } 5$$

A modified design program using NAG algorithm EØ4UAF was used to design a network subject to these constraints and fully feasible exchanger areas found, based on the same network structure as before. The results are as follows (using Linnhoff (8) values for sizing and costing):

Exchanger areas (m^2)

Exchanger	H	1	2	3	4
Nominal conditions only	12.63	30.66	18.30	74.42	15.61
Full 5% range	14.12	31.87	27.96	85.10	20.35
Reduced range	12.82	30.66	19.60	76.27	16.44

Costs ($) Nominal = 53260
Full 5% = 59330 (11.4% > Nom)
Reduced = 54200 (1.8% > Nom)

Therefore in this case there is a considerable incentive to try to reduce the required feasible range by specifying interactions between flows.

CONCLUSIONS

A method for designing flexible heat exchanger networks has been developed which can be used to design fully flexible networks for pinched and unpinched problems. It has been found that the careful placement of bypasses in a network can in many cases completely eliminate the need for optimizing control while ensuring maximum possible energy recovery. This also considerably simplifies the design problem because when the degrees of freedom are zero network constraints may be handled individually.

It has been shown that maintaining a fixed value of ΔT_{min} is not an appropriate objective for flexible design and more adaptable energy targets have been developed.

The design method allows for the existence of non-independent inlet variations which can considerably reduce the cost of the final network.

The requirement of extra flexibility provides a method of distinguishing between otherwise apparently similar networks designed for nominal conditions and by designing for different amounts of flexibility the cost associated with extra flexibility can be determined and compared with the cost of reducing the inlet variations.

ACKNOWLEDGEMENT

A.C. Beautyman was financially supported during the course of this work by an SERC award.

This work was part of a research programme supported by British Petroleum International plc in the form of an Extra-Mural Research Award.

REFERENCES

1) Morari, M., A.M. Lenhoff, D.F. Marselle and D.F. Rudd, AIChE 72nd Annual Meeting, San Francisco, November 1979, "Design of Resilient Energy Integrated Processing Systems".
2) Morari, M., D.F. Marselle and D.F. Rudd, AIChE Annual Meeting, Chicago, 1980, "Synthesis of Resilient Energy Management Systems".
3) Marselle, D.F., M.S. Thesis, Wisconsin-Madison University, 1980, "The Design of Resilient Heat Exchanger Networks and their Control Structure".
4) Marselle, D.F., M. Morari and D.F. Rudd, Chem.Eng.Sci. 37 (2), 259-270, 1982, "Design of Resilient Processing Plants 2: Design and Control of Energy Management Systems".
5) Grossmann, I.E., PhD Thesis, Imperial College, 1977, "Problems in the Optimum Design of Chemical Plants".
6) Grossmann, I.E., K.P. Halemane and R.E. Swaney, Proceedings of PSE '82, 239-262, Kyoto, Japan, August 1982, "Optimization Strategies for Flexible Chemical Processes".
7) Halemane, K.P. and I.E. Grossmann, AIChE 74th Annual Meeting, New Orleans, 1981, "Optimal Process Design Under Uncertainty".
8) Linnhoff, B., PhD Thesis, Leeds, 1979, "Thermodynamic Analysis in the Design of Process Networks".
9) Saboo, A.K. and M. Morari, Paper submitted to Chem.Eng.Sci, July 1982, "Design of Resilient Processing Plants 3: Some New Results on Heat Exchanger Network Synthesis".
10) Beautyman, A.C., Imperial College Annual Research Report, September 1982, "Flexibility of Heat Exchanger Networks".
11) Linnhoff, B. and J.R. Flower, AIChE Journal 24 (4), 633-654, July 1978, I. "Systematic Generation of Energy Optimal Networks"; II. "Evolutionary Generation of Networks with Various Criteria of Optimality".
12) Flower, J.R. and B. Linnhoff, AIChE Journal 26 (1), 1-9, January 1980, "A Thermodynamic-Combinatorial Approach to the Design of Optimum Heat Exchanger Networks".
13) Linnhoff, B. and E. Hindmarsh, Chem.Eng.Sci. 38, 745-763, 1983, "The Pinch Design Method for Heat Exchanger Networks".

REFERENCES

Al-Zakri, A.S. and K.J. Bell, "Estimate of the Performance of a System of Heat Exchangers when Uncertainties Exist, Paper presented at A.I.Ch.E. Spring National Meeting, Houston, (1981a).

Al-Zakri , A.S. and K.J. Bell, "Estimating Performance when Uncertainties Exist", Chem. Eng. Prog., 77, 39-49, July (1981b).

Avriel, M. and D.J. Wilde, "Engineering Design Under Uncertainty", Ind. Eng. Chem. Process Design Develop., 8(1), 124-131, (1969).

Bandler, J.W., "Optimization of Design Tolerances Using Nonlinear Programming", Journal Opt. Theory and Applications, 14 (1), 99-114, July (1974).

Berryman, J.E. and D.M. Himmelblau, "Effect of Stochastic Inputs and Parameters on Process Analysis and Design", Ind. Eng. Chem. Process Design Develop., 10 (4), 441-449, (1971).

Berryman, J.E. and D.M. Himmelblau, "Influence of Stochastic Inputs and Parameters on Heat Exchanger Design", Ind. Eng. Chem. Process Design Develop., 12 (2), 165-171, (1973).

Bingzhen, C., L. Yourun, H. Xiaorong and O. Pingkai, "Optimal Synthesis and Sensitivity Analysis of Heat Exchanger Networks" Paper presented at E.F.C.E. CHEMCOMP 1982, Antwerp, May (1982).

Buckley, P.S., "Sizing Process Equipment by Statistical Methods", Chem. Eng., 57, 112-114, September (1950).

Charnes, A. and W.W. Cooper, "Chance Constrained Programming", Man. Sci., 6 (1), 73-79, October (1959).

Chen, M.S.K., L.E. Erickson and L. Fan, "Consideration of Sensitivity and Parameter Uncertainty in Optimal Process Design", Ind. Eng. Chem. Process Design Develop., 9 (4), 514-521, (1970).

Dantzig, G.B., "Linear Programming Under Uncertainty", Man. Sci., 1 (3, 4), 197-206, April-July (1955).

Dantzig, G.B., "Recent Advances in Linear Programming", Man. Sci., 2 (2), 131-144, January (1956).

Dantzig, G.B., "Linear Programming and Extensions", Princeton University Press (1963), Chapter 25.

Dittmar, R. and K. Hartmann, "Calculation of Optimal Design Margins for Compensation of Parameter Uncertainty", Chem. Eng. Sci., 31, 563-568, (1976)

Doering, F.J. and J.L. Gaddy, "Simultaneous Overdesign and Optimization of Chemical Processes", Comp. and Chem. Eng., 3, 465-472, (1979).

Flower, J.R. and B. Linnhoff, "A Thermodynamic - Combinatorial Approach to the Design of Optimum Heat Exchanger Networks", A.I.Ch.E. Journal, 26 (1), 1-9, January (1980).

Freeman, R.A. and J.L. Gaddy, "Quantitative Overdesign of Chemical Processes", A.I.Ch.E. Journal, 21 (3), 436-440, May (1975).

Friedman, Y. and G.V. Reklaitis, "Flexible Solutions to Linear Programs Under Uncertainty: Inequality Constraints", A.I.Ch.E. Journal, 21 (1), 77-83, January (1975a).

Friedman, Y. and G.V. Reklaitis, "Flexible Solutions to Linear Programs Under Uncertainty: Equality Constraints", A.I.Ch.E. Journal, 21 (1), 83-90, January (1975b).

Gaddy, J.L. and O.L. Culberson, "Prediction of Variable Process Performance by Stochastic Flow Sheet Simulation", A.I.Ch.E. Journal, 19 (6), 1239-1243, (1973).

Gaddy, J.L. and R.A. Freeman, "Quantitative Overdesign of Pollution Control Systems", A.I.Ch.E. Symnposium Series No. 145, 71 128-135, (1974).

Grossmann, I.E., "Problems in the Optimum Design of Chemical Plants", PhD Thesis, Imperial College, (1977).

Grossmann, I.E. and K.P. Halemane, "Decomposition Strategy for Designing Flexible Chemical Plants", A.I.Ch.E. Journal, 28 (4), 686-694, July (1982).

Grossmann, I.E., K.P. Halemane and R.E. Swaney, "Optimization Strategies for Flexible Chemical Processes", Proceedings of PSE'82, 239-262, Kyoto, Japan, August (1982a).

Grossmann, I.E., K.P. Halmane and R.E. Swaney, "Optimization Strategies for Flexible Chemical Processes", Report DRC-06-33-82, Carnegie-Mellon University, December (1982b).

Grossmann, I.E., K.P. Halemane and R.E. Swaney, "Optimization Strategies for Flexible Chemical Processes", Comp. and Chem. Eng., 7 (4), 439-462, (1983).

Grossmann, I.E. and M. Morari, "Operability, Resiliency and Flexibility - Process Design Objectives for a Changing World", Paper presented at the 2nd International Conference on Foundations of Computer Aided Process Design, Snowmass, Colorado, June (1983).

Grossmann, I.E. and R.W.H. Sargent, "Optimum Design of Chemical Plants with Uncertain Parameters", A.I.Ch.E. Journal, 24 (6), 1021-1028, November (1978).

Grossmann, I.E. and R.W.H. Sargent, "Optimum Design of Multipurpose Chemical Plants", Ind. Eng. Chem. Process Design Develop., 18 (2), 343-348, (1979).

Halemane, K.P., "Studies in the Optimal Design of Flexible Chemical Plants", PhD Thesis, Carnegie-Mellon University, (1982a).

Halemane, K.P., "Studies in the Optimal Design of Flexible Chemical Plants",Report DRC-02-13-82, Carnegie-Mellon University, April (1982b).

Halemane, K.P. and I.E. Grossmann, "A Remark on the Paper "Theoretical and Computational Aspects of the Optimal Design Centering, Tolerancing and Tuning Problem"",I.E.E.E. Trans. on Circuits and Systems CAS-28 (2), 163-164, February (1981a).

Halemane, K.P. and I.E. Grossmann, "Optimal Process Design Under Uncertainty", Paper presented at A.I.Ch.E. 74th Meeting, New Orleans, (1981b).

Halemane, K.P. and I.E. Grossmann, "Optimal Process Design Under Uncertainty", Report DRC-06-30-82, Carnegie-Mellon University, April (1982).

Hanson, K.W., "The Synthesis of Heat Exchanger Networks Subject to Variable Sets of Constraints", PhD Thesis, Imperial College, London, (1984).

Hess, S.W. and H.A. Quigley, "Analysis of Risk in Investments Using Monte Carlo Techniques", A.I.Ch.E. Symposium Series No. 42, 59, 55-63, (1963).

Heydweiller, J.C. and L.T. Fan, "Process Synthesis Using Structural Parameters: Further Discussion of Inequality Constraints", A.I.Ch.E. Journal, 28 (1), 166-168, January (1982).

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

Institution of Chemical Engineers, "User Guide on Process Integration for the Efficient Use of Energy", I. Chem. E. (1982).

Johns, W.R., G. Marketos and D.W.T. Rippin, "The Optimal Design of Chemical Plants to meet Time-Varying Demands in the Prescence of Technical and Commercial Uncertainty", Paper presented at I. Chem. E. Design Congress, Birmingham, (1976).

Johns, W.R., G. Marketos and D.W.T. Rippin, "The Optimal Design of Chemical Plants to meet Time-Varying Demands in the Prescence of Technical and Commercial Uncertainty", Trans. I.Chem.E., 56, 249-257, (1978).

Kitterell, J.R. and C.C. Watson, "Don't Overdesign Process Equipment", Chem. Eng. Prog., 62 (4), 79-83, April (1966).

Kwak, B.M. and E.J. Haug, "Optimum Design in the Presence of Parametric Uncertainty", Journal Opt. Theory and Appl., 19 (4), 527-546, August (1976).

Lashmet, P.K. and S.Z. Szczepanski, "Efficiency, Uncertainty and Distillation Column Overdesign Factors", Ind. Eng. Chem. Process Design Develop., 13 (2), 103-106, (1974).

Linnhoff, B., "Thermodynamic Analysis in the Design of Process Networks", PhD Thesis, Leeds, (1979).

Linnhoff, B. and J.R. Flower, "Synthesis of Heat Exchanger Networks I. Systematic Generation of Energy Optimal Networks II. Evolutionary Generation of Networks with Various Criteria of Optimality", A.I.Ch.E. Journal, 24 (4), 633-654, July (1978).

Linnhoff, B. and E. Hindmarsh, "The Pinch Design Method for Heat Exchanger Networks", Chem. Eng. Sci., 3, 745-763, (1983).

Linnhoff, B. and E. Kotjabasakis, "Design of Heat Exchanger Networks for Specified Operability Requirements", Paper presented at First U.K. National Heat Transfer Conference, Leeds, July (1984).

Madansky, A., "Bounds on the Expectation of a Convex Function of a Multivariate Random Variable", Annals of Mathematical Statistics, 30, 743-746, (1959).

Malik, R.K. and R.R. Hughes, "Optimal Design of Flexible Chemical Processes", Comp. and Chem. Eng., 3, 473-485, (1979).

Mangasarian, O.L., "Nonlinear Programming Problems with Stochastic Objective Functions", Man.Sci., 10 (2), 353-359, January (1964).

Mangasarian, O.L. and J.B. Rosen, "Inequalities for Stochastic Nonlinear Programming Problems", Operations Research, 12 (1), 143-154, January-February (1964).

Marketos, G., "The Optimal Design of Chemical Plant Considering Uncertainty and Changing Circumstances", Diss. ETH 5607 Zurich, (1975).

Marselle, D.F., "The Design of Resilient Heat Exhcanger Networks and their Control Structure", MS Thesis, Wisconsin-Madison University, (1980).

Marselle, D.F., M. Morari and D.F. Rudd, "Design of Resilient Processing Plants 2: Design and Control of Energy Management Systems", Chem. Eng. Sci., 37 (2), 259-270, (1982).

Morari, M., "Operability Measures for Process Design", I.Chem.E. Symposium Series No. 74 'Understanding Process Integration', 131-140, (1982a).

Morari, M., "Flexibility and Resiliency of Process Systems", Proceedings of PSE'82, 223-237, Kyoto, Japan, August (1982b).

Morari, M., "Flexibility and Resiliency of Process Systems", Comp. and Chem. Eng., 7 (4), 423-437, (1983).

Morari, M., A.M. Lenhoff, D.F. Marselle and D.F. Rudd, "Design of Resilient Energy Integrated Processing Systems", Paper presented at A.I.Ch.E. 72nd Annual Meeting, San Francisco, November (1979).

Morari, M., D.F. Marselle and D.F. Rudd, "Synthesis of Resilient Energy Management Systems", Paper presented at A.I.Ch.E. Annual Meeting, Chicago, (1980).

Morari, M. and D.C. Woodcock, "Design of resilient processing plants VI", In preparation (1983).

Nishida, N., A. Ichikawa and E. Tazaki, "Optimal Design and Control in a Class of Distributed Parameter Systems under Uncertainty-Application to Tubular Reactor with Catalyst Deactivation", A.I.Ch.E. Journal, 18 (3), 561-568, May (1972).

Nishida, N. A. Ichikawa and E. Tazaki, "Synthesis of Optimal Process Systems with Uncertainty", Ind. Eng. Chem., Process Design Develop., 13 (3), 209-214, (1974).

Nishida, N., G. Stephanopoulos and A.W. Westerberg, "A Review of Process Synthesis", A.I.Ch.E. Journal, 27 (3), 321-351, May (1981).

Parkinson, A.R., J.S. Liebman, C.O. Pedersen and A.B. Templeman, "The Optimal Design of Resilient Heat Exchanger Networks", A.I.Ch.E. Symposium Series No. 214, 78, 85-98, (1982).

Perkins, J.D., "Efficient Solution of Design Problems Using a Sequential-Modular Flowsheeting Programme", Comp. and Chem. Eng., 3, 375-381, (1979).

Polak, E. and A. Sangiovanni-Vincentelli, "Theoretical and Computational Aspects of the Optimal Design Centering, Tolerancing and Tuning Problem", I.E.E.E. Trans. on Circuits and Systems CAS-26 (9), 795-813, September (1979).

Powers, G.J. and J.F. Mayer, "Design of a Catalytic Reactor-Separator System with Uncertainty in Catalyst Activity", Ind. Eng. Chem. Process Design Develop., 14 (1), 41-44, (1975).

Raghavan, S., "Heat Exchanger Network Synthesis", PhD Thesis, Purdue University, (1977).

Rohrer, R.A. and M. Sobral, "Sensitivity Considerations in Optimal System Design", I.E.E.E. Trans. Automatic Control AC-10, 43-48, (1965).

Rudd, D.F. and C.C. Watson, "Strategy of Process Engineering", Wiley N.Y. (1968).

Saboo, A.K. and M. Morari, "Design of Resilient Processing Plants-IV. Some New Results on Heat Exchanger Network Synthesis", Chem. Eng. Sci., 39 (3), 579-592, (1984).

Saletan, D.I., "Specifying Distillation Safety Factors", Chem. Eng. Prog., 65 (5), 80-82, May (1969).

Saletan, D.I. and A.V. Caselli, "Optimum Design Capacity of New Plants", Chem. Eng. Prog., 59 (5), 69-75, May (1963).

Savage, L.J., "The Foundations of Statistics", Wiley N.Y. (1954).

Schlosser, H., "An Approach to the Optimal Design of Chemical Plants Operating Under Variable Conditions", Comp. and Chem. Eng., 3, 461-463, (1979).

Seinfeld, J.H. and W.L. McBride, "Optimization with Multiple Performance Criteria", Ind. Eng. Chem. Process Design Develop., 9 (1), 53-57, (1970).

Smith, D.J., "Reliability and Maintainability in Perspective", MacMillan Press (1981).

Swaney, R.E. and I.E. Grossmann, "A Metric for Operational Flexibility in Chemical Process Design", Paper presented at A.I.Ch.E. Annual Meeting, Los Angeles, November (1982).

Takamatsu, T., I. Hashimoto and H. Ohno, "Optimal Design of a Large Complex System from the Viewpoint of Sensitivity Analysis", Ind. Eng. Chem. Process Design Develop., 9 (3), 368-379, (1970).

Takamatsu, T., I. Hashimoto and S. Shioya, "On Design Margin for Process Systems with Parameter Uncertainty", Journal Chem. Eng. Japan, 6 (5), 453-457, (1973).

Tayyabkhan, M.T. and T.C. Richardson, "Monte Carlo Techniques", Chem. Eng. Prog., 61 (1), 78-82, January (1965).

Wantanabe, N., Y. Nishimura and M. Matsubara, "Optimal Design of Chemical Processes Involving Parameter Uncertainty", Chem. Eng. Sci., 28, 905-913, (1973).

Weisman, J. and A.G. Holzman, "Optimal Process System Design Under Conditions of Risk", Ind. Eng. Chem. Process Design Develop., 11 (3), 386-396, (1972a).

Weisman, J. and A.G. Holzman, "Engineering Design Optimization Under Risk", Man. Sci., 19 (3), 235-249, November (1972b).

Wen, C.Y. and T.M. Chang, "Optimal Design of Systems Involving Parameter Uncertainty", Ind. Eng. Chem. Process Design Develop., 71 (1), 49-53, January (1968).

Westerberg, A.W. and R.L. McGalliard, "Structural Sensitivity Analysis in Design Synthesis", Chem. Eng. Journal, 4, 127-138, (1972).

White, C.H., "Operations Analysis Under Uncertainty", A.I.Ch.E. Symposium Series No. 220, 78, 30-35, (1982).

