Chapter VII

Computer Program for Heat Exchanger Network Design

7.1 Overview of The Program

The developed program is coded in Quick BASIC V.4.0 for IBM Personal Computer or compatible ones that run on MS-DOS or PC-DOS Version 2.1 or later.

The program is composed of six modules which are to be executed sequentially. The modules are

(1) PINCH.BAS

(2) RMATCHA.BAS

(3) RMATCHB.BAS

(4) PLOT.BAS

(5) BLOOP.BAS

(6) ECONOMIC.BAS

The functions of each module are as follows.

PINCH.BAS: In this module, the minimum utility requirements, or equivalently the maximum energy recovery (MER) extent, as well as the pinch temperature are predicted using the problem table method. In case of restricted matching, the extra utility requirement is calculated.

RMATCHA.BAS: This module is used to generate an above-the-pinch design using the pinch design method.

RMATCHB.BAS: This module is used to generate an below-the-pinch design.

PLOT.BAS: This module is used to create a grid representation of both above and below-the-pinch design structures, namely, the MER network.

BLOOP.BAS: The function of this module is to carry out evolutionary modification of the MER network. The number of heat exchangers required in the primary (MER) network is to be minimized evolutionarily with minimum increase in utility consumptions. There are two options for the evolution, namely, loop breaking and merging heat loads for any selected path. The first option is automatically carried out to search and break loops up to the second level, whereas the second option is used to help break loops of any level, which have been found by the user.

ECONOMIC.BAS: The annualised costs of new networks and the payback periods for revamped networks are evaluated in this module.

7.2 Assumption Used in The Developed Software

The developed software is based on the following assumptions.

(1) Process conditions are known and remain unchanged.

(2) No phase change of any fluid stream occurs during heat exchange.

(3) Heat loss to the environment is negligible.

7.3 Limitations of The Software

The developed program has been found to run successfully under the following limitations.

(1) The total number of hot and cold streams, excluding utilities, is less than 23.

(2) The number of exchangers in the generated network does not exceed 45 units.

7.4 Program Features

The flowchart of a sequential procedure for designing a heat exchanger network is shown in Figure 7.1. Each of these features is described below, according to the individual modules.

7.4.1 The PINCH.BAS module

The flowchart of this module is shown in Figure 7.2. The program requires that the user specifies all stream supply and target temperatures, their heat capacity flowrates and the desired minimum allowable temperature difference, δT_{min}. In case of restricted matching, all restricted stream pairs must be specified.

The problem table method is used to predict the energy targets and location of the pinch point. The outputs of this module include

(a) The problem table.

(b) The minimum hot and cold utility requirements.

(c) The pinch location.