PAPER COPIES MADE FROM THE
FICHE
APPENDIX F AND G

```
      PROGRAM HERON1 (INPUS,OUT,INPUT,OUTPUT,TAPE5=INPUS,TAPE6=OUT,TAPE7=
     +INPUT,TAPE8=OUTPUT)
C*
C*    PROGRAM HERON1. MODELS THE OPERATION OF A HEAT EXCHANGER NETWORK AND
C*    MAXIMIZES INDIVIDUAL CONSTRAINT FUNCTION VALUES.  (UNABLE TO MODEL
C*    OPTIMIZING CONTROL)
C*
C*    LIBRARY ROUTINES CALLED: E04JAF FROM NAG LIBRARY
C*                             BROYDN UNPUBLISHED ROUTINE SEE PERKINS (1979)
C*
C*    USE: BHERON (DATA,OUT,INPUT,OUTPUT)
C*         DATA-PROBLEM DATA
C*         OUT-RESULTS
C*         INPUT-LEVEL OF OUTPUT REQUIRED (0 TO 2) OR 9 FOR INTERACTIVE USE
C*         OUTPUT-PROMPTS FOR INTERACTIVE USE
C*
C*    ---------
C*
C*    PROBLEM DATA REQUIRED:
C*
C*    1) TOTAL NUMBER OF UNITS (TNUNTS) , TOTAL NUMBER OF SUBSTREAMS (TNSTR)
C*
C*    2) UNIT MATRIX (MATRXU (TNUNTS,5)). FOR EACH UNIT IN TURN:
C*
C*    UNIT NUMBER , INPUT STREAM 1 , INPUT 2 , OUTPUT 1 , OUTPUT 2
C*
C*    3) TOTAL NUMBER OF PROCESS STREAMS (TNINPT)
C*
C*    4) FOR EACH PROCESS STREAM INPUT IN TURN:
C*
C*    SUBSTREAM NUMBER , LOWER BOUND F.CP , UPPER BOUND F.CP , LOWER BOUND T ,
C*    UPPER BOUND T
C*
C*    5) TARGET T PROCESS STREAM 1 , ... , TARGET T PROCESS STREAM TNINPT
C*
C*    6) MIN PHYSICALL POSSIBLE TEMP. APPROACH (DELTMN) , MAXIMUM ALLOWABLE
C*    UTILITY USE AS A PERCENTAGE > THEORETICAL MIN.
C*
C*    7) U FOR EXCHANGER 1 , ... , U FOR EXCHANGER TNHXER
C*
C*    8) A FOR EXCHANGER 1 , ... , A FOR EXCHANGER TNHXER
C*
C*    9) NUMBER OF CONSTRAINT TO MAXIMIZE IF NOT IN INTERACTIVE MODE
C*
C*    ---------
C*
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      CHARACTER IQUEST
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /CON1C/ CC(200),KFEAS(200),IFEAS
      COMMON /CATC/ ICATCH,CFCP(50),CT(50)
      DIMENSION BL(50),BU(50),XC(50),IW(50),W(2000)
      REWIND 5
      REWIND 6
      IP=KNT=INTES=INTER=INITRN=0
      SN=-1.0
```

```
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      COMMON /CON1C/ CC(200),KFEAS(200),IFEAS
      IFEAS=0
      DO 3 J=1,((3*TNHXER)+1+TNSPLT)
      KFEAS(J)=0
    3 CONTINUE
      IF(IPINCH.EQ.1.OR.IPINCH.EQ.0.AND.LINN.EQ.0)THEN
      CC(1)=(TOTAL*(1+PRCENT/100.0)-TUTIL)/TOTAL
      IF(CC(1).LT.0.OR.CC(1).GT.(PRCENT/100.0))THEN
      KFEAS(1)=1
      IFEAS=1
      ENDIF
      ELSE
      CC(1)=0
      ENDIF
      DO 1 J=1,TNHXER
      CC((3*J)-1)=HXL(J)/100.0
      IF(CC((3*J)-1).LT.0)THEN
      KFEAS((3*J)-1)=1
      IFEAS=1
      ENDIF
      CC(3*J)=(T(MATRXU(NHXER(J),2))-T(MATRXU(NHXER(J),5))-DELTMN)/100.0
      IF(CC(3*J).LT.0)THEN
      KFEAS(3*J)=1
      IFEAS=1
      ENDIF
      CC((3*J)+1)=(T(MATRXU(NHXER(J),4))-T(MATRXU(NHXER(J),3))-DELTMN)
     +/100.0
      IF(CC((3*J)+1).LT.0)THEN
      KFEAS((3*J)+1)=1
      IFEAS=1
      ENDIF
    1 CONTINUE
      DO 2 J=1,TNSPLT
      CC((3*TNHXER)+1+J)=SN*SPLFRN(J)
      IF(SPLFRN(J).LT.0.OR.SPLFRN(J).GT.1)THEN
      KFEAS((3*TNHXER)+1+J)=1
      IFEAS=1
      ENDIF
    2 CONTINUE
      RETURN
      END
      SUBROUTINE FEAS(ISWT)
C
C*    PRODUCES OUTPUT OF CONSTRAINT VALUES
C*     (ISWT=0,FULL OUTPUT  ISWT=1,INFEASIBLE AND OPTIMIZED VALUES ONLY)
C
      LOGICAL LOGI,LOGIJ
      INTEGER TNHXER,TNSPLT
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      COMMON /CON1C/ CC(200),KFEAS(200),IFEAS
      IF(ISWT.EQ.1)THEN
      WRITE(6,61)
      ELSE
      WRITE(6,60)
      ENDIF
```

```
      IF (NTYPE (MATRXU (NUTIL (J) ,2) ) .NE.4) THEN
      NTYPE (MATRXU (NUTIL (J) ,2) ) =9
      ELSE
      NTYPE (MATRXU (NUTIL (J) ,2) ) =10
      ENDIF
      IRDF=IRDF+1
      ENDIF
      ENDIF
   17 CONTINUE
      ENDIF
      RETURN
      END
      SUBROUTINE EQSOL (ACCUR)
C
C*    SUBROUTINE SOLVES THE SET OF DESCRIBING EQUATIONS FOR THE
C*    NETWORK
C
      INTEGER TNINPT, TNSTR, TNUNKF, TNHXER, TNUTIL, TNMIX, TNSPLT, TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR (50,2) , TSR (50,2) , TT (50) , TNSTR, TNUNTS, TNINPT,
     +MATRXU (50,5) , NINPT (50) , U (50) , A (50) , DELTMN, PRCENT, IP
     +, IPRNT, INTER, IPRN2, INTES, KAM, SN, KNT, ICONST, IRDIF (50)
      COMMON /MASBLC/ IFLAG, TNUNKF, NUNKF (50)
      COMMON /COUNTC/ TNHXER, TNMIX, TNSPLT, TNUTIL, NHXER (50) , NMIX (20) ,
     +NSPLT (20) , NUTIL (50) , IFLAF
      COMMON /GUESSC/ HXL (50) , SPLFRN (20) , DELHMN
      COMMON /TYPEC/ NTYPE (100) , NHOT, IRDF, LINN
      COMMON /EQSOLC/ IFLEE, HUTIL (50) , INITRN, KALL, KOUNT, TUTIL
      COMMON /XSETC/ X (200) , NBROY, TNDF, IFLAX
      COMMON /TMPROC/ T (100) , FCP (100)
      DIMENSION F (200) , BOUND (150) , W (2500)
      IFLEE=IDELT=KOUNT=KOND=0
      IF (INITRN.EQ.0) CALL GUESS
C
C*    REMOVE REDUNDANT UTILITY EXCHANGERS FROM NETWORK
C
      IF (IRDF.GT.0) THEN
      DO 20 J=1, TNUTIL
      IF (NTYPE (MATRXU (NUTIL (J) ,2) ) .EQ.9.OR.NTYPE (MATRXU (NUTIL (J) ,2) ) .EQ.
     +10) THEN
      T (MATRXU (NUTIL (J) ,2) ) =T (MATRXU (NUTIL (J) ,4) )
      ENDIF
   20 CONTINUE
      ENDIF
C
C*    SET TEMPS. EQUAL AROUND SPLITTERS AND CALCULATE OUTPUT FCP'S
C
   17 IF (TNSPLT.GT.0) THEN
      DO 7 J=1, TNSPLT
      T (MATRXU (NSPLT (J) ,4) ) =T (MATRXU (NSPLT (J) ,2) )
      T (MATRXU (NSPLT (J) ,5) ) =T (MATRXU (NSPLT (J) ,2) )
      FCP (MATRXU (NSPLT (J) ,4) ) =FCP (MATRXU (NSPLT (J) ,2) ) *SPLFRN (J)
      FCP (MATRXU (NSPLT (J) ,5) ) =FCP (MATRXU (NSPLT (J) ,2) ) * (1-SPLFRN (J) )
    7 CONTINUE
      ENDIF
C
C*    SET ALL REMAINING UNKNOWN FCP'S TO CURRENT CORRECT VALUES
C
      DO 8 J=1, TNHXER
      DO 4 JJ=1, TNUNKF
      IF (MATRXU (NHXER (J) ,4) .EQ.NUNKF (JJ) .OR.MATRXU (NHXER (J) ,5) .EQ.
     +NUNKF (JJ) ) THEN
      IF (MATRXU (NHXER (J) ,4) .EQ.NUNKF (JJ) ) THEN
      FCP (MATRXU (NHXER (J) ,4) ) =FCP (MATRXU (NHXER (J) ,2) )
      ELSE
      FCP (MATRXU (NHXER (J) ,5) ) =FCP (MATRXU (NHXER (J) ,3) )
```

```
      WRITE(6,89) TUTIL
      IF (TNSPLT.GT.0) THEN
      WRITE(6,82)
      WRITE(6,87)
      DO 11   J=1,TNSPLT
      WRITE(6,88) J,MATRXU(NSPLT(J),1),SPLFRN(J)
   11 CONTINUE
      ENDIF
      WRITE(6,82)
   62 FORMAT(1H /)
   63 FORMAT(1H ,'FOR A MIN. TEMP. APPROACH OF     ',F9.2)
   64 FORMAT(1H ,'(COLD STREAM) PINCH TEMP. IS     ',F9.2)
   65 FORMAT(1H ,'MINIMUM HOT UTILITY USAGE IS     ',F9.2)
   66 FORMAT(1H ,'MINIMUM COLD UTILITY USAGE IS    ',F9.2)
   67 FORMAT(1H ,'MINIMUM TOTAL UTILITY USAGE IS   ',F9.2)
   79 FORMAT(1H ,'EXIT FROM BROYDN AFTER ',I5,' ITERATIONS')
   80 FORMAT(1H /1H ,2X,'STREAM NO.',4X,'TEMPERATURE',7X,'FCP'/)
   81 FORMAT(1H ,4X,I3,11X,F6.2,6X,F8.2)
   82 FORMAT(1H /)
   83 FORMAT(1H ,'HEAT EXCHANGER',2X,'UNIT NO.  ',3X,'HEAT LOAD'
     +,6X,'U',6X,'AREA'/)
   84 FORMAT(1H ,7X,I2,9X,I2,6X,F9.2,2X,F9.2,1X,F8.2)
   85 FORMAT(1H ,'UTILITY EXCHANGER',2X,'UNIT NO.',3X,'HEAT LOAD'/)
   86 FORMAT(1H ,7X,I2,12X,I2,5X,F9.2)
   87 FORMAT(1H ,'SPLITTER',3X,'UNIT NO.',3X,'SPLIT FRACTION'/)
   88 FORMAT(1H ,3X,I2,9X,I2,10X,F5.3)
   89 FORMAT(1H ,'TOTAL UTILITY USAGE IS       ',F9.2)
   90 FORMAT(1H ,7X,I2,9X,I2,7X,'NEGLIGIBLE',F9.2,1X,F8.2)
      RETURN
      END
```

```
      READ (7,*) IPRN2
      IF (IPRN2.EQ.9) THEN
      INTES=1
      WRITE (8,*) ('INPUT LEVEL OF OUTPUT  (0,1 OR 2) ')
      READ (7,*) IPRN2
      ENDIF
      IQUEST='N'
      IF (INTES.EQ.1) THEN
      WRITE (8,*) ('DO YOU WANT TO ENTER PRINT PARAMETER FOR EQN.SOLVER')
      WRITE (8,*) ('(DEFAULT=0)  (Y OR N)  ?')
      READ (7,7000) IQUEST
 7000 FORMAT (A1)
      IF (IQUEST.EQ. 'Y') THEN
      WRITE (8,*) ('INPUT VALUE ')
      READ (7,*) IPRNT
      ENDIF
      IF (IPRNT.EQ.9) THEN
      INTER=1
      WRITE (8,*) ('INPUT LEVEL OF OUTPUT  (0,1,2,3 OR 4) ')
      READ (7,*) IPRNT
      ENDIF
      ENDIF
      IF (IQUEST.NE. 'Y'.OR.INTES.EQ.0) IPRNT=0
      READ (5,*) TNUNTS,TNSTR
      READ (5,*) ((MATRXU (J,K) ,K=1,5) ,J=1,TNUNTS)
      READ (5,*) TNINPT
      DO 7 J=1,TNINPT
      READ (5,*) NINPT (J) , (FCPR (J,K) ,K=1,2) , (TSR (J,K) ,K=1,2)
    7 CONTINUE
      READ (5,*) (TT (J) ,J=1,TNINPT)
      READ (5,*) DELTMN,PRCENT
      CALL COUNT
      IF (IFLAF.EQ.1) GO TO 99
      READ (5,*) (U (J) ,J=1,TNMXER)
      READ (5,*) (A (J) ,J=1,TNMXER)
      READ (5,*) ICONST
      N=0
      DO 4 J=1,TNINPT
      AMINJ=AMIN1 (FCPR (J,1) ,FCPR (J,2) )
      AMINT=AMIN1 (TSR (J,1) ,TSR (J,2) )
      AMAXJ=AMAX1 (FCPR (J,1) ,FCPR (J,2) )
      AMAXT=AMAX1 (TSR (J,1) ,TSR (J,2) )
      FCPR (J,1) =AMINJ
      FCPR (J,2) =AMAXJ
      TSR (J,1) =AMINT
      TSR (J,2) =AMAXT
      IF (ABS (FCPR (J,1) -FCPR (J,2) ) .GT.1.0E-05) THEN
      N=N+1
      IRDIF (J) =1
      ELSE
      IRDIF (J) =0
      ENDIF
      IF (ABS (TSR (J,1) -TSR (J,2) ) .GT.1.0E-05) THEN
      N=N+1
      IRDIF (TNINPT+J) =1
      ELSE
      IRDIF (TNINPT+J) =0
      ENDIF
    4 CONTINUE
    1 CALL START (LEST)
      KAM=0
      IF (LEST.EQ.1) GO TO 98
      KNT=KNT+1
      WRITE (6,7001) KNT
 7001 FORMAT (1H , 'RUN NUMBER ',I3,' STARTS')
      CALL TMPRO
```

```
      ENDIF
      ENDIF
    4 CONTINUE
    8 CONTINUE
C
C*    WHERE POSSIBLE COMPLETE M&E BALANCES AROUND MIXERS
C
      IF (TNMIX.GT.0) THEN
      DO 9 J=1,TNMIX
      FCP (MATRXU (NMIX (J) ,4) ) =FCP (MATRXU (NMIX (J) ,2) ) +FCP (MATRXU (NMIX (J) ,3
     +) )
      IF (NTYPE (MATRXU (NMIX (J) ,4) ) .EQ.4) THEN
      T (MATRXU (NMIX (J) ,4) ) = ( (FCP (MATRXU (NMIX (J) ,2) ) *T (MATRXU (NMIX (J) ,2)
     +) ) + (FCP (MATRXU (NMIX (J) ,3) ) *T (MATRXU (NMIX (J) ,3) ) ) ) /FCP (MATRXU (NMIX
     + (J) ,4) )
      ENDIF
    9 CONTINUE
      ENDIF
      IF (INITRN.EQ.1) CALL GUESS
C
C*    CALCULATE UTILITY LOADS
C
      DO 10 J=1,TNUTIL
      HUTIL (J) = (T (MATRXU (NUTIL (J) ,4) ) -T (MATRXU (NUTIL (J) ,2) ) ) *
     *FCP (MATRXU (NUTIL (J) ,2) )
   10 CONTINUE
      IFLAX=0
      IF (INITRN.EQ.2) CALL XSET
      IF (IFLAX.EQ.1) GO TO 29
      I=1
      TUTIL=0.0
      DO 19 J=1,TNUTIL
      TUTIL=TUTIL+ABS (HUTIL (J) )
   19 CONTINUE
C
C*    CALCULATE EQUATION VALUES (F) FOR USE IN BROYDN
C*    SHOULD ALL BE ZERO AT THE SOLUTION
C
      DO 11 J=1, (3*TNHXER) ,3
C
C*    ENERGY BALANCE ON HEAT EXCHANGER HOT STREAM
C
      F (J) = ( (T (MATRXU (NHXER (I) ,2) ) -T (MATRXU (NHXER (I) ,4) ) ) *FCP
     + (MATRXU (NHXER (I) ,2) ) ) -HXL (I)
C
C*    ENERGY BALANCE ON HEAT EXCHANGER COLD STREAM
C
      F (J+1) = ( (T (MATRXU (NHXER (I) ,3) ) -T (MATRXU (NHXER (I) ,5) ) ) *FCP
     + (MATRXU (NHXER (I) ,3) ) ) +HXL (I)
      DELT1=T (MATRXU (NHXER (I) ,2) ) -T (MATRXU (NHXER (I) ,5) )
      DELT2=T (MATRXU (NHXER (I) ,4) ) -T (MATRXU (NHXER (I) ,3) )
      CALL DELTA (DELT1,DELT2,DELTAX,IDELT)
C
C*    OVERALL ENERGY BALANCE ON HEAT EXCHANGERS
C*    EXCHANGER HEAT LOADS < DELHMN ARE TREATED AS
C*    BEING NEGLIGIBLE (IE. CORRECT SCALING IS
C*    REQUIRED)
C
      IF (ABS (HXL (I) ) .LT.DELHMN) THEN
      F (J+2) =HXL (I) * (ACCUR**2)
      ELSE
      F (J+2) = (U (I) *A (I) ) *DELTAX-HXL (I)
      ENDIF
      I=I+1
   11 CONTINUE
      I=1
```

```
      IF (ISWT.EQ.1) WRITE (6,62) ICONST
      J=1
      CALL LOGIT (LOGI, ISWT, J, KFEAS (J) , ICONST, LOGIJ)
      IF (IPINCH.EQ.0.AND.LINN.EQ.1) THEN
      IF (LOGI) WRITE (6,63) J, CC (J)
      ELSE
      IF (LOGIJ) WRITE (6,64)
      IF (LOGI) WRITE (6,65) J, CC (J) , ( (TUTIL-TOTAL) /TOTAL) *100.0
      ENDIF
      DO 1 K=1, TNHXER
      J=J+1
      CALL LOGIT (LOGI, ISWT, J, KFEAS (J) , ICONST, LOGIJ)
      IF (LOGIJ) WRITE (6,64)
      IF (ABS (CC (J) *100.0) .LT. DELHMN) WRITE (6,66)
      IF (LOGI) WRITE (6,67) J, NHXER (K) , CC (J) , HXL (K)
      J=J+1
      CALL LOGIT (LOGI, ISWT, J, KFEAS (J) , ICONST, LOGIJ)
      IF (LOGIJ) WRITE (6,64)
      IF (ABS (HXL (K) ) .LT. DELHMN) WRITE (6,70)
      IF (LOGI) WRITE (6,68) J, NHXER (K) , CC (J) , (T (MATRXU (NHXER (K) , 2) ) -T (MATRX
     +U (NHXER (K) , 5) ) )
      J=J+1
      CALL LOGIT (LOGI, ISWT, J, KFEAS (J) , ICONST, LOGIJ)
      IF (LOGIJ) WRITE (6,64)
      IF (ABS (HXL (K) ) .LT. DELHMN) WRITE (6,70)
      IF (LOGI) WRITE (6,69) J, NHXER (K) , CC (J) , (T (MATRXU (NHXER (K) , 4) ) -T (MATRX
     +U (NHXER (K) , 3) ) )
    1 CONTINUE
      IF (TNSPLT.GT.0) THEN
      DO 2 K=1, TNSPLT
      J=J+1
      CALL LOGIT (LOGI, ISWT, J, KFEAS (J) , ICONST, LOGIJ)
      IF (LOGIJ) WRITE (6,64)
      IF (LOGI) WRITE (6,71) J, NSPLT (K) , CC (J) , SPLFRN (K)
    2 CONTINUE
      ENDIF
      WRITE (6,72)
      IF (ISWT.EQ.0) THEN
      IF (IFEAS.EQ.0) THEN
      WRITE (6,73)
      ELSE
      WRITE (6,74)
      ENDIF
      ENDIF
      RETURN
   60 FORMAT (1H , 'CONSTRAINT VALUES')
   61 FORMAT (1H , 'MINIMIZED AND INFEASIBLE CONSTRAINTS')
   62 FORMAT (1H , 'CONSTRAINT BEING MINIMIZED IS NO.  ', I2)
   63 FORMAT (1H , 'NO. ', I2, '  (UTILITY USAGE)                C = ', E10.3, ' (USE
     +=THEOR.MIN  ) ')
   64 FORMAT (1H , 'XXXXX  NEXT VALUE IS INFEASIBLE  XXXXX')
   65 FORMAT (1H , 'NO. ', I2, '  (UTILITY USAGE)                  C = ', E10.3, ' (', F5
     +.1, ' P.C.> MIN ) ')
   66 FORMAT (1H , '*****  NEXT VALUE IS NEGLIGIBLE  *****')
   67 FORMAT (1H , 'NO. ', I2, '  (HEAT LOAD, UNIT NO.  ', I2, ' ) ', ' C = ', E10.3,
     +' ( HT.LOAD =  ', F5.1, ') ')
   68 FORMAT (1H , 'NO. ', I2, '  (UNIT ', I2, '  HOT END DELT)  C = ', E10.3, ' (
     +', F6.1, '  DEGREES ) ')
   69 FORMAT (1H , 'NO. ', I2, '  (UNIT ', I2, '  COLD END DELT) C = ', E10.3, ' (
     +', F6.1, '  DEGREES ) ')
   70 FORMAT (1H , '*****  NEXT VALUE IS IRRELEVANT  *****')
   71 FORMAT (1H , 'NO. ', I2, '  (UNIT ', I2, ' SPLIT FRACTION) C = ', E10.3, ' (
     + SPLIT =  ', F7.4, ') ')
   72 FORMAT (1H )
   73 FORMAT (1H , 'ALL CONSTRAINTS FEASIBLE')
   74 FORMAT (1H , 'CURRENT POINT INFEASIBLE')
```

```
      WRITE(6,60)
   60 FORMAT(1H /1H ,50('%'),1H /)
      CALL SETUP(N,BL,BU,XC,LIW,LW,IFAIL,IBOUND)
      IF(INTES.EQ.1)THEN
      WRITE(8,*)('WHICH CONSTRAINT TO BE MINIMIZED')
      READ(7,*)ICONST
      TNHX3=3*TNHXER
      IF(ICONST.GT.(TNHX3+1).AND.ICONST.LE.(TNHX3+1+TNSPLT))THEN
      WRITE(8,*)('SPLIT FRACTION   ENTER + FOR MAX , - FOR MIN')
      READ(7,7000)IQUEST
      IF(IQUEST.EQ.'+')THEN
      SN=-1.0
      ELSE
      SN=1.0
      ENDIF
      ENDIF
      IF(ICONST.GT.(TNHX3+1+TNSPLT))THEN
      WRITE(6,*)('XXXXX  CONSTRAINT NUMBER DOES NOT EXIST  XXXXX')
      GO TO 98
      ENDIF
      ENDIF
      CALL FUNCT1(N,XC,FC)
      WRITE(6,*)('CALLING OPTIMIZING ROUTINE E04JAF FROM NAG8F')
      IF(IFLAX.EQ.1)THEN
      CALL WRYTE
      WRITE(6,60)
      GO TO 99
      ENDIF
      CALL E04JAF(N,IBOUND,BL,BU,XC,FC,IW,LIW,W,LW,IFAIL)
      WRITE(6,*)('EXITING FROM OPTIMIZING ROUTINE')
      CALL RSLTS(IFAIL)
      CALL WRYTE
      ISWT=1
      CALL FEAS(ISWT)
      IF(IPRN2.GT.0)THEN
      CALL RITE
      ISWT=0
      CALL FEAS(ISWT)
      ENDIF
      WRITE(6,60)
      IF(ICATCH.EQ.1)THEN
      WRITE(6,*)('*****  WARNING  *****')
      WRITE(6,*)('A SIGNIFICANTLY LOWER CONSTRAINT VALUE FOUND AT')
      DO 5 J=1,TNINPT
      FCP(NINPT(J))=CFCP(J)
      T(NINPT(J))=CT(J)
    5 CONTINUE
      CALL WRYTE
      WRITE(6,60)
      ENDIF
      IF(INTES.EQ.1)THEN
      WRITE(8,*)('GO AGAIN (Y OR N)  ?')
      READ(7,7000)IQUEST
      IF(IQUEST.EQ.'Y')GO TO 1
      ENDIF
   99 WRITE(6,*)('PROGRAM ENDS ')
   98 STOP
      END
      SUBROUTINE START(LEST)
C
C*    PROVIDES INITIAL STARTING POINT FOR OPTIMIZATION
C
      INTEGER TNINPT
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
```

```
      END
      SUBROUTINE LOGIT(LOGI,ISWT,J,KFEASJ,ICONST,LOGIJ)
C
C*    USED IN FEAS TO CONTROL PRINTING
C
      LOGICAL LOGI,LOGJ,LOGIJ
      LOGI=ISWT.EQ.1.AND.KFEASJ.EQ.1
      LOGJ=J.EQ.ICONST.OR.ISWT.EQ.0
      LOGI=LOGI.OR.LOGJ
      LOGIJ=ISWT.EQ.0.AND.KFEASJ.EQ.1
      RETURN
      END
      SUBROUTINE CATCH(FC)
C
C*    STORES INPUT GIVING LOWEST CONSTRAINT VALUE
C
      INTEGER TNINPT
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /CATC/ ICATCH,CFCP(50),CT(50)
      IF(KAM.EQ.1)THEN
      CFC=1.0E100
      ICATCH=0
      RETURN
      ENDIF
      IF(FC.LT.CFC)THEN
      DO 1 J=1,TNINPT
      CFCP(J)=FCP(NINPT(J))
      CT(J)=T(NINPT(J))
    1 CONTINUE
      CFC=FC
      ICATCH=0
      ELSE
      IF(ABS(FC-CFC).GT.1.0E-04)ICATCH=1
      ENDIF
      RETURN
      END
      SUBROUTINE MASBL
C
C*    SUBROUTINE ASSIGNS THE CORRECT FCP VALUES TO STREAMS WHERE
C*    POSSIBLE.STREAMS WITH UNKNOWN FCP'S (IE.THOSE IN STREAM SPLITS)
C*    ARE IDENTIFIED AND ASSIGNED -VE FCP'S.
C*    THE UNIT MATRIX IS CHECKED FOR CONSISTENCY.
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /MASBLC/ IFLAG,TNUNKF,NUNKF(50)
      COMMON /TMPROC/ T(100),FCP(100)
      DO 13 J=1,TNSTR
      DO 14 K=1,TNINPT
      IF(J.EQ.NINPT(K))GO TO 13
   14 CONTINUE
      FCP(J)=0.0
   13 CONTINUE
      IFLAG=0
      DO 3 J=1,TNINPT
      K=NINPT(J)-1
    7 K=K+1
      I=1
    4 IF(MATRXU(I,2).EQ.K.OR.MATRXU(I,3).EQ.K)THEN
      IF(MATRXU(I,3).GT.0.AND.MATRXU(I,5).GT.0)THEN
```

```
      IF (TNMIX.GT.0) THEN
      DO 12 J=1,TNMIX
      IF (NTYPE (MATRXU (NMIX (J) ,4)) .EQ.8.OR.NTYPE (MATRXU (NMIX (J) ,4)) .EQ.
     +10) THEN
C
C*    ENERGY BALANCE ON MIXERS FROM WHICH A TARGET TEMP. MUST BE MET
C
      F ( (3*TNHXER) +I) = (FCP (MATRXU (NMIX (J) ,4)) *T (MATRXU (NMIX (J) ,4)) )
     +- (FCP (MATRXU (NMIX (J) ,2)) *T (MATRXU (NMIX (J) ,2)) ) - (FCP (MATRXU (
     +NMIX (J) ,3)) *T (MATRXU (NMIX (J) ,3)) )
      I=I+1
      ENDIF
   12 CONTINUE
      ENDIF
      IF (INITRN.EQ.3) THEN
      INITRN=4
      DO 13 J=1,NBROY
      BOUND (J) =-1.0E100
      BOUND (NBROY+J) =1.0E100
   13 CONTINUE
      KALL=0
      NB=2*NBROY
      NW=NBROY* (NBROY+3)
      IF (IPRNT.EQ.4) THEN
      JPRNT=2
      ELSE
      JPRNT=-1
      ENDIF
      ENDIF
      IF (KALL.GT.0.AND.KOND.EQ.0) KALL=0
      KALL=KALL+1
C
C*    CALL EQUATION SOLVING ROUTINE BROYDN (FROM CELIB)
C
      CALL BROYDN (NBROY,X,F,ACCUR,BOUND,NB,W,NW,KOND,KALL,JPRNT)
C
C*    CONVERT X VALUES FROM BROYDN TO PHYSICAL EQUIVALENTS
C
      IF (TNSPLT.GT.0) THEN
      DO 2 J=1,TNSPLT
      SPLFRN (J) =X (J)
    2 CONTINUE
      ENDIF
      DO 15 J=1,TNHXER
      HXL (J) =X (TNSPLT+J)
   15 CONTINUE
      I=1
      DO 16 J=1,TNSTR
      IF (NTYPE (J) .EQ.2) THEN
      T (J) =X (TNSPLT+TNHXER+I)
      I=I+1
      ENDIF
   16 CONTINUE
      IF (KALL.GT.1000) IFLEE=1
      IF (KOND.EQ.1.AND.IFLEE.EQ.0) GO TO 17
      IF (IFLEE.EQ.1) WRITE (6,*) ('XXXXX  ITERATIONS IN BROYDN EXCEEDED SET
     + NO.  XXXXX')
      IF (KOND.NE.0) THEN
      IFLEE=1
      IF (KOND.EQ.-4) WRITE (6,*) ('XXXXX  DIMENSIONS OF W OR BOUND TOO SMAL
     +L  XXXXX')
      IF (KOND.EQ.-3) WRITE (6,*) ('XXXXX  EQUATIONS MAY NOT BE INDEPENDENT
     + XXXXX')
      IF (KOND.EQ.-1) WRITE (6,*) ('XXXXX  BOUNDS ARE IMPEDING PROGRESS  XXX
     +XX')
      ENDIF
```

```
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /STARTC/ FST(50),TST(50)
      LEST=0
      IF(INTES.EQ.1)THEN
      WRITE(8,*)('INPUT STARTING POINT IDENTIFIER 1=LOWER BOUND,')
      WRITE(8,*)('2=UPPER BOUND,3=MID WAY,4=INPUT VALUES,')
      IF(KNT.GT.0)WRITE(8,*)('5=CURRENT POINT')
      READ(7,*)IST
      ELSE
      IST=3
      ENDIF
      IF(IST.EQ.1)THEN
      DO 7 J=1,TNINPT
      FST(J)=FCPR(J,1)+(FCPR(J,2)-FCPR(J,1))
      TST(J)=TSR(J,1)+(TSR(J,2)-TSR(J,1))
    7 CONTINUE
      ELSE IF(IST.EQ.2)THEN
      DO 8 J=1,TNINPT
      FST(J)=FCPR(J,2)-(FCPR(J,2)-FCPR(J,1))
      TST(J)=TSR(J,2)-(TSR(J,2)-TSR(J,1))
    8 CONTINUE
      ELSE IF(IST.EQ.3)THEN
      DO 9 J=1,TNINPT
      FST(J)=(FCPR(J,1)+FCPR(J,2))/2.0
      TST(J)=(TSR(J,1)+TSR(J,2))/2.0
    9 CONTINUE
      ELSE IF(IST.EQ.4)THEN
      DO 10 J=1,TNINPT
      WRITE(8,8000)
 8000 FORMAT(' INPUT (FCP,TEMP) FOR INLET OF STREAM',I4)
      READ(7,*)FST(J),TST(J)
   10 CONTINUE
      ELSE IF(IST.EQ.5.AND.KNT.GT.0)THEN
      DO 11 J=1,TNINPT
      FST(J)=FCP(NINPT(J))
      TST(J)=T(NINPT(J))
   11 CONTINUE
      ELSE
      WRITE(6,*)('XXXXX  ERROR IN START  XXXXX')
      LEST=1
      ENDIF
      RETURN
      END
      SUBROUTINE TMPRO
C
C*    ASSIGNS CORRECT VALUES TO INLET AND OUTLET STREAMS
C
      INTEGER TNINPT,TNSTR
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /STARTC/ FST(50),TST(50)
      NINPT(TNINPT+1)=TNSTR+1
      DO 7 J=1,TNINPT
      T(NINPT(J))=TST(J)
      T(NINPT(J+1)-1)=TT(J)
      FCP(NINPT(J))=FST(J)
    7 CONTINUE
      RETURN
      END
      SUBROUTINE FUNCT1(N,XC,FC)
C
C*    EQUATION SOLVING ROUTINE
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
```

```
      IF (MATRXU(I,3) .EQ.K) THEN
      FCP (MATRXU(I,5)) =FCP (MATRXU(I,3))
      ELSE
      FCP (MATRXU(I,4)) =FCP (MATRXU(I,2))
      ENDIF
      ELSE IF (MATRXU(I,3) .EQ.0.AND.MATRXU(I,5) .EQ.0) THEN
      FCP (MATRXU(I,4)) =FCP (MATRXU(I,2))
      ELSE IF (MATRXU(I,3) .EQ.0) THEN
      FCP (MATRXU(I,4)) =-ABS (FCP (MATRXU(I,2)))
      FCP (MATRXU(I,5)) =-ABS (FCP (MATRXU(I,2)))
      ELSE IF (MATRXU(I,5) .EQ.0) THEN
      FCP (MATRXU(I,2)) =-ABS (FCP (MATRXU(I,2)))
      FCP (MATRXU(I,3)) =-ABS (FCP (MATRXU(I,3)))
       IF (FCP (MATRXU(I,4)) .GT.-0.001.AND.FCP (MATRXU(I,4)) .LT.0.001) THEN
      FCP (MATRXU(I,4)) =ABS (FCP (MATRXU(I,3)))
      ENDIF
      ELSE
         GO TO 11
         ENDIF
         GO TO 7
         ENDIF
         IF (I.NE.TNUNTS) THEN
         I=I+1
         GO TO 4
         ENDIF
         IF (K.NE. (NINPT(J+1)-1)) GO TO 11
       3 CONTINUE
         DO 12 L=1,TNSTR
         IF (FCP(L) .LT.0.001.AND.FCP(L) .GT.-0.001) GO TO 11
      12 CONTINUE
         TNUNKF=0
         DO 15 K=1,TNSTR
         IF (FCP(K) .LT.0.0) THEN
         TNUNKF=TNUNKF+1
         NUNKF (TNUNKF) =K
         ENDIF
      15 CONTINUE
         RETURN
      11 WRITE (6,*) ('XXXXX ERROR IN UNIT MATRIX  XXXXX')
         IFLAG=1
         RETURN
         END
         SUBROUTINE COUNT
C
C*       SUBROUTINE IDENTIFIES THE UNITS INVOLVED IN THE PROBLEM
C
         INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
        +TNDF
         COMMON /INPUTC/ FCPR(50,2) ,TSR(50,2) ,TT(50) ,TNSTR,TNUNTS,TNINPT,
      +MATRXU(50,5) ,NINPT(50) ,U(50) ,A(50) ,DELTMN,PRCENT,IP
      +, IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
         COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50) ,NMIX(20) ,
      +NSPLT(20) ,NUTIL(50) ,IFLAF
         TNHXER=TNUTIL=TNMIX=TNSPLT=IFLAF=0
         DO 1 I=1,TNUNTS
         IF (MATRXU(I,3) .GT.0.AND.MATRXU(I,5) .GT.0) THEN
         IF (MATRXU(I,2) .GT.0.AND.MATRXU(I,4) .GT.0) THEN
         TNHXER=TNHXER+1
         NHXER (TNHXER) =I
         ELSE
         GO TO 10
         ENDIF
         ELSE IF (MATRXU(I,3) .EQ.0.AND.MATRXU(I,5) .EQ.0) THEN
         TNUTIL=TNUTIL+1
         NUTIL (TNUTIL) =I
         ELSE IF (MATRXU(I,3) .EQ.0) THEN
```

```
   29 RETURN
      END
      SUBROUTINE XSET
C
C*    SUBROUTINE IDENTIFIES THE VARIABLES INVOLVED IN THE PROBLEM
C*    AND TRANSFERS THEIR VALUES TO A VECTOR X FOR USE IN BROYDN.
C*    CHECKS ARE MADE FOR DEGREES OF FREEDOM > 0 AND FOR NO.
C*    OF VARIABLES = NO. EQUATIONS
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
C     CHARACTER IQUEST
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /TMPROC/ T(100),FCP(100)
      INITRN=3
      KSKIP=0
C
C*    CALCULATE EFFECTIVE DEGREES OF FREEDOM
C
      TNDF=TNUTIL+TNSPLT-TNINPT-IRDF
      IF (TNDF.LT.0) GO TO 30
      IF (TNDF.GT.0) THEN
      WRITE(6,*) ('XXXXX DF MUST = 0 IN CURRENT VERSION XXXXX')
      GO TO 27
      ENDIF
      IF (TNSPLT.GT.0) THEN
      DO 6 J=1,TNSPLT
      X(J)=SPLFRN(J)
    6 CONTINUE
      ENDIF
      IF (KSKIP.GT.1) GO TO 29
      DO 9 J=1,TNHXER
      X(TNSPLT+J)=HXL(J)
    9 CONTINUE
      I=1
      DO 10 J=1,TNSTR
      IF (NTYPE(J).EQ.2) THEN
      X(TNSPLT+TNHXER+I)=T(J)
      I=I+1
      ENDIF
   10 CONTINUE
      NBROY=TNSPLT+TNHXER+I-1
      IK=0
      DO 11 J=1,TNSTR
      IF (NTYPE(J).EQ.8.OR.NTYPE(J).EQ.10) IK=IK+1
   11 CONTINUE
      NEQ=(3*TNHXER)+IK
      IF (NEQ.NE.NBROY) THEN
      WRITE(6,*) ('XXXXX  FAILURE IN XSET , NO. VARIABLES N.E. NO. EQNS.
     + XXXXX')
      IFLAX=1
      ENDIF
      GO TO 29
   30 WRITE(6,*) ('XXXXX  DEGREES FREEDOM L.T. ZERO ,NO SOLUTION POSSIBLE
     +  XXXXX')
   27 IFLAX=1
   29 RETURN
      END
```

```
+TNDF
 COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
+MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
+,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
 COMMON /MASBLC/ IFLAG,TNUNKF,NUNKF(50)
 COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
+NSPLT(20),NUTIL(50),IFLAF
 COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
 COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
 COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
 COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
 COMMON /TMPROC/ T(100),FCP(100)
 COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
 COMMON /SETC/ ABL(50),ABU(50)