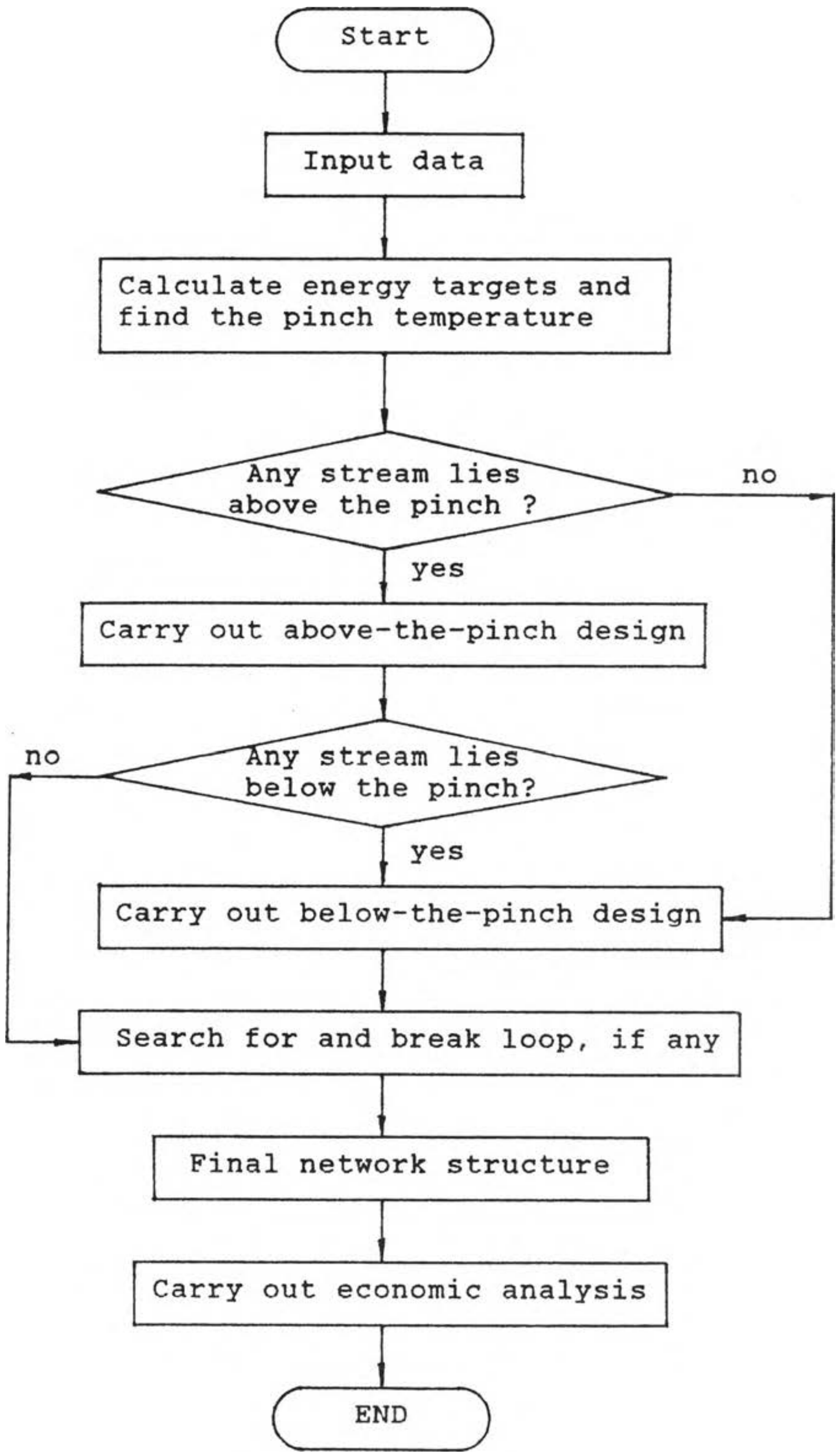

Figure 7.1 Simplified overall flowchart of the developed program

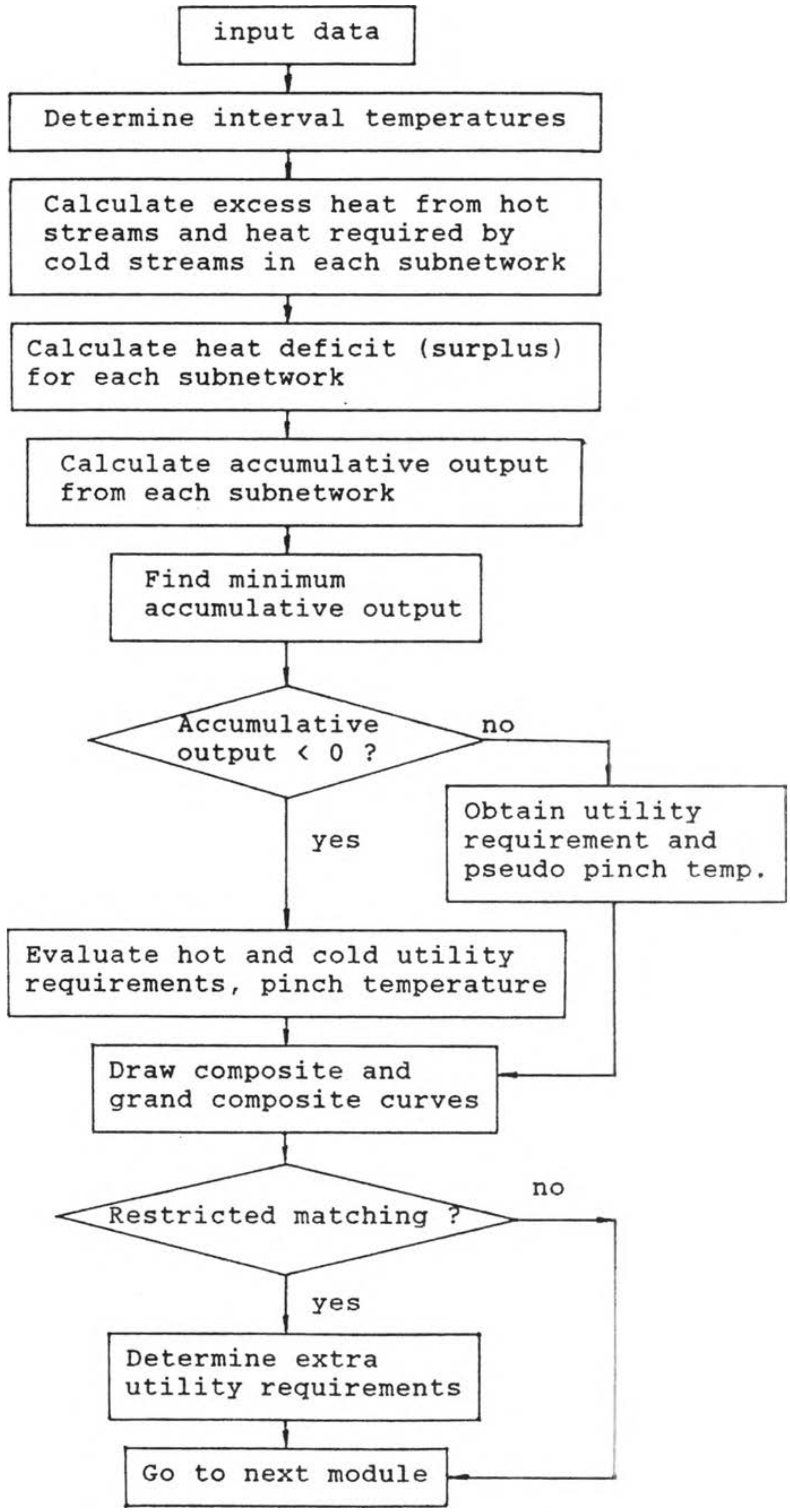

Figure 7.2 Flowchart of the PINCH.BAS module

(d) The hot and cold composite curves.

(e) The grand composite curve (plot of heat flow on the T/H diagram.

(f) The extra utility requirements incurred by restricted matching.

Deficient heat for each subnetwork is calculated according to equation (2-3). Accumulative output from any subnetwork is the energy cascaded down to the adjacent subnetwork.

$$(\text{accumulative output})_i = (\text{accumulative output})_{i-1} - (\text{deficient heat})_i \quad \text{.........(7-1)}$$

The subscript i represents the subnetwork number. If accumulative output is less than zero the problem is categorized as "pinch" type, and the pinch is located at the lower temperature of the subnetwork that generates the minimum accumulative output. Otherwise, it is a threshold problem, and a pseudo-pinch temperature will be evaluated instead.

If the problem is constrained by restricted stream/stream matches, the extra utility requirements will be calculated. The calculation procedure is quite similar to the "problem table" method except that the heat load belonging to a the restricted stream pair will not be interchanged. Figure 7.3 illustrates the typical interchange in a subnetwork problem containing two restricted stream pairs. The accumulative output is the