 COMMON /CON1C/ CC(200),KFEAS(200),IFEAS
 DIMENSION XC(N)
   KAM=KAM+1
   IF(IPRN2.GT.0)WRITE(6,60)
60 FORMAT(1H /1H ,50('-'),1H /)
   IF(IPRN2.GE.1)THEN
   WRITE(6,61)KAM
61 FORMAT(' CALLING EQUQTION SOLVER ON ITERATION',I6)
   ENDIF
   N1=1
   DO 1 J=1,2*TNINPT
   IF(J.LE.TNINPT.AND.IRDIF(J).EQ.1)THEN
   FCP(NINPT(J))=(XC(N1)*(ABU(N1)-ABL(N1))/2.0)+((ABL(N1)+ABU(N1)
  +)/2.0)
   N1=N1+1
   ENDIF
    IF(J.GT.TNINPT.AND.IRDIF(J).EQ.1)THEN
   T(NINPT(J-TNINPT))=(XC(N1)*(ABU(N1)-ABL(N1))/2.0)+
  +((ABL(N1)+ABU(N1))/2.0)
   N1=N1+1
   ENDIF
1 CONTINUE
   IF(IPRN2.GE.1)THEN
   CALL WRYTE
   ENDIF
   CALL MASBL
   IF(IFLAG.EQ.1)GO TO 99
   CALL MXUTIL
   IF(KAM.EQ.1)THEN
   CALL TYPE
   ENDIF
   ACCUR=1.0E-07
   CALL EQSOL(ACCUR)
   IF(IPRN2.GE.1)WRITE(6,*)('LEAVING EQUATION SOLVER')
   IF(IFLAX.EQ.1)GO TO 99
   IF(IFLEE.EQ.1)GO TO 99
   CALL CON1
   FC=CC(ICONST)
   CALL CATCH(FC)
   IF(IPRN2.EQ.1)THEN
   ISWT=1
   CALL FEAS(ISWT)
   IF(KAM.EQ.1)THEN
   CALL RITE
   ISWT=0
   CALL FEAS(ISWT)
   ENDIF
   ENDIF
   IF(IPRN2.EQ.2)THEN
   ISWT=1
   CALL FEAS(ISWT)
   CALL RITE
```

```
      TNSPLT=TNSPLT+1
      NSPLT(TNSPLT)=I
      ELSE IF(MATRXU(I,5).EQ.0)THEN
      TNMIX=TNMIX+1
      NMIX(TNMIX)=I
      ELSE
      GO TO 10
      ENDIF
    1 CONTINUE
      RETURN
   10 WRITE(6,*)('XXXXX  FAILURE IN COMNT  XXXXX')
      IFLAF=1
      RETURN
      END
      SUBROUTINE GUESS
C
C*    SUBROUTINE PROVIDES AN INITIAL GUESS OF SPLIT FRACTIONS,
C*    INTERMEDIATE STREAM TEMPS. ETC. (NEEDED ONCE ONLY DURING ANY
C*    RUN)
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      CHARACTER IQUEST
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /COMNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /TMPROC/ T(100),FCP(100)
      IF(INITRN.EQ.1)THEN
      INITRN=2
      DO 3 J=1,TNHXER
      HXL(J)=(T(MATRXU(NHXER(J),2))-T(MATRXU(NHXER(J),4)))
     +*FCP(MATRXU(NHXER(J),2))
    3 CONTINUE
      ENDIF
      IF(INITRN.EQ.0)THEN
      INITRN=1
      IQUEST='Y'
      KFLAG=0
      DO 7 I=1,TNINPT
      IF((NINPT(I+1)-1).GT.(NINPT(I)+1))THEN
      DELINT=((T(NINPT(I+1)-1)-T(NINPT(I)))/((NINPT(I+1)-1)
     +-NINPT(I)))
      DO 8 J=(NINPT(I)+1),(NINPT(I+1)-2)
      IF(INTER.EQ.1.AND.IQUEST.EQ.'Y')THEN
      IF(KFLAG.EQ.0)THEN
      WRITE(8,*)('DO YOU WANT TO INPUT THE ')
      WRITE(8,*)('INTERMEDIATE STREAM TEMPS ?  (Y OR N) ')
      READ(7,7000)IQUEST
 7000 FORMAT(A1)
      KFLAG=1
      ENDIF
      IF(IQUEST.EQ.'Y')THEN
      WRITE(8,6000)J
 6000 FORMAT('ESTIMATE FOR STREAM NO.',I4,' ?')
      READ(7,*)T(J)
      ENDIF
      ENDIF
      IF(INTER.EQ.0.OR.IQUEST.NE.'Y')T(J)=T(J-1)+DELINT
    8 CONTINUE
      ENDIF
    7 CONTINUE
      IF(TNSPLT.GT.0)THEN
```

```
      SUBROUTINE DELTA(DELT1,DELT2,DELTAX,IDELT)
C
C*    SUBROUTINE CALCULATES LOG MEAN TEMPERATURE DIFFERENCE WITH
C*    CORRECTIONS FOR DISCONTINUITIES,SINGULARITIES AND UNDEFINED
C*    REGIONS
C
      PDELTA(XX,YY)=(XX-YY)/ALOG(XX/YY)
      VDELTX(XX,YY)=-((ABS(XX)-YY)/ALOG(ABS(XX)/YY))
      VDELTY(XX,YY)=-((XX-ABS(YY))/ALOG(XX/ABS(YY)))
      E=1.0E-10
      IF(DELT1.LT.E.AND.DELT1.GT.-E.AND.DELT2.LT.E.AND.DELT2.GT.-E)THEN
      DELTAX=0.0
      IDELT=1
      ELSE IF(DELT1.LT.E.AND.DELT1.GT.-E)THEN
      IDELT=1
      IF(DELT2.GT.0)THEN
      DELTAX=(((PDELTA(E,DELT2)-VDELTX(-E,DELT2))*(DELT1+E)/
     +(2.0*E))+VDELTX(-E,DELT2))
      ELSE
      DELTAX=(((VDELTY(E,DELT2)+PDELTA(-E,DELT2))*(DELT1+E)/
     +(2.0*E))-PDELTA(-E,DELT2))
      ENDIF
      ELSE IF(DELT2.LT.E.AND.DELT2.GT.-E)THEN
      IDELT=1
      IF(DELT1.GT.0)THEN
      DELTAX=(((PDELTA(DELT1,E)-VDELTY(DELT1,-E))*(DELT2+E)/
     +(2.0*E))+VDELTY(DELT1,-E))
      ELSE
      DELTAX=(((VDELTX(DELT1,E)+PDELTA(DELT1,-E))*(DELT2+E)/
     +(2.0*E))-PDELTA(DELT1,-E))
      ENDIF
      ELSE IF((DELT1/DELT2).LT.(1.0+E).AND.(DELT1/DELT2).GT.(1.0-E))THEN
      DELTAX=((ABS(DELT1+DELT2)/2.0))
      IDELT=1
      ELSE IF(DELT1.GT.0.AND.DELT2.LT.0)THEN
      IDELT=1
      DELTAX=VDELTY(DELT1,DELT2)
      ELSE IF(DELT1.LT.0.AND.DELT2.GT.0)THEN
      IDELT=1
      DELTAX=VDELTX(DELT1,DELT2)
      ELSE IF(DELT1.LT.0.AND.DELT2.LT.0)THEN
      IDELT=1
      DELTAX=(-PDELTA(DELT1,DELT2))
      ELSE
      DELTAX=PDELTA(DELT1,DELT2)
      ENDIF
      RETURN
      END
      SUBROUTINE MXUTIL
C
C*    SUBROUTINE CALCULATES LINNHOFF PROBLEM TABLE
C
      INTEGER TNINPT,TNSTR
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      DIMENSION XX1(50),XX2(50),IS(50),X(100),SIGMA
     +(100),TSTAR(100),SIG(50),FLOW(100),XFIX(100)
      NINPT2=2*TNINPT
      DO 1 J=1,TNINPT
      IF(T(NINPT(J)).GT.T(NINPT(J+1)-1))THEN
      XX1(J)=T(NINPT(J))-DELTMN
      XX2(J)=T(NINPT(J+1)-1)-DELTMN
      IS(J)=1
```

```
ISWT=0
CALL FEAS(ISWT)
ENDIF
IF(IPRN2.EQ.0)THEN
IF(KAM.EQ.1)THEN
CALL WRYTE
ISWT=1
CALL FEAS(ISWT)
ENDIF
ENDIF
ENDIF
 IF(IFEAS.EQ.1)THEN
 IP=IP+1
 IF(IPRN2.GT.0.OR.IP.EQ.1)THEN
WRITE(6,*)('***** INFEASIBLE POINT FOUND *****')
ENDIF
      ENDIF
      IF(IPRN2.GT.0)WRITE(6,60)
      RETURN
   99 IFLAX=1
      WRITE(6,*)('EQUATIONS HAVE NOT BEEN SOLVED ')
      CALL WRYTE
      STOP
      END
      SUBROUTINE SETUP(N,BL,BU,XC,LIW,LW,IFAIL,IBOUND)
C
C*    SETS UP VARIABLES FOR OPTIMIZING ROUTINE
C
      INTEGER TNINPT
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /SETC/ ABL(50),ABU(50)
      DIMENSION BL(N),BU(N),XC(N)
      IBOUND=0
      N1=1
      DO 1 J=1,2*TNINPT
      IF(J.LE.TNINPT.AND.IRDIF(J).EQ.1)THEN
      ABL(N1)=FCPR(J,1)
      ABU(N1)=FCPR(J,2)
      XC(N1)=FCP(NINPT(J))
      N1=N1+1
      ENDIF
      IF(J.GT.TNINPT.AND.IRDIF(J).EQ.1)THEN
      ABL(N1)=TSR(J-TNINPT,1)
      ABU(N1)=TSR(J-TNINPT,2)
      XC(N1)=T(NINPT(J-TNINPT))
      N1=N1+1
      ENDIF
    1 CONTINUE
      DO 2 J=1,N
      XC(J)=(XC(J)-((ABL(J)+ABU(J))/2.0))/((ABU(J)-ABL(J))/2.0)
      BL(J)=-1
      BU(J)=1
    2 CONTINUE
      IF(N.NE.(N1-1))THEN
      WRITE(6,*)('XXXXX  ERROR IN SETUP  XXXXX')
      STOP
      ENDIF
      LIW=N+2
      LW=MAX0((N*(N-1)/2+12)*N),13)
      IFAIL=1
      RETURN
      END
      SUBROUTINE WRYTE
C
```

```
      IQUEST='Y'
      KFLAG=0
      DO 10 J=1,TNSPLT
      IF(INTER.EQ.1.AND.IQUEST.EQ.'Y')THEN
      IF(KFLAG.EQ.0)THEN
      WRITE(8,*)('DO YOU WANT TO INPUT VALUES')
      WRITE(8,*)('FOR SPLIT FRACTIONS ? (Y OR N) ')
      READ(7,7000)IQUEST
      KFLAG=1
      ENDIF
      IF(IQUEST.EQ.'Y')THEN
      WRITE(8,6001)NSPLT(J)
 6001 FORMAT('ESTIMATE OF SPLIT FRACTION UNIT NO.  ',I4,' ?')
      READ(7,*)SPLFRN(J)
      ENDIF
      ENDIF
      IF(INTER.EQ.0.OR.IQUEST.NE.'Y')SPLFRN(J)=0.9
   10 CONTINUE
      ENDIF
      IF(INTER.EQ.1)THEN
      WRITE(8,8000)
 8000 FORMAT(' DO YOU WANT TO INPUT THE MINIMUM HEAT EXCHANGER HEAT LOAD
     +  ',1H /,' WHICH IS TO BE CONSIDERED SIGNIFICANT (DEFAULT 1.0)  ?',1
     +H /1H ,'(Y OR N) ')
      READ(7,7000)IQUEST
      IF(IQUEST.EQ.'Y')THEN
      WRITE(8,*)('MINIMUM HEAT LOAD ?')
      READ(7,*)DELHMN
      ENDIF
      ENDIF
      IF(INTER.EQ.0.OR.IQUEST.NE.'Y')DELHMN=1.0
      ENDIF
      RETURN
      END
      SUBROUTINE TYPE
C
C*    SUBROUTINE ASSIGNS AN IDENTIFIER TO EACH STREAM ACCORDING TO
C*    TYPE.THEY ARE :
C*    1-----AN INPUT STREAM TO THE NETWORK
C*    2-----OUTPUT STREAM FROM A HEAT EXCHANGER (NOT IF TYPE 6)
C*    3-----OUTPUT FROM A SPLITTER
C*    4-----OUTPUT FROM A MIXER (NOT IF TYPE 8)
C*    5-----OUTPUT FROM A UTILITY EXCHANGER
C*    6-----OUTPUT FROM A HEAT EXCHANGER WHICH IS ALSO AN OUTPUT FROM
C*          THE NETWORK
C*    8-----OUTPUT FROM A MIXER WHICH IS ALSO AN OUTPUT FROM THE
C*          NETWORK
C*    9-----INPUT TO A REDUNDANT UTILITY EXCHANGER TREATED AS AN OUTPUT
C*          FROM THE NETWORK (MAY OCCUR IF PROBLEM IS UNPINCHED)
C*   10-----OUTPUT FROM A MIXER WHICH IS ALSO AN INPUT TO A REDUNDANT UTILITY
C*          EXCHANGER
C
      INTEGER TNINPT,TNSTR,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
     +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      DO 7 J=1,TNSTR
      NTYPE(J)=0
      DO 8 JJ=1,TNINPT
      IF(J.EQ.NINPT(JJ))THEN
      NTYPE(J)=1
```

```
      ELSE
      XX1(J)=T(NINPT(J))
      XX2(J)=T(NINPT(J+1)-1)
      IS(J)=-1
      ENDIF
    1 CONTINUE
      DO 2 J=1,TNINPT
      X(J)=XX1(J)
      X(TNINPT+J)=XX2(J)
    2 CONTINUE
      DO 3 K=2,NINPT2
      XVAL=X(K)
      DO 4 L=1,K-1
      IF(XVAL.LT.X(L))THEN
      XFIX(L)=X(L)
      DO 5 M=L+1,K,1
      XFIX(M)=X(M)
      X(M)=XFIX(M-1)
    5 CONTINUE
      X(L)=XVAL
      GO TO 3
      ENDIF
    4 CONTINUE
    3 CONTINUE
      DO 6 K=2,NINPT2
      SIGMA(K-1)=0.0
      TSTAR(K)=(X(K)+X(K-1))/2.0
      IF(ABS(TSTAR(K)).GT.0.001)THEN
      DO 7 J=1,TNINPT
      SIG(J)=0.0
      IF(IS(J).EQ.1.AND.TSTAR(K).GT.(T(NINPT(J+1)-1)-DELTMN).AND.
     +TSTAR(K).LT.(T(NINPT(J))-DELTMN))THEN
      SIG(J)=(X(K)-X(K-1))*FLOAT(IS(J))*FCP(NINPT(J))
      ELSE IF(IS(J).EQ.-1.AND.TSTAR(K).GT.T(NINPT(J)).AND.TSTAR(K).LT.T(
     +NINPT(J+1)-1))THEN
      SIG(J)=(X(K)-X(K-1))*FLOAT(IS(J))*FCP(NINPT(J))
      ENDIF
      SIGMA(K-1)=SIGMA(K-1)+SIG(J)
    7 CONTINUE
      ENDIF
    6 CONTINUE
      FLOW(1)=0.0
      DO 8 J=2,NINPT2
      FLOW(J)=FLOW(J-1)+SIGMA(NINPT2-J+1)
    8 CONTINUE
      HMIN=1.0E50
      DO 9 J=2,NINPT2,2
      HMIN1=AMIN1(FLOW(J),FLOW(J-1))
      IF(HMIN1.LT.HMIN)THEN
      IF(FLOW(J).LT.FLOW(J-1))THEN
      IPINCH=J
      ELSE
      IPINCH=J-1
      ENDIF
      HMIN=HMIN1
      ENDIF
    9 CONTINUE
      IF(ABS(HMIN).LT.0.001)THEN
      TOTAL=FLOW(NINPT2)
      HCOLD=TOTAL
      HHOT=0.0
      MFLAG=1
      ELSE IF(HMIN.GT.0.001)THEN
      TOTAL=FLOW(NINPT2)
      HCOLD=TOTAL
      HHOT=0.0
```

```
C*     PRODUCES OUTPUT OF INLET CONDITIONS
C
       INTEGER TNINPT
       COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
      +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
      +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
       COMMON /TMPROC/ T(100),FCP(100)
       WRITE(6,*)('CURRENT INLET CONDITIONS')
       WRITE(6,62)
       WRITE(6,*)('INLET NO. VARIABLE LOWER BOUND UPPER BOUND ACTUAL VALU
      +E')
       DO 1 J=1,TNINPT
       WRITE(6,60)J,FCPR(J,1),FCPR(J,2),FCP(NINPT(J))
       WRITE(6,61)J,TSR(J,1),TSR(J,2),T(NINPT(J))
     1 CONTINUE
       WRITE(6,62)
       RETURN
    60 FORMAT(5X,I2,7X,'FCP',F11.2,2X,F10.2,4X,F9.2)
    61 FORMAT(5X,I2,8X,'T',1X,F11.2,2X,F10.2,4X,F9.2)
    62 FORMAT(1H /)
       END
       SUBROUTINE RSLTS(IFAIL)
C
C*     INTERPRETS OUTPUT FROM OPTIMIZER
C
       WRITE(6,60)
    60 FORMAT(1H /1H ,50('+'),1H /)
       IF(IFAIL.EQ.1)THEN
       WRITE(6,*)('PARAMETER OUTSIDE EXPECTED RANGE')
       ELSE IF(IFAIL.EQ.2)THEN
       WRITE(6,*)('ALGORITHM DOES NOT SEEM TO BE CONVERGING')
       ELSE IF(IFAIL.EQ.3)THEN
       WRITE(6,*)('ALGORITHM FAILS TO FIND A LOWER POINT OR A MINIMUM')
       ELSE IF(IFAIL.EQ.4)THEN
       WRITE(6,*)('OVERFLOW . RESTART FROM CURRENT POINT')
       ELSE IF(IFAIL.EQ.5)THEN
       WRITE(6,*)('PROBABLE THAT A MINIMUM HAS BEEN FOUND')
       ELSE IF(IFAIL.EQ.6)THEN
       WRITE(6,*)('POSSIBLE THAT A MINIMUM HAS BEEN FOUND')
       ELSE IF(IFAIL.EQ.7)THEN
       WRITE(6,*)('UNLIKELY THAT A MINIMUM HAS BEEN FOUND')
       ELSE IF(IFAIL.EQ.8)THEN
       WRITE(6,*)('VERY UNLIKELY THAT A MINIMUM HAS BEEN FOUND')
       ELSE IF(IFAIL.EQ.9)THEN
       WRITE(6,*)('MODULUS OF A VARIABLE > 1.0E06')
       ELSE IF(IFAIL.EQ.0)THEN
       WRITE(6,*)('A MINIMUM HAS BEEN FOUND')
       ELSE
       WRITE(6,*)('XXXXX  IFAIL TAKES AN UNEXPECTED VALUE  XXXXX')
       ENDIF
       WRITE(6,60)
       RETURN
       END
       SUBROUTINE CON1
C
C*     GENERATES CONSTRAINT VALUES
C
       INTEGER TNHXER,TNSPLT
       COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
      +MATRXU(50,5),NINPT(50),U(50),A(50),DELTMN,PRCENT,IP
      +,IPRNT,INTER,IPRN2,INTES,KAM,SN,KNT,ICONST,IRDIF(50)
       COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
      +NSPLT(20),NUTIL(50),IFLAF
       COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
       COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
       COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
```

```
      GO TO 7
      ENDIF
    8 CONTINUE
      DO 9 JJ=1,TNHXER
      IF (J.EQ.MATRXU(NHXER(JJ),4).OR.J.EQ.MATRXU(NHXER(JJ),5)) THEN
      NTYPE(J)=2
      GO TO 7
      ENDIF
    9 CONTINUE
      IF (TNSPLT.GT.0) THEN
      DO 10 JJ=1,TNSPLT