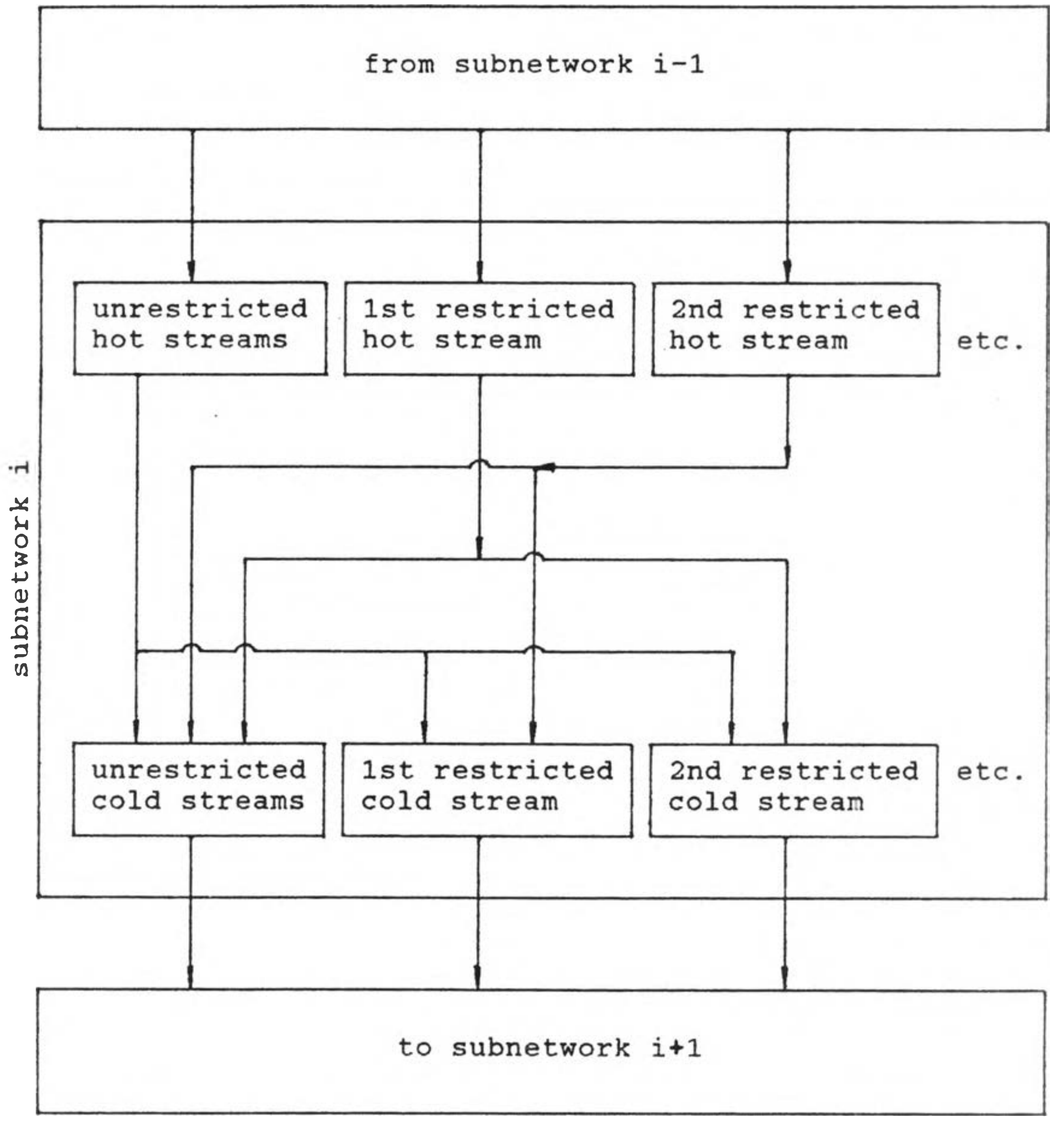

Figure 7.3 Typical interchange in a subnetwork constrained by restricted matches.

sum of the accumulative outputs of unrestricted stream pairs and those of restricted pairs. All the other subnetworks can be calculated in the same manner.

If the minimum accumulative output is less than zero, an equal amount of heat is added to the first subnetwork and cascaded down through the system. The amount of heat flow into the first subnetwork is the minimum hot utility requirement and the amount that flows out of the last subnetwork is the minimum cold utility requirement. The difference in hot (or cold) utility requirements between restricted and unrestricted conditions is the extra hot (or cold) utility requirement.