      IF (J.EQ.MATRXU(NSPLT(JJ),4).OR.J.EQ.MATRXU(NSPLT(JJ),5)) THEN
      NTYPE(J)=3
      GO TO 7
      ENDIF
   10 CONTINUE
      ENDIF
      IF (TNMIX.GT.0) THEN
      DO 11 JJ=1,TNMIX
      IF (J.EQ.MATRXU(NMIX(JJ),4)) THEN
      NTYPE(J)=4
      GO TO 7
      ENDIF
   11 CONTINUE
      ENDIF
      DO 12 JJ=1,TNUTIL
      IF (J.EQ.MATRXU(NUTIL(JJ),4)) THEN
      NTYPE(J)=5
      ENDIF
   12 CONTINUE
    7 CONTINUE
      DO 13 J=1,TNSTR
      DO 14 JJ=1,TNINPT
      IF (J.EQ.(NINPT(JJ+1)-1).AND.NTYPE(J).EQ.2) THEN
      NTYPE(J)=6
      ENDIF
      IF (J.EQ.(NINPT(JJ+1)-1).AND.NTYPE(J).EQ.4) THEN
      NTYPE(J)=8
      ENDIF
   14 CONTINUE
   13 CONTINUE
      DO 15 J=1,TNINPT
      IF (T(NINPT(J+1)-1).LT.T(NINPT(J))) THEN
      NHOT=NINPT(J+1)-1
      ELSE
      GO TO 16
      ENDIF
   15 CONTINUE
C
C*    REMOVE REDUNDANT UTILITY EXCHANGERS FROM NETWORK
C
   16 IRDF=0
      LINN=0
      IF (IPINCH.EQ.0.AND.LINN.EQ.1) THEN
      DO 17 J=1,TNUTIL
      IF (NHOT.GT.HCOLD) THEN
      IF (MATRXU(NUTIL(J),4).LE.NHOT) THEN
      IF (NTYPE(MATRXU(NUTIL(J),2)).NE.4) THEN
      NTYPE(MATRXU(NUTIL(J),2))=9
      ELSE
      NTYPE(MATRXU(NUTIL(J),2))=10
      ENDIF
      IRDF=IRDF+1
      ENDIF
      ELSE
      IF (MATRXU(NUTIL(J),4).GT.NHOT) THEN
```

```
      MFLAG=2
      ELSE  IF (HMIN.LT.-0.001) THEN
      TOTAL=FLOW (NINPT2) + (2.0*ABS (HMIN) )
      MFLAG=3
      HHOT=ABS (HMIN)
      HCOLD=TOTAL-HHOT
      ENDIF
       IF (ABS (HHOT) .GT.0.001.AND.ABS (HCOLD) .GT.0.001) THEN
      TPINCH=X (NINPT2+1-IPINCH)
       IPINCH=1
      ELSE
       IPINCH=0
      ENDIF
       RETURN
       END
       SUBROUTINE RITE
C
C*     SUBROUTINE PRODUCES OUTPUT OF NETWORK VARIABLES
C
       INTEGER TNSTR,TNHXER,TNUTIL,TNSPLT
       COMMON /INPUTC/ FCPR (50,2) ,TSR (50,2) ,TT (50) ,TNSTR,TNUNTS,TNINPT,
     +MATRXU (50,5) ,NINPT (50) ,U (50) ,A (50) ,DELTMN,PRCENT,IP
     +, IPRNT, INTER, IPRN2, INTES, KAM, SN, KNT, ICONST, IRDIF (50)
       COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER (50) ,NMIX (20) ,
     +NSPLT (20) ,NUTIL (50) , IFLAF
       COMMON /GUESSC/ HXL (50) ,SPLFRN (20) ,DELHMN
       COMMON /EQSOLC/ IFLEE,HUTIL (50) , INITRN,KALL,KOUNT,TUTIL
       COMMON /TMPROC/ T (100) ,FCP (100)
       COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
       WRITE (6,62)
       WRITE (6,63) DELTMN
       IF (IPINCH.EQ.1) THEN
       WRITE (6,64)
       ELSE
       WRITE (6,*) ('PROBLEM IS UNPINCHED')
       ENDIF
       WRITE (6,65) HHOT
       WRITE (6,66) HCOLD
       WRITE (6,67) TOTAL
       WRITE (6,62)
       IF (IFLEE.EQ.1) THEN
       WRITE (6,*) ('VALUES AT TIME OF FAILURE ARE')
       ELSE
       WRITE (6,*) ('BROYDN HAS CONVERGED TO REQUIRED ACCURACY')
       ENDIF
       WRITE (6,82)
       WRITE (6,79) KALL
       WRITE (6,80)
       DO 8 J=1,TNSTR
       WRITE (6,81) J,T (J) ,FCP (J)
     8 CONTINUE
       WRITE (6,82)
       WRITE (6,83)
       DO 9 J=1,TNHXER
       IF (ABS (HXL (J) ) .LT.DELHMN) THEN
       WRITE (6,90) J,MATRXU (NHXER (J) ,1) ,U (J) ,A (J)
       ELSE
       WRITE (6,84) J,MATRXU (NHXER (J) ,1) ,HXL (J) ,U (J) ,A (J)
       ENDIF
     9 CONTINUE
       WRITE (6,82)
       WRITE (6,85)
       DO 10 J=1,TNUTIL
       WRITE (6,86) J,MATRXU (NUTIL (J) ,1) ,HUTIL (J)
    10 CONTINUE
       WRITE (6,82)
```

```
      PROGRAM HERON(INPUS,OUT,INPUT,OUTPUT,TAPE5=INPUS,
     +TAPE6=OUT,TAPE7=INPUT,TAPE8=OUTPUT)
C*
C*    PROGRAM HERON. MODELS THE OPERATION OF A HEAT EXCHANGER NETWORK AT
C*    CORNERPOINTS OF AN OPERATING RANGE. ABLE TO MODEL OPTIMIZING CONTROL.
C*
C*    LIBRARY ROUTINES CALLED: E04UAF FROM NAG LIBRARY
C*                             BROYDN UNPUBLISHED ROUTINE SEE PERKINS(1979)
C*
C*    USE: BHERON(DATA,OUT,INPUT,OUTPUT)
C*         DATA-PROBLEM DATA
C*         OUT-RESULTS
C*         INPUT-LEVEL OF OUTPUT REQUIRED (0 TO 4) OR 9 FOR INTERACTIVE USE
C*         OUTPUT-PROMPTS FOR INTERACTIVE USE
C*
C*    ----------
C*
C*    PROBLEM DATA REQUIRED:
C*
C*    1) TOTAL NUMBER OF UNITS(TNUNTS) , TOTAL NUMBER OF SUBSTREAMS(TNSTR)
C*
C*    2) UNIT MATRIX(MATRXU(TNUNTS,5)). FOR EACH UNIT IN TURN:
C*
C*    UNIT NUMBER , INPUT STREAM 1 , INPUT 2 , OUTPUT 1 , OUTPUT 2
C*
C*    3) TOTAL NUMBER OF PROCESS STREAMS(TNINPT)
C*
C*    4) FOR EACH PROCESS STREAM INPUT IN TURN:
C*
C*    SUBSTREAM NUMBER , LOWER BOUND F.CP , UPPER BOUND F.CP , LOWER BOUND T ,
C*    UPPER BOUND T
C*
C*    5) TARGET T PROCESS STREAM 1 , ... , TARGET T PROCESS STREAM TNINPT
C*
C*    6) MIN PHYSICALL POSSIBLE TEMP. APPROACH(DELTMN) , MAXIMUM ALLOWABLE
C*    UTILITY USE AS A PERCENTAGE > THEORETICAL MIN.
C*
C*    7) U FOR EXCHANGER 1 , ... , U FOR EXCHANGER TNHXER
C*
C*    8) A FOR EXCHANGER 1 , ... , A FOR EXCHANGER TNHXER
C*
C*    9) NUMBER OF CORNERPOINT TO START SEARCH FROM , NUMBER OF CORNERPOINT TO
C*    END SEARCH ON (THIS VALUE CAN BE GREATER THAN THE TOTAL NUMBER OF
C*    CORNERPOINTS IN WHICH CASE THE PROGRAM AUTOMATICALLY TERMINATES
C*    WHEN THE CURRENT CPT. IS > TOTAL NO.)
C*
C*    ----------
C*
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /MASBLC/ IFLAG,TNUNKF,NUNKF(50)
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /PERMC/ IPERM1(50),IPERM2(50),IFLAP
      COMMON /PER1C/ KPERM(50),KPERM1(50)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
```

```
      ENDIF
   17 CONTINUE
      ENDIF
      RETURN
      END
      SUBROUTINE EQSOL(ACCUR)
C
C*    SUBROUTINE SOLVES THE SET OF DESCRIBING EQUATIONS FOR THE
C*    NETWORK
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /MASBLC/ IFLAG,TNUNKF,NUNKF(50)
      COMMON /COWNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /PER1C/ KPERM(50),KPERM1(50)
      DIMENSION F(200),BOUND(150),W(2500)
      IFLEE=IDELT=KOUNT=KOND=0
      IF(INITRN.EQ.0)CALL GUESS
C
C*    REMOVE REDUNDANT UTILITY EXCHANGERS FROM NETWORK
C
      IF(IRDF.GT.0)THEN
      DO 20 J=1,TNUTIL
      IF(NTYPE(MATRXU(NUTIL(J),2)).EQ.9.OR.NTYPE(MATRXU(NUTIL(J),2)).EQ.
     +10)THEN
      T(MATRXU(NUTIL(J),2))=T(MATRXU(NUTIL(J),4))
      ENDIF
   20 CONTINUE
      ENDIF
C
C*    SET TEMPS. EQUAL AROUND SPLITTERS AND CALCULATE OUTPUT FCP'S
C
   17 IF(TNSPLT.GT.0)THEN
      DO 7 J=1,TNSPLT
      T(MATRXU(NSPLT(J),4))=T(MATRXU(NSPLT(J),2))
      T(MATRXU(NSPLT(J),5))=T(MATRXU(NSPLT(J),2))
      FCP(MATRXU(NSPLT(J),4))=FCP(MATRXU(NSPLT(J),2))*SPLFRN(J)
      FCP(MATRXU(NSPLT(J),5))=FCP(MATRXU(NSPLT(J),2))*(1-SPLFRN(J))
    7 CONTINUE
      ENDIF
C
C*    SET ALL REMAINING UNKNOWN FCP'S TO CURRENT CORRECT VALUES
C
      DO 8 J=1,TNHXER
      DO 4 JJ=1,TNUNKF
      IF(MATRXU(NHXER(J),4).EQ.NUNKF(JJ).OR.MATRXU(NHXER(J),5).EQ.
     +NUNKF(JJ))THEN
      IF(MATRXU(NHXER(J),4).EQ.NUNKF(JJ))THEN
      FCP(MATRXU(NHXER(J),4))=FCP(MATRXU(NHXER(J),2))
      ELSE
      FCP(MATRXU(NHXER(J),5))=FCP(MATRXU(NHXER(J),3))
      ENDIF
      ENDIF
    4 CONTINUE
    8 CONTINUE
C
C*    WHERE POSSIBLE COMPLETE M&E BALANCES AROUND MIXERS
```

```
C*      SUBROUTINE GENERATES ALL THE INEQUALITY CONSTRAINTS WHICH MUST
C*      BE SATISFIED FOR FEASIBILITY (INFEASIBLE CONSTRAINTS TAKE -VE
C*      VALUES) AND THE RANGE CONSTRAINTS ON THE SPLIT FRACTIONS
C*      (CONSTRAINTS ARE SCALED AS REQUIRED BY OPTIMIZING ROUTINE)
C
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /COWNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /PER1C/ KPERM(50),KPERM1(50)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      COMMON /FEASC/ IFEAS
      COMMON /CON1C/ HHUTIL,HCUTIL
      INTEGER TNHXER,TNDF,TNSPLT,TNUTIL
      DIMENSION XC(N),CC(M)
C
C*      ACTUAL UTILITY USAGE > THEORETICAL MINIMUM USAGE CALCULATED AT
C*      THE MINIMUM PHYSICALLY POSSIBLE DELTA T * A USER SUPPLIED
C*      PERCENTAGE DEFINING THE MINIMUM ACCEPTABLE DEVIATION FROM
C*      THEORETICAL MINIMUM USAGE.APPLIES TO PINCHED PROBLEMS.
C*      (CONSTRAINT VALUE SCALED TO BE APPROX. IN RANGE -1 TO 1)
C
      IF(IPINCH.EQ.1.OR.IPINCH.EQ.0.AND.LINN.EQ.0)THEN
      CC((3*TNHXER)+TNSPLT-TNDF+1)=(TOTAL*(1+PRCENT/100.0)-TUTIL)/TOTAL
      KN=0
      ELSE
C
C*      FOR UNPINCHED PROBLEMS THEN EITHER HOT OR COLD UTILITY USAGE
C*      MUST EQUAL THE MINIMUM THEORETICAL USAGE.
C*      IF LINN=1
C
      KN=1
      HHUTIL=HCUTIL=0
      IF(HHOT.GT.HCOLD)THEN
      DO 6 J=1,TNUTIL
      IF(MATRXU(NUTIL(J),4).GT.NHOT)THEN
      HHUTIL=HUTIL(J)+HHUTIL
      ENDIF
    6 CONTINUE
      CC(1)=(HHOT-HHUTIL)/100. 0
      ELSE
      DO 7 J=1,TNUTIL
      IF(MATRXU(NUTIL(J),4).LE.NHOT)THEN
      HCUTIL=HUTIL(J)+HCUTIL
      ENDIF
    7 CONTINUE
      CC(1)=(HCOLD+HCUTIL)/100.0
      ENDIF
      ENDIF
      DO 1 J=KN+1,(TNHXER+KN)
C
C*      ALL HEAT EXCHANGER HEAT LOADS MUST BE > 0 (I.E. HOT STREAMS HAVE
C*      THEIR TEMP. REDUCED WHEN PASSING THROUGH AN EXCHANGER ETC.)
C*      (IF FCP VALUES PROVIDED ARE SCALED TO RANGE FROM 1 TO 10 THE
C*      SCALING FACTOR OF 100.0 WILL PRODUCE CONSTRAINT VALUES APPROX IN
C*      RANGE 0 TO 10)
C
      IF(ABS(HXL(J-KN)).GE.DELHMN)THEN
      CC(J)=HXL(J-KN)/100.0
      ELSE
```

```
      IF (CC (LCLU) .LT.0.0.OR.CC (LCLU) .GT. (PRCENT/100.0)) THEN
      WRITE (6,59)
      ENDIF
      IF (CC (LCLU) .LT. (PRCENT/100.0)) THEN
      WRITE (6,51) LCLU,CC (LCLU) , ((TUTIL-TOTAL) /TOTAL) *100.0
      ELSE
      WRITE (6,64) LCLU,CC (LCLU) ,TOTAL-TUTIL
      ENDIF
      ELSE
      KN=1
      WRITE (6,61)
      IF (CC (KN) .GT.0.0001.OR.CC (KN) .LT.-0.0001) WRITE (6,59)
      IF (HHOT.GT.HCOLD) THEN
      WRITE (6,62) KN,CC (KN) ,HHUTIL,HHOT
      ELSE
      WRITE (6,63) KN,CC (KN) ,-HCUTIL,HCOLD
      ENDIF
      ENDIF
      DO 1 J=KN+1, (TNHXER+KN)
      IF (CC (J) .LT.0) THEN
      WRITE (6,59)
      ENDIF
      IF (ABS (CC (J) *100.0) .LT.DELHMN) WRITE (6,66)
      WRITE (6,52) J,NHXER (J-KN) ,CC (J) ,HXL (J-KN)
    1 CONTINUE
      DO 2 J= (TNHXER+KN+1) , (3*TNHXER+KN) ,2
      IREP= (J-TNHXER-KN+1) /2
      IF (CC (J) .LT.0) THEN
      WRITE (6,59)
      ENDIF
      IF (ABS (HXL (IREP)) .LT.DELHMN) THEN
      WRITE (6,65)
      ENDIF
      WRITE (6,53) J,NHXER (IREP) ,CC (J) , (T (MATRXU (NHXER (IREP) ,2)) -T (MATRXU (
     +NHXER (IREP) ,5)))
      IF (CC (J+1) .LT.0) THEN
      WRITE (6,59)
      ENDIF
      IF (ABS (HXL (IREP)) .LT.DELHMN) THEN
      WRITE (6,65)
      ENDIF
      WRITE (6,54) (J+1) ,NHXER (IREP) ,CC (J+1) , (T (MATRXU (NHXER (IREP) ,4)) -T (M
     +ATRXU (NHXER (IREP) ,3)))
    2 CONTINUE
      IF (TNDF.GT.0) THEN
      WRITE (6,55)
      DO 3 J=1,TNDF
      IF (SPLFRN (KPERM (J)) .LT.0.OR.SPLFRN (KPERM (J)) .GT.1) THEN
      WRITE (6,59)
      ENDIF
      WRITE (6,56) NSPLT (KPERM (J)) ,SPLFRN (KPERM (J))
    3 CONTINUE
      ENDIF
      IF (TNSPLT.GT.TNDF) THEN
      WRITE (6,57)
      DO 4 J=1,TNSPLT-TNDF
      IF (TNDF.GT.0) THEN
      IF (SPLFRN (KPERM1 (J)) .LT.0.OR.SPLFRN (KPERM1 (J)) .GT.1) THEN
      WRITE (6,59)
      ENDIF
      WRITE (6,56) NSPLT (KPERM1 (J)) ,SPLFRN (KPERM1 (J))
      ELSE
      IF (SPLFRN (J) .LT.0.OR.SPLFRN (J) .GT.1) THEN
      WRITE (6,59)
      ENDIF
      WRITE (6,56) NSPLT (J) ,SPLFRN (J)
```

```
      IF (CC (LCLU) .LT.0.0.OR.CC (LCLU) .GT. (PRCENT/100.0)) THEN
      WRITE (6,59)
      ENDIF
      IF (CC (LCLU) .LT. (PRCENT/100.0)) THEN
      WRITE (6,51) LCLU,CC (LCLU) , ( (TUTIL-TOTAL) /TOTAL) *100.0
      ELSE
      WRITE (6,64) LCLU,CC (LCLU) ,TOTAL-TUTIL
      ENDIF
      ELSE
      KN=1
      WRITE (6,61)
      IF (CC (KN) .GT.0.0001.OR.CC (KN) .LT.-0.0001) WRITE (6,59)
      IF (HHOT.GT.HCOLD) THEN
      WRITE (6,62) KN,CC (KN) ,HHUTIL,HHOT
      ELSE
      WRITE (6,63) KN,CC (KN) ,-HCUTIL,HCOLD
      ENDIF
      ENDIF
      DO 1 J=KN+1, (TNHXER+KN)
      IF (CC (J) .LT.0) THEN
      WRITE (6,59)
      ENDIF
      IF (ABS (CC (J) *100.0) .LT.DELHMN) WRITE (6,66)
      WRITE (6,52) J,NHXER (J-KN) ,CC (J) ,HXL (J-KN)
    1 CONTINUE
      DO 2 J= (TNHXER+KN+1) , (3*TNHXER+KN) ,2
      IREP= (J-TNHXER-KN+1) /2
      IF (CC (J) .LT.0) THEN
      WRITE (6,59)
      ENDIF
      IF (ABS (HXL (IREP) ) .LT.DELHMN) THEN
      WRITE (6,65)
      ENDIF
      WRITE (6,53) J,NHXER (IREP) ,CC (J) , (T (MATRXU (NHXER (IREP) ,2) ) -T (MATRXU (
     +NHXER (IREP) ,5) ) )
      IF (CC (J+1) .LT.0) THEN
      WRITE (6,59)
      ENDIF
      IF (ABS (HXL (IREP) ) .LT.DELHMN) THEN
      WRITE (6,65)
      ENDIF
      WRITE (6,54) (J+1) ,NHXER (IREP) ,CC (J+1) , (T (MATRXU (NHXER (IREP) ,4) ) -T (M