7.4.2 <u>The RMATCHA.BAS module</u>

Above the-pinch-design is composed of two design strategies, near-the-pinch design and heuristic away-from-the-pinch design, as described in Chapter 2.

7.4.2.1 Design at the pinch (above the pinch)

As can be seen from the pinch design algorithm shown in Figure 2.13(a), there are numerous alternatives for placing a match, as illustrated in Figure 2.17. However, in this thesis selection of a stream pair for placing match is definite, and splitting of a hot stream is carried out only when the CP criterion is violated. The pinch design algorithm of the developed program is shown in Figure 7.4. Here the hot streams as well as cold streams are first sorted out in the order of

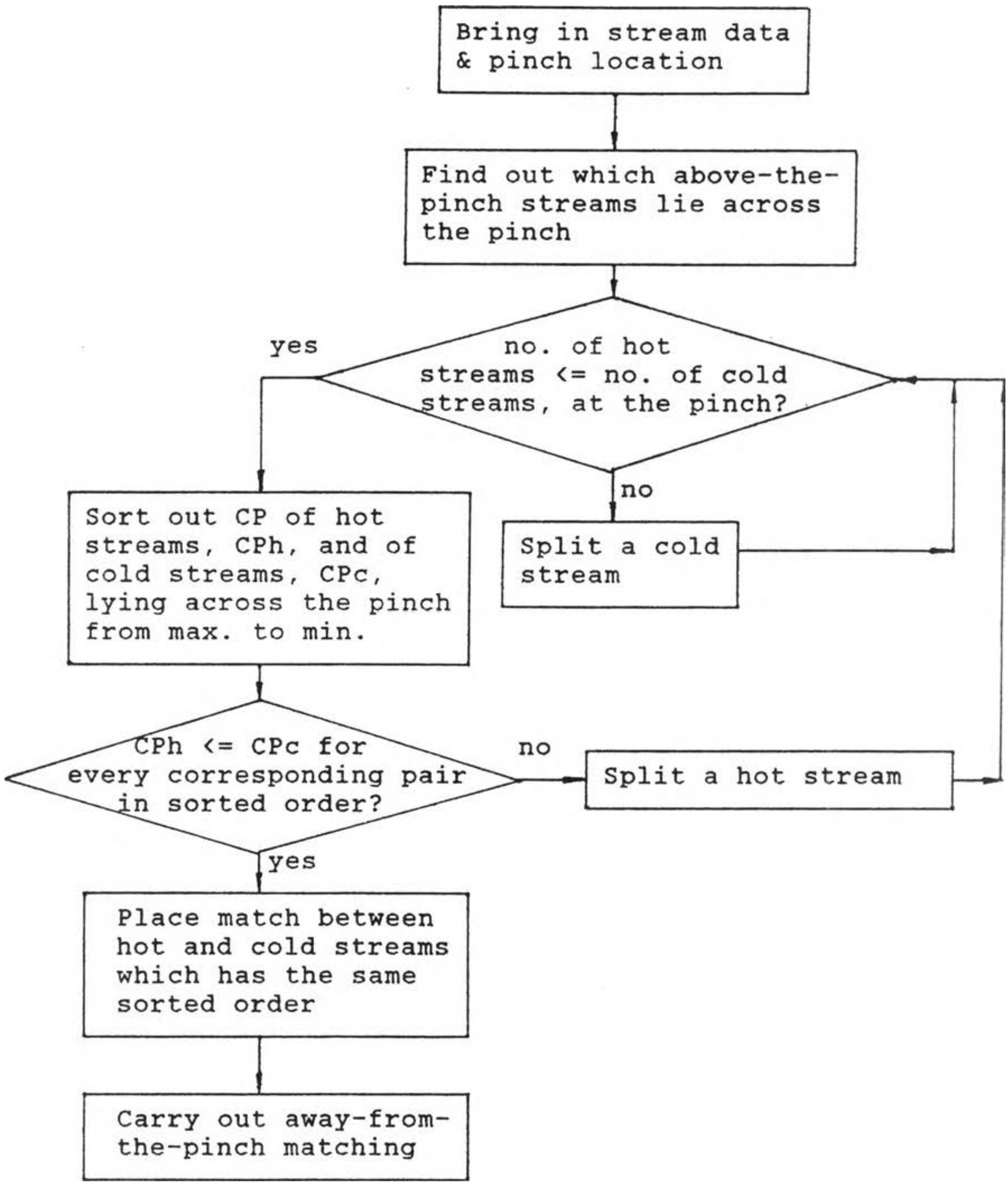