     +ATRXU (NHXER (IREP) ,3) ) )
    2 CONTINUE
      IF (TNDF.GT.0) THEN
      WRITE (6,55)
      DO 3 J=1,TNDF
      IF (SPLFRN (KPERM (J) ) .LT.0.OR.SPLFRN (KPERM (J) ) .GT.1) THEN
      WRITE (6,59)
      ENDIF
      WRITE (6,56) NSPLT (KPERM (J) ) ,SPLFRN (KPERM (J) )
    3 CONTINUE
      ENDIF
      IF (TNSPLT.GT.TNDF) THEN
      WRITE (6,57)
      DO 4 J=1,TNSPLT-TNDF
      IF (TNDF.GT.0) THEN
      IF (SPLFRN (KPERM1 (J) ) .LT.0.OR.SPLFRN (KPERM1 (J) ) .GT.1) THEN
      WRITE (6,59)
      ENDIF
      WRITE (6,56) NSPLT (KPERM1 (J) ) ,SPLFRN (KPERM1 (J) )
      ELSE
      IF (SPLFRN (J) .LT.0.OR.SPLFRN (J) .GT.1) THEN
      WRITE (6,59)
      ENDIF
      WRITE (6,56) NSPLT (J) ,SPLFRN (J)
```

```
      COMMON /FEASC/ IFEAS
      REWIND 5
      REWIND 6
      NCOUNT=INTER=0
      READ(7,*) IPRNT
      IF(IPRNT.EQ.9) THEN
      INTER=1
      WRITE(8,*) ('INPUT LEVEL OF OUTPUT (0,1,2,3 OR 4) ')
      READ(7,*) IPRNT
      ENDIF
      READ(5,*) TNUNTS,TNSTR
      READ(5,*) ((MATRXU(J,K),K=1,5),J=1,TNUNTS)
      READ(5,*) TNINPT
      DO 7 J=1,TNINPT
      READ(5,*) NINPT(J), (FCPR(J,K),K=1,2), (TSR(J,K),K=1,2)
    7 CONTINUE
      READ(5,*) (TT(J),J=1,TNINPT)
      READ(5,*) DELTMN,PRCENT
      CALL COUNT
      IF(IFLAF.EQ.1) GO TO 99
      READ(5,*) (U(J),J=1,TNHXER)
      READ(5,*) (A(J),J=1,TNHXER)
      READ(5,*) ISTART,IEND
      DO 98 NCOUNT=ISTART,IEND
      IF(NCOUNT.EQ.ISTART) INITRN=0
      CALL PERM
      IF(IFLAP.EQ.1) GO TO 99
      CALL TMPRO
      CALL MASBL
      IF(IFLAG.EQ.1) GO TO 99
      CALL MXUTIL
      CALL TYPE
      IF(IPRNT.NE.0) THEN
      WRITE(6,62)
      WRITE(6,*) ('*************************************************')
      WRITE(6,*) ('*                                               *')
      WRITE(6,6000) NCOUNT
 6000 FORMAT(' *      CORNERPOINT NUMBER ',I5,20X,'*')
      WRITE(6,*) ('*                                               *')
      WRITE(6,*) ('*************************************************')
      WRITE(6,62)
      WRITE(6,63) DELTMN
      IF(IPINCH.EQ.1) THEN
      WRITE(6,64) TPINCH
      ELSE
      WRITE(6,*) ('PROBLEM IS UNPINCHED')
      ENDIF
      WRITE(6,65) HHOT
      WRITE(6,66) HCOLD
      WRITE(6,67) TOTAL
      WRITE(6,62)
      ENDIF
      ACCUR=1.0E-05
      CALL EQSOL(ACCUR)
      IF(IFLAX.EQ.1) GO TO 99
      IF(IPRNT.GT.0) THEN
      WRITE(6,62)
      CALL RITE
      ENDIF
      IF(IFLEE.EQ.1) GO TO 99
      IFLAO=0
      CALL OPT(IFLAO)
      IF(IFEAS.EQ.1.AND.TNDF.GT.0) THEN
      NCYC=0
      IFLAO=1
      CALL OPT(IFLAO)
```

```
C
      IF (TNMIX.GT.0) THEN
      DO 9 J=1,TNMIX
      FCP (MATRXU (NMIX (J) ,4) ) =FCP (MATRXU (NMIX (J) ,2) ) +FCP (MATRXU (NMIX (J) ,3
     +) )
      IF (NTYPE (MATRXU (NMIX (J) ,4) ) .EQ.4) THEN
      T (MATRXU (NMIX (J) ,4) ) = ( (FCP (MATRXU (NMIX (J) ,2) ) *T (MATRXU (NMIX (J) ,2)
     +) ) + (FCP (MATRXU (NMIX (J) ,3) ) *T (MATRXU (NMIX (J) ,3) ) ) ) /FCP (MATRXU (NMIX
     + (J) ,4) )
      ENDIF
    9 CONTINUE
      ENDIF
      IF (INITRN.EQ.1) CALL GUESS
C
C*    CALCULATE UTILITY LOADS
C
      DO 10 J=1,TNUTIL
      HUTIL (J) = (T (MATRXU (NUTIL (J) ,4) ) -T (MATRXU (NUTIL (J) ,2) ) ) *
     *FCP (MATRXU (NUTIL (J) ,2) )
   10 CONTINUE
      IFLAX=0
      IF (INITRN.EQ.2) CALL XSET
      IF (IFLAX.EQ.1) GO TO 29
      I=1
      TUTIL=0.0
      DO 19 J=1,TNUTIL
      TUTIL=TUTIL+ABS (HUTIL (J) )
   19 CONTINUE
C
C*    CALCULATE EQUATION VALUES (F) FOR USE IN BROYDN
C*    SHOULD ALL BE ZERO AT THE SOLUTION
C
      DO 11 J=1, (3*TNHXER) ,3
C
C*    ENERGY BALANCE ON HEAT EXCHANGER HOT STREAM
C
      F (J) = ( (T (MATRXU (NHXER (I) ,2) ) -T (MATRXU (NHXER (I) ,4) ) ) *FCP
     + (MATRXU (NHXER (I) ,2) ) ) -HXL (I)
C
C*    ENERGY BALANCE ON HEAT EXCHANGER COLD STREAM
C
      F (J+1) = ( (T (MATRXU (NHXER (I) ,3) ) -T (MATRXU (NHXER (I) ,5) ) ) *FCP
     + (MATRXU (NHXER (I) ,3) ) ) +HXL (I)
      DELT1=T (MATRXU (NHXER (I) ,2) ) -T (MATRXU (NHXER (I) ,5) )
      DELT2=T (MATRXU (NHXER (I) ,4) ) -T (MATRXU (NHXER (I) ,3) )
      CALL DELTA (DELT1,DELT2,DELTAX,IDELT)
C
C*    OVERALL ENERGY BALANCE ON HEAT EXCHANGERS
C*    EXCHANGER HEAT LOADS < DELHMN ARE TREATED AS
C*    BEING NEGLIGIBLE (IE. CORRECT SCALING IS
C*    REQUIRED)
C
      IF (ABS (HXL (I) ) .LT.DELHMN) THEN
      F (J+2) =HXL (I) * (ACCUR**2)
      ELSE
      F (J+2) = (U (I) *A (I) ) *DELTAX-HXL (I)
      ENDIF
      I=I+1
   11 CONTINUE
      I=1
      IF (TNMIX.GT.0) THEN
      DO 12 J=1,TNMIX
      IF (NTYPE (MATRXU (NMIX (J) ,4) ) .EQ.8.OR.NTYPE (MATRXU (NMIX (J) ,4) ) .EQ.
     +10) THEN
C
C*    ENERGY BALANCE ON MIXERS FROM WHICH A TARGET TEMP. MUST BE MET
```

```
      CC(J)=0.0
      ENDIF
    1 CONTINUE
      DO 2 J=(TNHXER+KN+1),(3*TNHXER+KN),2
      IREP=(J-TNHXER-KN+1)/2
C
C*    FOR THE HOT AND COLD ENDS OF ALL HEAT EXCHANGERS ACTUAL DELTA T
C*    > MIN. THEORETICAL POSSIBLE DELTA T.
C*     (SCALED BY FACTOR OF 100.0 TO BE APPROX IN RANGE 0 TO 10)
C
      IF(ABS(HXL(IREP)).LT.DELHMN)THEN
      CC(J)=0
      CC(J+1)=0
      ELSE
      CC(J)=(T(MATRXU(NHXER(IREP),2))-T(MATRXU(NHXER(IREP),5
     +))-DELTMN)/100.0
      CC(J+1)=(T(MATRXU(NHXER(IREP),4))-T(MATRXU(NHXER(IREP)
     +,3))-DELTMN)/100.0
      ENDIF
    2 CONTINUE
      IF(TNSPLT.GT.TNDF.AND.TNDF.GT.0)THEN
      DO 3 J=(3*TNHXER+KN+1),(3*TNHXER+(TNSPLT-TNDF)+KN)
C
C*    DEFINE RANGE CONSTRAINTS , SPLIT FRACTIONS NOT USED AS DEGREES
C*    OF FREEDOM (MUST BE >0 <1)
C
      CC(J)=SPLFRN(KPERM1(J-(3*TNHXER)-KN))
    3 CONTINUE
      ENDIF
      IFEAS=KKKM=0
      IF(LINN.EQ.1)THEN
      IF(IPINCH.EQ.0.AND.CC(1).GT.0.0001.OR.IPINCH.EQ.0.AND.
     +CC(1).LT.-0.0001)IFEAS=1
      ENDIF
      IF(IPINCH.EQ.1.OR.LINN.EQ.0)THEN
      KKKM=1
      LCLU=3*TNHXER+TNSPLT-TNDF+1
      IF(CC(LCLU).LT.0.0.OR.CC(LCLU).GT.(PRCENT/100.0))IFEAS=1
      ENDIF
      DO 4 J=(2-KKKM),3*TNHXER+1-KKKM
      IF(CC(J).LT.0)IFEAS=1
    4 CONTINUE
      IF(TNSPLT.GT.0)THEN
      DO 5 J=1,TNSPLT
      IF(SPLFRN(J).GT.1.OR.SPLFRN(J).LT.0)IFEAS=1
    5 CONTINUE
      ENDIF
      IF(IFEAS.EQ.0)IFLAA=-2
      RETURN
      END
      SUBROUTINE FUNCT1(IFLAU,N,XC,FC)
C
C*    SUBROUTINE TAKES CURRENT VALUES PROVIDED BY E04UAF FOR THE SPLIT
C*    FRACTIONS USED AS DEGREES OF FREEDOM AND CALLS EQSOL TO SOLVE
C*    FOR THE REMAINING VARIABLES.
C*    PROVIDES VALUE OF THE OBJECTIVE FUNCTION BEING MINIMIZED
C
      INTEGER TNDF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /PER1C/ KPERM(50),KPERM1(50)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      DIMENSION XC(N)
      DO 1 J=1,TNDF
      SPLFRN(KPERM(J))=XC(J)
```

```
   ENDIF
 4 CONTINUE
   ENDIF
   IF (IFEAS.EQ.0) THEN
   WRITE (6,58)
   ELSE
   WRITE (6,60)
   ENDIF
   RETURN
50 FORMAT (1H /1H , 'CONSTRAINT VALUES (C)  (SHOULD ALL BE +VE',
  +1H /1H , 'EXCEPT CC(',I2,') WHICH SHOULD BE IN RANGE',F6.3,' TO ',
  +F6.3,') ',1H /)
51 FORMAT (1H , 'NO. ',I2,'   (UTILITY USAGE)            C =',E10.3,
  +' (',F5.1,' P.C.> MIN ) ')
52 FORMAT (1H , 'NO. ',I2,'   (HEAT LOAD,UNIT NO.  ',I2,' ) ',' C =',E10.
  +3,' ( HT.LOAD =  ',F5.1,') ')
53 FORMAT (1H , 'NO. ',I2,'   (UNIT ',I2,'  HOT END DELT)   C =',E10.
  +3,' (',F6.1,'  DEGREES ) ')
54 FORMAT (1H , 'NO. ',I2,'   (UNIT ',I2,'  COLD END DELT) C =',E10.
  +3,' (',F6.1,'  DEGREES ) ')
55 FORMAT (1H /1H , 'SPLIT FRACTIONS USED AS DEGREES OF FREEDOM',1H //
  +1H ,'      UNIT NO.      SPLIT FRACTION',1H /)
56 FORMAT (1H ,8X,I2,8X,F11.4)
57 FORMAT (1H /1H , 'CALCULATED SPLIT FRACTIONS',1H //1H ,'     UNIT NO
  +.      SPLIT FRACTION',1H /)
58 FORMAT (1H /1H , '** ALL CONSTRAINTS ARE FEASIBLE',1H /)
59 FORMAT (1H , 'XXXXX   NEXT VALUE IS INFEASIBLE  XXXXX')
60 FORMAT (1H /1H , '** CURRENT VALUES ARE INFEASIBLE',1H /)
61 FORMAT (1H /1H , 'CONSTRAINT VALUES (C)  (SHOULD ALL BE +VE '
  +,1H /1H , 'EXCEPT C(1) WHICH SHOULD EQUAL ZERO ) ',1H /)
62 FORMAT (1H , 'NO. ',I2,'   (HOT UTILITY USE)          C =',
  +E10.3,' (',F5.0,'CF MIN',F5.0,') ')
63 FORMAT (1H , 'NO. ',I2,'   (COLD UTILITY USE)         C =',
  +E10.3,' (',F5.0,'CF MIN',F5.0,') ')
64 FORMAT (1H , 'NO. ',I2,'   (UTILITY USAGE)            C =',E10.3,
  +' (',F5.1,'<THEOR MIN) ')
65 FORMAT (' *****  NEXT VALUE IS IRRELEVANT  *****')
66 FORMAT (' *****  NEXT VALUE IS NEGLIGIBLE  *****')
   END
   SUBROUTINE AMONIT (N,M,XC,FC,CC,NITER,NF,GLNORM,COND,POSDEF,RHO,RLA
  +M)
C
C*    SUBROUTINE REQUIRED BY E04UAF TO MONITOR PROGRESS OF ROUTINE
C
      LOGICAL POSDEF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /OPTC/ MEQ,MINEQ,MRNGE,CL(20),CU(20)
      DIMENSION XC(N),CC(M),RLAM(M)
      IF (NITER.NE.-1) THEN
      WRITE (6,10) NITER
      RETURN
      ELSE
      NCYC=NCYC+1
      WRITE (6,11) NCYC
      ENDIF
      CALL RITE
      CALL CON1 (IFLAA,N,M,XC,CC)
      CALL FEAS (CC,M,IFEAS)
      IF (IPRNT.GE.3) THEN
      DUMMY=0.0
      DO 60 IS=1,MINEQ
      IL=MEQ+IS
      CTEMP=CC(IL)
      IF (CTEMP.LT.0.0) DUMMY=DUMMY+CTEMP**2
```

```
      IF (IPRNT.GT.0) THEN
      WRITE (6,*) ('=============================================== ')
      WRITE (6,*) ('=                FINAL     RESULTS             = ')
      WRITE (6,*) ('=============================================== ')
      CALL RITE
      IFLAO=3
      CALL OPT (IFLAO)
      ENDIF
      ENDIF
      IF (IFEAS.EQ.1) THEN
      WRITE (6,*) ('XXXXX  CURRENT POINT INFEASIBLE  XXXXX')
      GO TO 99
      ENDIF
   98 CONTINUE
      WRITE (6,*) ('ALL POINTS TESTED')
   99 WRITE (6,60) NCOUNT
      STOP
   60 FORMAT (1H ///1H , 'PROGRAM ENDS AT CORNERPOINT NO.    ', I5///)
   62 FORMAT (1H /)
   63 FORMAT (1H , 'FOR A MIN. TEMP. APPROACH OF          ', F9.2)
   64 FORMAT (1H , '(COLD STREAM) PINCH TEMP. IS          ', F9.2)
   65 FORMAT (1H , 'MINIMUM HOT UTILITY USAGE IS          ', F9.2)
   66 FORMAT (1H , 'MINIMUM COLD UTILITY USAGE IS         ', F9.2)
   67 FORMAT (1H , 'MINIMUM TOTAL UTILITY USAGE IS        ', F9.2)
      END
      SUBROUTINE MASBL
C
C*    SUBROUTINE ASSIGNS THE CORRECT FCP VALUES TO STREAMS WHERE
C*    POSSIBLE.STREAMS WITH UNKNOWN FCP'S (IE.THOSE IN STREAM SPLITS)
C*    ARE IDENTIFIED AND ASSIGNED -VE FCP'S.
C*    THE UNIT MATRIX IS CHECKED FOR CONSISTENCY.
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /MASBLC/ IFLAG,TNUNKF,NUNKF(50)
      COMMON /TMPROC/ T(100),FCP(100)
      DO 13 J=1,TNSTR
      DO 14 K=1,TNINPT
      IF (J.EQ.NINPT(K)) GO TO 13
   14 CONTINUE
      FCP(J)=0.0
   13 CONTINUE
      IFLAG=0
      DO 3 J=1,TNINPT
      K=NINPT(J)-1
    7 K=K+1
      I=1
    4 IF (MATRXU(I,2).EQ.K.OR.MATRXU(I,3).EQ.K) THEN
      IF (MATRXU(I,3).GT.0.AND.MATRXU(I,5).GT.0) THEN
      IF (MATRXU(I,3).EQ.K) THEN
      FCP(MATRXU(I,5))=FCP(MATRXU(I,3))
      ELSE
      FCP(MATRXU(I,4))=FCP(MATRXU(I,2))
      ENDIF
      ELSE IF (MATRXU(I,3).EQ.0.AND.MATRXU(I,5).EQ.0) THEN
      FCP(MATRXU(I,4))=FCP(MATRXU(I,2))
      ELSE IF (MATRXU(I,3).EQ.0) THEN
      FCP(MATRXU(I,4))=-ABS(FCP(MATRXU(I,2)))
      FCP(MATRXU(I,5))=-ABS(FCP(MATRXU(I,2)))
      ELSE IF (MATRXU(I,5).EQ.0) THEN
      FCP(MATRXU(I,2))=-ABS(FCP(MATRXU(I,2)))
      FCP(MATRXU(I,3))=-ABS(FCP(MATRXU(I,3)))
      IF (FCP(MATRXU(I,4)).GT.-0.001.AND.FCP(MATRXU(I,4)).LT.0.001) THEN
```

```
C
      F((3*TNHXER)+I)=(FCP(MATRXU(NMIX(J),4))*T(MATRXU(NMIX(J),4)))
     +-(FCP(MATRXU(NMIX(J),2))*T(MATRXU(NMIX(J),2)))-(FCP(MATRXU(
     +NMIX(J),3))*T(MATRXU(NMIX(J),3)))
      I=I+1
      ENDIF
   12 CONTINUE
      ENDIF
      IF(INITRN.EQ.3)THEN
      INITRN=4
      DO 13 J=1,NBROY
      BOUND(J)=-1.0E100
      BOUND(NBROY+J)=1.0E100
   13 CONTINUE
      KALL=0
      NB=2*NBROY
      NW=NBROY*(NBROY+3)
      IF(IPRNT.EQ.4)THEN
      JPRNT=2
      ELSE
      JPRNT=-1
      ENDIF
      ENDIF
      ENDIF
      IF(KALL.GT.0.AND.KOND.EQ.0)KALL=0
      KALL=KALL+1
C
C*    CALL EQUATION SOLVING ROUTINE BROYDN (FROM CELIB)
C
      CALL BROYDN(NBROY,X,F,ACCUR,BOUND,NB,W,NW,KOND,KALL,JPRNT)
C
C*    CONVERT X VALUES FROM BROYDN TO PHYSICAL EQUIVALENTS
C
      IF(TNSPLT.NE.TNDF)THEN
      IF(TNDF.GT.0)THEN
      DO 14 J=1,(TNSPLT-TNDF)
      SPLFRN(KPERM1(J))=X(J)
   14 CONTINUE
      ELSE
      IF(TNSPLT.GT.0)THEN
      DO 2 J=1,TNSPLT
      SPLFRN(J)=X(J)
    2 CONTINUE
      ENDIF
      ENDIF
      ENDIF
      DO 15 J=1,TNHXER
      HXL(J)=X(TNSPLT-TNDF+J)
   15 CONTINUE
      I=1
      DO 16 J=1,TNSTR
      IF(NTYPE(J).EQ.2)THEN
      T(J)=X(TNSPLT-TNDF+TNHXER+I)
      I=I+1
      ENDIF
   16 CONTINUE
      IF(KALL.GT.1000.AND.NCOUNT.GT.ISTART)IFLEE=1
      IF(KALL.GT.5000.AND.NCOUNT.EQ.ISTART)IFLEE=1
      IF(KOND.EQ.1.AND.IFLEE.EQ.0)GO TO 17
      IF(IFLEE.EQ.1)WRITE(6,*)('XXXXX  ITERATIONS IN BROYDN EXCEEDED SET
     + NO.  XXXXX')
      IF(KOND.NE.0)THEN
      IFLEE=1
      IF(KOND.EQ.-4)WRITE(6,*)('XXXXX  DIMENSIONS OF W OR BOUND TOO SMAL
     +L  XXXXX')
      IF(KOND.EQ.-3)WRITE(6,*)('XXXXX  EQUATIONS MAY NOT BE INDEPENDENT
     + XXXXX')
```

```
    1 CONTINUE
      ACCUR=1.0E-07
      CALL EQSOL(ACCUR)
      IF(IFLAX.EQ.1.OR.IFLEE.EQ.1) IFLAU=-1
C
C*    OBJECTIVE FUNCTION TO BE MINIMIZED IS THE ACTUAL UTILITY USAGE
C*     (SCALED)
C
      FC=((TUTIL-TOTAL)/TOTAL)**2
      RETURN
      END
      SUBROUTINE PERM
C
C*    SUBROUTINE GENERATES COMBINATION OF UPPER/LOWER BOUNDS (ON INPUT
C*    CONDITIONS) CORRESPONDING TO CURRENT CORNERPOINT
C
      INTEGER TNINPT
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /PERMC/ IPERM1(50),IPERM2(50),IFLAP
      DIMENSION IPERM(100),NCHANJ(100)
      IFLAP=0
      IF(NCOUNT.LE.(4**TNINPT)) THEN
      DO 7 J=1,(2*TNINPT)
      NCHANJ(J)=2**(J-1)
      NCOUNP=NCOUNT
      DO 6 JJ=1,(4**TNINPT)
      IF(NCOUNP.LE.NCHANJ(J)) GO TO 9
      NCOUNP=NCOUNP-NCHANJ(J)
    6 CONTINUE
    9 JJ=JJ-1
      TRUNC=AINT(FLOAT(JJ)/2.0)
      IF((TRUNC+0.1).GT.(FLOAT(JJ)/2.0)) THEN
      IPERM(J)=1
      ELSE
      IPERM(J)=2
      ENDIF
    7 CONTINUE
      DO 8 J=1,TNINPT
      IPERM1(J)=IPERM(J)
      IPERM2(J)=IPERM(TNINPT+J)
    8 CONTINUE
      ELSE
      WRITE(6,*) ('NEXT CORNERPOINT G.T. NO. OF POSSIBLE COMBINATIONS')
      IFLAP=1
      ENDIF
      RETURN
      END
      SUBROUTINE DELTA(DELT1,DELT2,DELTAX,IDELT)
C
C*    SUBROUTINE CALCULATES LOG MEAN TEMPERATURE DIFFERENCE WITH
C*    CORRECTIONS FOR DISCONTINUITIES,SINGULARITIES AND UNDEFINED
C*    REGIONS
C
      PDELTA(XX,YY) = (XX-YY)/ALOG(XX/YY)
      VDELTX(XX,YY) =-((ABS(XX)-YY)/ALOG(ABS(XX)/YY))
      VDELTY(XX,YY) =-((XX-ABS(YY))/ALOG(XX/ABS(YY)))
      E=1.0E-10
      IF(DELT1.LT.E.AND.DELT1.GT.-E.AND.DELT2.LT.E.AND.DELT2.GT.-E) THEN
      DELTAX=0.0
      IDELT=1
      ELSE IF(DELT1.LT.E.AND.DELT1.GT.-E) THEN
      IDELT=1
      IF(DELT2.GT.0) THEN
      DELTAX=(((PDELTA(E,DELT2)-VDELTX(-E,DELT2))*(DELT1+E)/
```

```
 60 CONTINUE
    DO 61 IT=1,MRNGE
    IL=MEQ+MINEQ+IT
    CTEMP=CC(IL)
    IF(CTEMP.LT.CL(IT))DUMMY=DUMMY+(CL(IT)-CTEMP)**2
    IF(CTEMP.GT.CU(IT))DUMMY=DUMMY+(CTEMP-CU(IT))**2
 61 CONTINUE
    CNORM=SQRT(DUMMY)
    WRITE(6,12)NF,GLNORM,CNORM
    WRITE(6,14)COND
    WRITE(6,13)RHO,(RLAM(J),J=1,M)
    ENDIF
    RETURN
 10 FORMAT(1H /1H ,'CALLING MONITOR AT ITERATION NO.    ',I4/)
 11 FORMAT(1H /1H ,'CALLING MONITOR AT END OF CYCLE ',I4,10X,18('*'),1
   +H /)
 12 FORMAT(1H //1H ,'NO. FUNCTION CALLS =   ',I15,1H /1H ,'GLNORM (SMAL
   +L AT SOL)  ',E15.7,1H /1H ,'CNORM  (      ...      )  ',E15.7)
 13 FORMAT(1H ,'RHO                        ',E15.7,1H /1H ,'LAGRANGIAN MULT
   +S.    ',1H /,4(1H ,E15.7))
 14 FORMAT(1H ,'CONDITIONING FACTOR    ',E15.7)
    END
```

```
      FCP(MATRXU(I,4))=ABS(FCP(MATRXU(I,3)))
      ENDIF
      ELSE
      GO TO 11
      ENDIF
      GO TO 7
      ENDIF
      ENDIF
      IF(I.NE.TNUNTS)THEN
      I=I+1
      GO TO 4
      ENDIF
      ENDIF
      IF(K.NE.(NINPT(J+1)-1))GO TO 11
    3 CONTINUE
      DO 12 L=1,TNSTR
      IF(FCP(L).LT.0.001.AND.FCP(L).GT.-0.001)GO TO 11
   12 CONTINUE
      TNUNKF=0
      DO 15 K=1,TNSTR
      IF(FCP(K).LT.0.0)THEN
      TNUNKF=TNUNKF+1
      NUNKF(TNUNKF)=K
      ENDIF
   15 CONTINUE
      RETURN
   11 WRITE(6,*)('XXXXX ERROR IN UNIT MATRIX  XXXXX')
      IFLAG=1
      RETURN
      END
      SUBROUTINE COUNT
C
C*    SUBROUTINE IDENTIFIES THE UNITS INVOLVED IN THE PROBLEM
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      TNHXER=TNUTIL=TNMIX=TNSPLT=IFLAF=0
      DO 1 I=1,TNUNTS
      IF(MATRXU(I,3).GT.0.AND.MATRXU(I,5).GT.0)THEN
      IF(MATRXU(I,2).GT.0.AND.MATRXU(I,4).GT.0)THEN
      TNHXER=TNHXER+1
      NHXER(TNHXER)=I
      ELSE
      GO TO 10
      ENDIF
      ELSE IF(MATRXU(I,3).EQ.0.AND.MATRXU(I,5).EQ.0)THEN
      TNUTIL=TNUTIL+1
      NUTIL(TNUTIL)=I
      ELSE IF(MATRXU(I,3).EQ.0)THEN
      TNSPLT=TNSPLT+1
      NSPLT(TNSPLT)=I
      ELSE IF(MATRXU(I,5).EQ.0)THEN
      TNMIX=TNMIX+1
      NMIX(TNMIX)=I
      ELSE
      GO TO 10
      ENDIF
    1 CONTINUE
      RETURN
   10 WRITE(6,*)('XXXXX  FAILURE IN COUNT  XXXXX')
      IFLAF=1
      RETURN
      END
```

```
      IF (KOND.EQ.-3.AND.TNDF.GT.0) THEN
      IF (IPRNT.GT.0) THEN
      WRITE (6,*) ('FOR DEGREES OF FREEDOM')
      WRITE (6,*) ('                                ')
      WRITE (6,*) ('SPLIT FRACTION    UNIT NUMBER')
      DO 18 J=1,TNDF
      WRITE (6,90) KPERM (J) ,MATRXU (NSPLT (KPERM (J) ) ,1)
   90 FORMAT (I8,15X,I1)
   18 CONTINUE
      WRITE (6,*) (' ')
      ENDIF
      KALL=IFLEE=0
       INITRN=2
       GO TO 17
       ENDIF
       IF (KOND.EQ.-1) WRITE (6,*) ('XXXXX  BOUNDS ARE IMPEDING PROGRESS  XXX
      +XX')
       ENDIF
   29 RETURN
       END
       SUBROUTINE XSET
C
C*     SUBROUTINE IDENTIFIES THE VARIABLES INVOLVED IN THE PROBLEM
C*     AND TRANSFERS THEIR VALUES TO A VECTOR X FOR USE IN BROYDN.
C*     CHECKS ARE MADE FOR DEGREES OF FREEDOM > 0 AND FOR NO.
C*     OF VARIABLES = NO. EQUATIONS
C
       INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
      +TNDF
       CHARACTER IQUEST
       COMMON /INPUTC/ FCPR (50,2) ,TSR (50,2) ,TT (50) ,TNSTR,TNUNTS,TNINPT,
      +MATRXU (50,5) ,NINPT (50) ,U (50) ,A (50) ,NCOUNT,DELTMN,PRCENT,ISTART
      +, IPRNT,INTER,NCYC
       COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER (50) ,NMIX (20) ,
      +NSPLT (20) ,NUTIL (50) ,IFLAF
       COMMON /GUESSC/ HXL (50) ,SPLFRN (20) ,DELHMN
       COMMON /TYPEC/ NTYPE (100) ,NHOT,IRDF,LINN
       COMMON /EQSOLC/ IFLEE,HUTIL (50) ,INITRN,KALL,KOUNT,TUTIL
       COMMON /XSETC/ X (200) ,NBROY,TNDF,IFLAX
       COMMON /TMPROC/ T (100) ,FCP (100)
       COMMON /PER1C/ KPERM (50) ,KPERM1 (50)
       INITRN=3
       IQUEST='N'
       KSKIP=0
C
C*     CALCULATE EFFECTIVE DEGREES OF FREEDOM
C
       TNDF=TNUTIL+TNSPLT-TNINPT-IRDF
       IF (TNDF.LT.0) GO TO 30
       IF (TNDF.GT.0.AND.TNDF.GT.TNSPLT) GO TO 28
       IF (TNDF.GT.0) THEN
C
C*     REMOVE SUFFICIENT SPLIT FRACTIONS FOR USE AS DEGREES OF FREEDOM
C
       IF (INTER.EQ.1) THEN
       WRITE (8,*) ('DO YOU WANT TO SPECIFY WHICH SPLIT FRACTIONS')
       WRITE (8,*) ('TO USE AS DEGREES OF FREEDOM ? (Y OR N) ')
       READ (7,7000) IQUEST
 7000  FORMAT (A1)
       ENDIF
       IF (IQUEST.NE. 'Y') THEN
       KOUNT=KOUNT+1
       CALL PERM1 (TNDF,TNSPLT,KOUNT,IFLAP1,KSKIP)
       IF (IFLAP1.EQ.1) GO TO 27
       ELSE
       WRITE (8,*) ('THE SPLITTERS ARE UNITS')
```

```
+(2.0*E))+VDELTX(-E,DELT2))
 ELSE
 DELTAX=(((VDELTY(E,DELT2)+PDELTA(-E,DELT2))*(DELT1+E)/
+(2.0*E))-PDELTA(-E,DELT2))
 ENDIF
 ELSE IF(DELT2.LT.E.AND.DELT2.GT.-E)THEN
 IDELT=1
 IF(DELT1.GT.0)THEN
 DELTAX=(((PDELTA(DELT1,E)-VDELTY(DELT1,-E))*(DELT2+E)/
+(2.0*E))+VDELTY(DELT1,-E))
 ELSE
 DELTAX=(((VDELTX(DELT1,E)+PDELTA(DELT1,-E))*(DELT2+E)/
+(2.0*E))-PDELTA(DELT1,-E))
 ENDIF
 ELSE IF((DELT1/DELT2).LT.(1.0+E).AND.(DELT1/DELT2).GT.(1.0-E))THEN
 DELTAX=((ABS(DELT1+DELT2)/2.0))
 IDELT=1
 ELSE IF(DELT1.GT.0.AND.DELT2.LT.0)THEN
 IDELT=1
 DELTAX=VDELTY(DELT1,DELT2)
 ELSE IF(DELT1.LT.0.AND.DELT2.GT.0)THEN
 IDELT=1
 DELTAX=VDELTX(DELT1,DELT2)
 ELSE IF(DELT1.LT.0.AND.DELT2.LT.0)THEN
 IDELT=1
 DELTAX=(-PDELTA(DELT1,DELT2))
 ELSE
 DELTAX=PDELTA(DELT1,DELT2)
 ENDIF
 RETURN
 END
 SUBROUTINE MXUTIL
C
C*    SUBROUTINE CALCULATES LINNHOFF PROBLEM TABLE
C
 INTEGER TNINPT,TNSTR
 COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
+MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
+,IPRNT,INTER,NCYC
 COMMON /TMPROC/ T(100),FCP(100)
 COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
 DIMENSION XX1(50),XX2(50),IS(50),X(100),SIGMA
+(100),TSTAR(100),SIG(50),FLOW(100),XFIX(100)
 NINPT2=2*TNINPT
 DO 1 J=1,TNINPT
 IF(T(NINPT(J)).GT.T(NINPT(J+1)-1))THEN
 XX1(J)=T(NINPT(J))-DELTMN
 XX2(J)=T(NINPT(J+1)-1)-DELTMN
 IS(J)=1
 ELSE
 XX1(J)=T(NINPT(J))
 XX2(J)=T(NINPT(J+1)-1)
 IS(J)=-1
 ENDIF
1 CONTINUE
 DO 2 J=1,TNINPT
 X(J)=XX1(J)
 X(TNINPT+J)=XX2(J)
2 CONTINUE
 DO 3 K=2,NINPT2
 XVAL=X(K)
 DO 4 L=1,K-1
 IF(XVAL.LT.X(L))THEN
 XFIX(L)=X(L)
 DO 5 M=L+1,K,1
 XFIX(M)=X(M)
```

```
      SUBROUTINE GUESS
C
C*    SUBROUTINE PROVIDES AN INITIAL GUESS OF SPLIT FRACTIONS,
C*    INTERMEDIATE STREAM TEMPS. ETC. (NEEDED ONCE ONLY DURING ANY
C*    RUN)
C
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      CHARACTER IQUEST
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /TMPROC/ T(100),FCP(100)
      IF(INITRN.EQ.1)THEN
      INITRN=2
      DO 3 J=1,TNHXER
      HXL(J)=(T(MATRXU(NHXER(J),2))-T(MATRXU(NHXER(J),4)))
     +*FCP(MATRXU(NHXER(J),2))
    3 CONTINUE
      ENDIF
      IF(INITRN.EQ.0)THEN
      INITRN=1
      IQUEST='Y'
      KFLAG=0
      DO 7 I=1,TNINPT
      IF((NINPT(I+1)-1).GT.(NINPT(I)+1))THEN
      DELINT=((T(NINPT(I+1)-1)-T(NINPT(I)))/((NINPT(I+1)-1)
     +-NINPT(I)))
      DO 8 J=(NINPT(I)+1),(NINPT(I+1)-2)
      IF(INTER.EQ.1.AND.IQUEST.EQ.'Y')THEN
      IF(KFLAG.EQ.0)THEN
      WRITE(8,*)('DO YOU WANT TO INPUT THE ')
      WRITE(8,*)('INTERMEDIATE STREAM TEMPS ? (Y OR N) ')
      READ(7,7000)IQUEST
 7000 FORMAT(A1)
      KFLAG=1
      ENDIF
      IF(IQUEST.EQ.'Y')THEN
      WRITE(8,6000)J
 6000 FORMAT('ESTIMATE FOR STREAM NO.',I4,' ?')
      READ(7,*)T(J)
      ENDIF
      ENDIF
      IF(INTER.EQ.0.OR.IQUEST.NE.'Y')T(J)=T(J-1)+DELINT
    8 CONTINUE
      ENDIF
    7 CONTINUE
      IF(TNSPLT.GT.0)THEN
      IQUEST='Y'
      KFLAG=0
      DO 10 J=1,TNSPLT
      IF(INTER.EQ.1.AND.IQUEST.EQ.'Y')THEN
      IF(KFLAG.EQ.0)THEN
      WRITE(8,*)('DO YOU WANT TO INPUT VALUES')
      WRITE(8,*)('FOR SPLIT FRACTIONS ? (Y OR N) ')
      READ(7,7000)IQUEST
      KFLAG=1
      ENDIF
      IF(IQUEST.EQ.'Y')THEN
      WRITE(8,6001)NSPLT(J)
 6001 FORMAT('ESTIMATE OF SPLIT FRACTION UNIT NO. ',I4,' ?')
      READ(7,*)SPLFRN(J)
```

```
      DO 12 KJ=1,TNSPLT
      WRITE(8,*)NSPLT(KJ)
   12 CONTINUE
      WRITE(8,8000)TNDF
 8000 FORMAT('INPUT',I4,' OF THESE')
      READ(7,*) (X(JL),JL=1,TNDF)
      KAWNT=KAWNT1=0
      DO 13 KJ=1,TNSPLT
      DO 14 KKJ=1,TNDF
      IF(X(KKJ).EQ.NSPLT(KJ))THEN
      KAWNT=KAWNT+1
      KPERM(KAWNT)=KJ
      GO TO 13
      ENDIF
   14 CONTINUE
      KAWNT1=KAWNT1+1
      KPERM1(KAWNT1)=KJ
   13 CONTINUE
      ENDIF
      ENDIF
      IF(TNDF.NE.TNSPLT.AND.TNDF.GT.0)THEN
      DO 8 J=1,(TNSPLT-TNDF)
      X(J)=SPLFRN(KPERM1(J))
    8 CONTINUE
      ELSE IF(TNDF.EQ.0)THEN
      IF(TNSPLT.GT.0)THEN
      DO 6 J=1,TNSPLT
      X(J)=SPLFRN(J)
    6 CONTINUE
      ENDIF
      ENDIF
      IF(KSKIP.GT.1)GO TO 29
      DO 9 J=1,TNHXER
      X(TNSPLT-TNDF+J)=HXL(J)
    9 CONTINUE
      I=1
      DO 10 J=1,TNSTR
      IF(NTYPE(J).EQ.2)THEN
      X(TNSPLT-TNDF+TNHXER+I)=T(J)
      I=I+1
      ENDIF
   10 CONTINUE
      NBROY=TNSPLT-TNDF+TNHXER+I-1
      IK=0
      DO 11 J=1,TNSTR
      IF(NTYPE(J).EQ.8.OR.NTYPE(J).EQ.10)IK=IK+1
   11 CONTINUE
      NEQ=(3*TNHXER)+IK
      IF(NEQ.NE.NBROY)THEN
      WRITE(6,*)('XXXXX  FAILURE IN XSET , NO. VARIABLES N.E. NO. EQNS.
     + XXXXX')
      IFLAX=1
      ENDIF
      GO TO 29
   28 WRITE(6,*)('XXXXX  DEGREES FREEDOM G.T. NO. SPLITS ,PROGRAM FAILS
     + XXXXX')
      GO TO 27
   30 WRITE(6,*)('XXXXX  DEGREES FREEDOM L.T. ZERO ,NO SOLUTION POSSIBLE
     +  XXXXX')
   27 IFLAX=1
   29 RETURN
      END
      SUBROUTINE TMPRO
C
C*    SUBROUTINE SETS INPUT TEMPS. AND FCP'S TO VALUES CORRESPONDING
C*    TO CURRENT CORNERPOINT
```

```
      X(M)=XFIX(M-1)
    5 CONTINUE
      X(L)=XVAL
      GO TO 3
      ENDIF
    4 CONTINUE
    3 CONTINUE
      DO 6 K=2,NINPT2
      SIGMA(K-1)=0.0
      TSTAR(K)=(X(K)+X(K-1))/2.0
      IF(ABS(TSTAR(K)).GT.0.001)THEN
      DO 7 J=1,TNINPT
      SIG(J)=0.0
      IF(IS(J).EQ.1.AND.TSTAR(K).GT.(T(NINPT(J+1)-1)-DELTMN).AND.
     +TSTAR(K).LT.(T(NINPT(J))-DELTMN))THEN
      SIG(J)=(X(K)-X(K-1))*FLOAT(IS(J))*FCP(NINPT(J))
      ELSE IF(IS(J).EQ.-1.AND.TSTAR(K).GT.T(NINPT(J)).AND.TSTAR(K).LT.T(
     +NINPT(J+1)-1))THEN
      SIG(J)=(X(K)-X(K-1))*FLOAT(IS(J))*FCP(NINPT(J))
      ENDIF
      SIGMA(K-1)=SIGMA(K-1)+SIG(J)
    7 CONTINUE
      ENDIF
    6 CONTINUE
      FLOW(1)=0.0
      DO 8 J=2,NINPT2
      FLOW(J)=FLOW(J-1)+SIGMA(NINPT2-J+1)
    8 CONTINUE
      HMIN=1.0E50
      DO 9 J=2,NINPT2,2
      HMIN1=AMIN1(FLOW(J),FLOW(J-1))
      IF(HMIN1.LT.HMIN)THEN
      IF(FLOW(J).LT.FLOW(J-1))THEN
      IPINCH=J
      ELSE
      IPINCH=J-1
      ENDIF
      HMIN=HMIN1
      ENDIF
    9 CONTINUE
      IF(ABS(HMIN).LT.0.001)THEN
      TOTAL=FLOW(NINPT2)
      HCOLD=TOTAL
      HHOT=0.0
      MFLAG=1
      ELSE IF(HMIN.GT.0.001)THEN
      TOTAL=FLOW(NINPT2)
      HCOLD=TOTAL
      HHOT=0.0
      MFLAG=2
      ELSE IF(HMIN.LT.-0.001)THEN
      TOTAL=FLOW(NINPT2)+(2.0*ABS(HMIN))
      MFLAG=3
      HHOT=ABS(HMIN)
      HCOLD=TOTAL-HHOT
      ENDIF
      IF(ABS(HHOT).GT.0.001.AND.ABS(HCOLD).GT.0.001)THEN
      TPINCH=X(NINPT2+1-IPINCH)
      IPINCH=1
      ELSE
      IPINCH=0
      ENDIF
      RETURN
      END
      SUBROUTINE PERM1(TNDF,TNSPLT,KOUNT,IFLAP1,KSKIP)
```

```
      ENDIF
      ENDIF
      IF (INTER.EQ.0.OR.IQUEST.NE.'Y') SPLFRN(J)=0.9
   10 CONTINUE
      ENDIF
      IF (INTER.EQ.1) THEN
      WRITE(8,8000)
 8000 FORMAT(' DO YOU WANT TO INPUT THE MINIMUM HEAT EXCHANGER HEAT LOAD
     +  ',1H /,' WHICH IS TO BE CONSIDERED SIGNIFICANT (DEFAULT 1.0) ?',1
     +H /1H ,'(Y OR N) ')
      READ(7,7000) IQUEST
      IF (IQUEST.EQ.'Y') THEN
      WRITE(8,*) ('MINIMUM HEAT LOAD ?')
      READ(7,*) DELHMN
      ENDIF
      ENDIF
      IF (INTER.EQ.0.OR.IQUEST.NE.'Y') DELHMN=1.0
      ENDIF
      RETURN
      END
      SUBROUTINE TYPE
C
C*    SUBROUTINE ASSIGNS AN IDENTIFIER TO EACH STREAM ACCORDING TO
C*    TYPE.THEY ARE :
C*    1-----AN INPUT STREAM TO THE NETWORK
C*    2-----OUTPUT STREAM FROM A HEAT EXCHANGER (NOT IF TYPE 6)
C*    3-----OUTPUT FROM A SPLITTER
C*    4-----OUTPUT FROM A MIXER (NOT IF TYPE 8)
C*    5-----OUTPUT FROM A UTILITY EXCHANGER
C*    6-----OUTPUT FROM A HEAT EXCHANGER WHICH IS ALSO AN OUTPUT FROM
C*          THE NETWORK
C*    8-----OUTPUT FROM A MIXER WHICH IS ALSO AN OUTPUT FROM THE
C*          NETWORK
C*    9-----INPUT TO A REDUNDANT UTILITY EXCHANGER TREATED AS AN OUTPUT
C*          FROM THE NETWORK (MAY OCCUR IF PROBLEM IS UNPINCHED)
C*   10-----OUTPUT FROM A MIXER WHICH IS ALSO AN INPUT TO A REDUNDANT UTILITY
C*          EXCHANGER
C
      INTEGER TNINPT,TNSTR,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      DO 7 J=1,TNSTR
      NTYPE(J)=0
      DO 8 JJ=1,TNINPT
      IF (J.EQ.NINPT(JJ)) THEN
      NTYPE(J)=1
      GO TO 7
      ENDIF
    8 CONTINUE
      DO 9 JJ=1,TNHXER
      IF (J.EQ.MATRXU(NHXER(JJ),4).OR.J.EQ.MATRXU(NHXER(JJ),5)) THEN
      NTYPE(J)=2
      GO TO 7
      ENDIF
    9 CONTINUE
      IF (TNSPLT.GT.0) THEN
      DO 10 JJ=1,TNSPLT
      IF (J.EQ.MATRXU(NSPLT(JJ),4).OR.J.EQ.MATRXU(NSPLT(JJ),5)) THEN
      NTYPE(J)=3
      GO TO 7
```

```
      INTEGER TNINPT,TNSTR,TNUNKF,TNHXER,TNUTIL,TNMIX,TNSPLT,TNUNTS,
     +TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /PERMC/ IPERM1(50),IPERM2(50),IFLAP
      NINPT(TNINPT+1)=TNSTR+1
      DO 7 J=1,TNINPT
      T(NINPT(J))=TSR(J,IPERM1(J))
      T(NINPT(J+1)-1)=TT(J)
      FCP(NINPT(J))=FCPR(J,IPERM2(J))
    7 CONTINUE
      RETURN
      END
      SUBROUTINE OPT(IFLAO)
C
C*    SUBROUTINE CALLED WITH IFLAO=0 OR 3 PROVIDES CONSTRAINT AND
C*    FEASIBILITY INFORMATION.
C*    CALLED WITH IFLAO=1 MANIPULATES THE DEGREES OF FREEDOM USING
C*    AN OPTIMIZATION ROUTINE IN AN ATTEMPT TO FIND A FEASIBLE POINT
C*    WITH THE SECONDARY TASK OF MINIMIZING A GIVEN OBJECTIVE FUNCTION.
C
      INTEGER TNHXER,TNSPLT,TNDF
      CHARACTER IQUEST
      LOGICAL LAMSET
      EXTERNAL E04WAY
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /COMNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /PER1C/ KPERM(50),KPERM1(50)
      COMMON /FEASC/ IFEAS
      COMMON /OPTC/ MEQ,MINEQ,MRNGE,CL(20),CU(20)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      DIMENSION XC(20),CC(200),RLAM(200),WW(1000),
     +XL(20),XU(20),IW(200)
      IF(IFLAO.EQ.0)THEN
      N=TNDF
      M=3*TNHXER+1+TNSPLT-TNDF
      CALL CON1(IFLAO,N,M,XC,CC)
      IF(IPRNT.GT.0)THEN
      CALL FEAS(CC,M,IFEAS)
      ENDIF
      ENDIF
      IF(IFLAO.EQ.1)THEN
      IF(IPRNT.GT.0)THEN
      WRITE(6,20)
   20 FORMAT(1H /1H ,'CALLING OPTIMIZATION ROUTINE E04UAF  (FROM NAG8F)
     +',1H /)
      ENDIF
      IF(IPINCH.EQ.0)THEN
      MEQ=1
      MRNGE=TNSPLT-TNDF
      ELSE
      MEQ=0
      MRNGE=TNSPLT-TNDF+1
      ENDIF
      MINEQ=3*TNHXER
      IF(IPRNT.EQ.0.OR.IPRNT.EQ.1)THEN
      IPRINT=-1
      ELSE
```

```
C*     SUBROUTINE SELECTS TNDF SPLIT FRACTIONS FROM THE TOTAL OF TNSPLT
C*     FOR USE AS DEGREES OF FREEDOM.ALL POSSIBLE COMBINATIONS ARE
C*     GENERATED IF REQUIRED.
C
       INTEGER TNDF,TNSPLT
       COMMON /PER1C/ KPERM(50),KPERM1(50)
       DIMENSION NCHANJ(100),IPERM(100)
       KSKIP=KSKIP+1
       IFLAP1=0
   10  IF (KOUNT.LE. (2**TNSPLT)) THEN
       DO 7 J=1,TNSPLT
       NCHANJ(J)=2** (J-1)
       NCOUNP=KOUNT
       DO 6 JJ=1, (2**TNSPLT)
       IF (NCOUNP.LE.NCHANJ(J)) GO TO 9
       NCOUNP=NCOUNP-NCHANJ(J)
    6  CONTINUE
    9  JJ=JJ-1
       TRUNC=AINT (FLOAT (JJ) /2.0)
       IF ( (TRUNC+0.1) .GT. (FLOAT (JJ) /2.0)) THEN
       IPERM(J)=1
       ELSE
       IPERM(J)=2
       ENDIF
    7  CONTINUE
       KJ=KK=0
       DO 8 J=1,TNSPLT
       IF (IPERM(J) .EQ.1) THEN
       KK=KK+1
       KPERM1 (KK) =J
       ELSE
       KJ=KJ+1
       KPERM(KJ) =J
       ENDIF
    8  CONTINUE
       IF (KK.NE. (TNSPLT-TNDF)) THEN
       KOUNT=KOUNT+1
       GO TO 10
       ENDIF
       ELSE
       WRITE(6,*) ('ALL COMBIMATIONS OF SPLIT FRACTIONS HAVE BEEN TESTED'
      +)
       IFLAP1=1
       ENDIF
       RETURN
       END
       SUBROUTINE RITE
C
C*     SUBROUTINE PRODUCES OUTPUT OF NETWORK VARIABLES
C
       INTEGER TNSTR,TNHXER,TNUTIL,TNSPLT
       COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
      +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
      +,IPRNT,INTER,NCYC
       COMMON /COWNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
      +NSPLT(20),NUTIL(50),IFLAF
       COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
       COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
       COMMON /TMPROC/ T(100),FCP(100)
       IF (IFLEE.EQ.1) THEN
       WRITE(6,*) ('VALUES AT TIME OF FAILURE ARE')
       ELSE
       WRITE(6,*) ('BROYDN HAS CONVERGED TO REQUIRED ACCURACY')
       ENDIF
       WRITE(6,82)
       WRITE(6,79) KALL
```

```
      ENDIF
   10 CONTINUE
      ENDIF
      IF (TNMIX.GT.0) THEN
      DO 11 JJ=1,TNMIX
      IF (J.EQ.MATRXU (NMIX (JJ) ,4) ) THEN
      NTYPE (J) =4
      GO TO 7
      ENDIF
   11 CONTINUE
      ENDIF
      DO 12 JJ=1,TNUTIL
      IF (J.EQ.MATRXU (NUTIL (JJ) ,4) ) THEN
      NTYPE (J) =5
      ENDIF
   12 CONTINUE
    7 CONTINUE
      DO 13 J=1,TNSTR
      DO 14 JJ=1,TNINPT
      IF (J.EQ. (NINPT (JJ+1) -1) .AND.NTYPE (J) .EQ.2) THEN
      NTYPE (J) =6
      ENDIF
      IF (J.EQ. (NINPT (JJ+1) -1) .AND.NTYPE (J) .EQ.4) THEN
      NTYPE (J) =8
      ENDIF
   14 CONTINUE
   13 CONTINUE
      DO 15 J=1,TNINPT
      IF (T (NINPT (J+1) -1) .LT.T (NINPT (J) ) ) THEN
      NHOT=NINPT (J+1) -1
      ELSE
      GO TO 16
      ENDIF
   15 CONTINUE
C
C*    REMOVE REDUNDANT UTILITY EXCHANGERS FROM NETWORK
C
      16 IRDF=0
      LINN=0
C
C*    IF LINN=1 LINNHOFF'S ASSUMPTIONS ABOUT UNPINCHED NETWORKS
C*    ARE USED  IE. ONLY HEATING OR COOLING UTILITIES (NOT
C*    BOTH ) MUST BE USED WITH UTILITY USE = THEOR. MIN.
C*    IF LINN=0 HEATING AND COOLING CAN BE USED AND ENERGY
C*    RECOVERY TREATED AS FOR PINCHED PROBLEMS.
C
      IF (IPINCH.EQ.0.AND.LINN.EQ.1) THEN
      DO 17 J=1,TNUTIL
      IF (NHOT.GT.HCOLD) THEN
      IF (MATRXU (NUTIL (J) ,4) .LE.NHOT) THEN
      IF (NTYPE (MATRXU (NUTIL (J) ,2) ) .NE.4) THEN
      NTYPE (MATRXU (NUTIL (J) ,2) ) =9
      ELSE
      NTYPE (MATRXU (NUTIL (J) ,2) ) =10
      ENDIF
      IRDF=IRDF+1
      ENDIF
      ELSE
      IF (MATRXU (NUTIL (J) ,4) .GT.NHOT) THEN
      IF (NTYPE (MATRXU (NUTIL (J) ,2) ) .NE.4) THEN
      NTYPE (MATRXU (NUTIL (J) ,2) ) =9
      ELSE
      NTYPE (MATRXU (NUTIL (J) ,2) ) =10
      ENDIF
      IRDF=IRDF+1
      ENDIF
```

```
      IPRINT=0
      ENDIF
      MAXCAL=100*(N+5)*(N+MINEQ+MRNGE)
      ETA=1.0E-03
      XTOL=0
      STEPMX=1.0E05
      IF (MRNGE.GT.0) THEN
      LCLU=MRNGE
      IF (TNSPLT.GT.TNDF) THEN
      DO 1 J=1,TNSPLT-TNDF
      CL(J)=0.01
      CU(J)=0.99
    1 CONTINUE
      ENDIF
      IF (IPINCH.EQ.1) THEN
      CL(LCLU)=0.0
      CU(LCLU)=PRCENT/100.0
      ENDIF
      ELSE
      LCLU=1
      ENDIF
      IBOUND=3
      DO 2 J=1,TNDF
      XL(J)=0.01
      XU(J)=0.99
      XC(J)=SPLFRN(KPERM(J))
    2 CONTINUE
      LAMSET=.FALSE.
      IF (INTER.EQ.1) THEN
      INTER=2
      WRITE(8,*) ('DO YOU WANT TO INPUT A STARTING VALUE')
      WRITE(8,*) ('FOR THE PARAMETER RHO USED IN THE OPTIMIZING')
      WRITE(8,*) ('ROUTINE ? (Y OR N)')
      READ(7,7000) IQUEST
 7000 FORMAT(A1)
      IF (IQUEST.EQ.'Y') THEN
      WRITE(8,*) ('INPUT RHO (LOW VALUES (IE. 1.0) GIVES OPTIMIZATION')
      WRITE(8,*) ('AT EXPENSE OF FEASIBILITY.VICE VERSA FOR HIGH VALUES')
      WRITE(8,*) ('(IE. 5.0))')
      READ(7,*) RHO
      ENDIF
      ENDIF
      IF (INTER.EQ.0.OR.IQUEST.NE.'Y') RHO=5
      LIW=N+MINEQ+MRNGE+M+12
      NN=N+MINEQ+MRNGE
      LW=15*NN+NN*(NN-1)/2+2*N*M+2*N+7*M+MRNGE+3
      IFAIL=1
      CALL E04UAF(N,MEQ,MINEQ,MRNGE,M,E04WAY,IPRINT,MAXCAL,ETA,XTOL,
     +STEPMX,CL,CU,LCLU,IBOUND,XL,XU,LAMSET,XC,RHO,RLAM,FC,CC,IW,LIW,WW,
     +LW,IFAIL)
      IF (IFAIL.EQ.-1) RETURN
      IF (IFAIL.EQ.1) WRITE(6,*) ('XXXXX  ERROR EXIT TYPE 1 FROM E04UAF  XX
     +XXX')
      IF (IFAIL.EQ.2.AND.IPRNT.GT.0) WRITE(6,*) ('EXITING FROM E04UAF AFTER
     +  10 CYCLES')
      IF (IFAIL.EQ.-2.AND.IPRNT.GT.0) WRITE(6,*) ('FEASIBLE POINT FOUND')
      WRITE(6,*) ('    ')
      ENDIF
      IF (IFLAO.EQ.3) THEN
      CALL CON1(IFLAO,N,M,XC,CC)
      CALL FEAS(CC,M,IFEAS)
      ENDIF
      RETURN
      END
      SUBROUTINE CON1(IFLAA,N,M,XC,CC)
C
```

```
      WRITE(6,80)
      DO 8 J=1,TNSTR
      WRITE(6,81) J,T(J),FCP(J)
    8 CONTINUE
      WRITE(6,82)
      WRITE(6,83)
      DO 9 J=1,TNHXER
      IF(ABS(HXL(J)).LT.DELHMN) THEN
      WRITE(6,90) J,MATRXU(NHXER(J),1),U(J),A(J)
      ELSE
      WRITE(6,84) J,MATRXU(NHXER(J),1),HXL(J),U(J),A(J)
      ENDIF
    9 CONTINUE
      WRITE(6,82)
      WRITE(6,85)
      DO 10 J=1,TNUTIL
      WRITE(6,86) J,MATRXU(NUTIL(J),1),HUTIL(J)
   10 CONTINUE
      WRITE(6,82)
      WRITE(6,89) TUTIL
      IF(TNSPLT.GT.0) THEN
      WRITE(6,82)
      WRITE(6,87)
      DO 11   J=1,TNSPLT
      WRITE(6,88) J,MATRXU(NSPLT(J),1),SPLFRN(J)
   11 CONTINUE
      ENDIF
      WRITE(6,82)
   79 FORMAT(1H ,'EXIT FROM BROYDN AFTER ',I5,' ITERATIONS')
   80 FORMAT(1H /1H ,2X,'STREAM NO.',4X,'TEMPERATURE',7X,'FCP'/)
   81 FORMAT(1H ,4X,I3,11X,F6.2,6X,F8.2)
   82 FORMAT(1H /)
   83 FORMAT(1H ,'HEAT EXCHANGER',2X,'UNIT NO.  ',3X,'HEAT LOAD'
     +,6X,'U',6X,'AREA'/)
   84 FORMAT(1H ,7X,I2,9X,I2,6X,F9.2,2X,F9.2,1X,F8.2)
   85 FORMAT(1H ,'UTILITY EXCHANGER',2X,'UNIT NO.',3X,'HEAT LOAD'/)
   86 FORMAT(1H ,7X,I2,12X,I2,5X,F9.2)
   87 FORMAT(1H ,'SPLITTER',3X,'UNIT NO.',3X,'SPLIT FRACTION'/)
   88 FORMAT(1H ,3X,I2,9X,I2,10X,F5.3)
   89 FORMAT(1H ,'TOTAL UTILITY USAGE IS       ',F9.2)
   90 FORMAT(1H ,7X,I2,9X,I2,7X,'NEGLIGIBLE',F9.2,1X,F8.2)
      RETURN
      END
      SUBROUTINE FEAS(CC,M,IFEAS)
C
C*    SUBROUTINE PRODUCES OUTPUT OF CONSTRAINT VALUES
C
      INTEGER TNHXER,TNSPLT,TNDF
      COMMON /INPUTC/ FCPR(50,2),TSR(50,2),TT(50),TNSTR,TNUNTS,TNINPT,
     +MATRXU(50,5),NINPT(50),U(50),A(50),NCOUNT,DELTMN,PRCENT,ISTART
     +,IPRNT,INTER,NCYC
      COMMON /COUNTC/ TNHXER,TNMIX,TNSPLT,TNUTIL,NHXER(50),NMIX(20),
     +NSPLT(20),NUTIL(50),IFLAF
      COMMON /GUESSC/ HXL(50),SPLFRN(20),DELHMN
      COMMON /EQSOLC/ IFLEE,HUTIL(50),INITRN,KALL,KOUNT,TUTIL
      COMMON /XSETC/ X(200),NBROY,TNDF,IFLAX
      COMMON /TYPEC/ NTYPE(100),NHOT,IRDF,LINN
      COMMON /TMPROC/ T(100),FCP(100)
      COMMON /PER1C/ KPERM(50),KPERM1(50)
      COMMON /MAXUC/ TOTAL,HHOT,HCOLD,MFLAG,IPINCH,TPINCH
      COMMON /CON1C/ HHUTIL,HCUTIL
      DIMENSION CC(M)
      IF(IPINCH.EQ.1.OR.IPINCH.EQ.0.AND.LINN.EQ.0) THEN
      KN=0
      LCLU=3*TNHXER+TNSPLT-TNDF+1
      WRITE(6,50) LCLU,FLOAT(KN),PRCENT/100.0
```