Figure 7.4 Design at pinch (above the pinch)

their CP's from maximum to minimum CP's. Then starting from the top, a hot and a cold stream is matched, pair by pair, while making the interchange load for each pair the maximum possible.

7.4.2.2 Heuristic design away from the pinch (above the pinch).

Heuristic procedure for away-from-the-pinch design is shown in Figure 2.15. Selection of a stream pair for placing match has been described in Chapter 2. However, some of the design steps are described below.

(a) Recombining of the split branches.

Since the design is gradually away from the pinch, we shall call the junction of branches which is

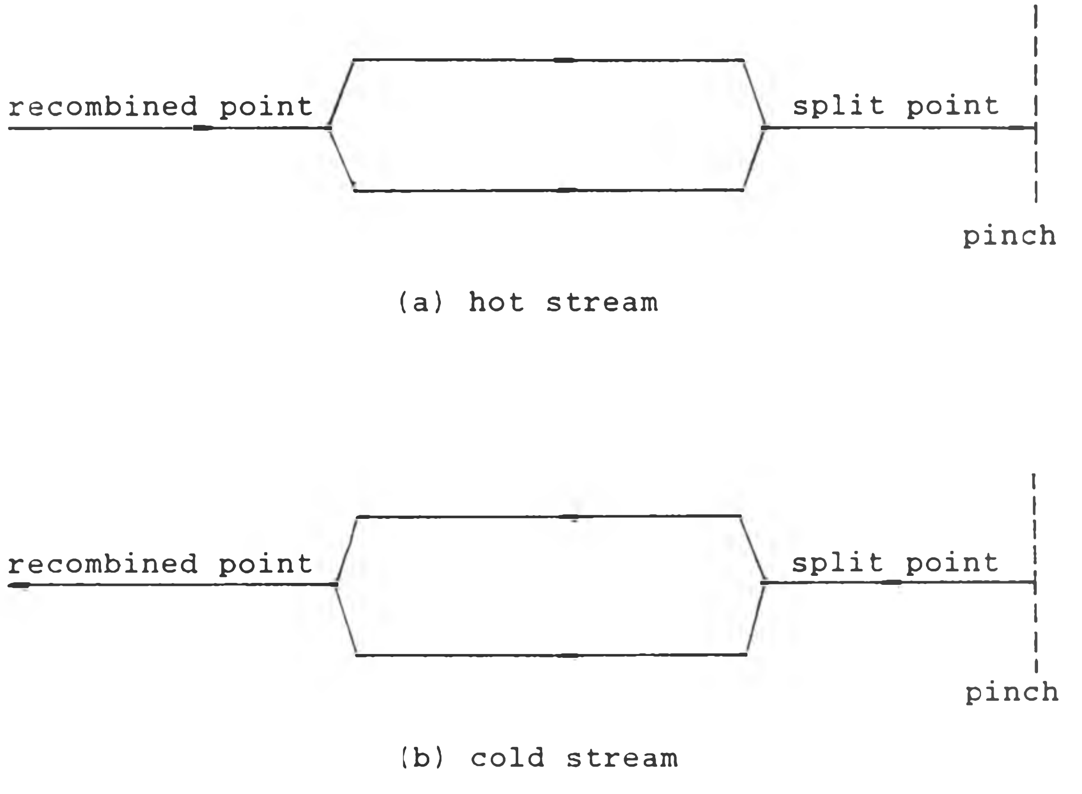

Figure 7.5 Naming of split and recombined points

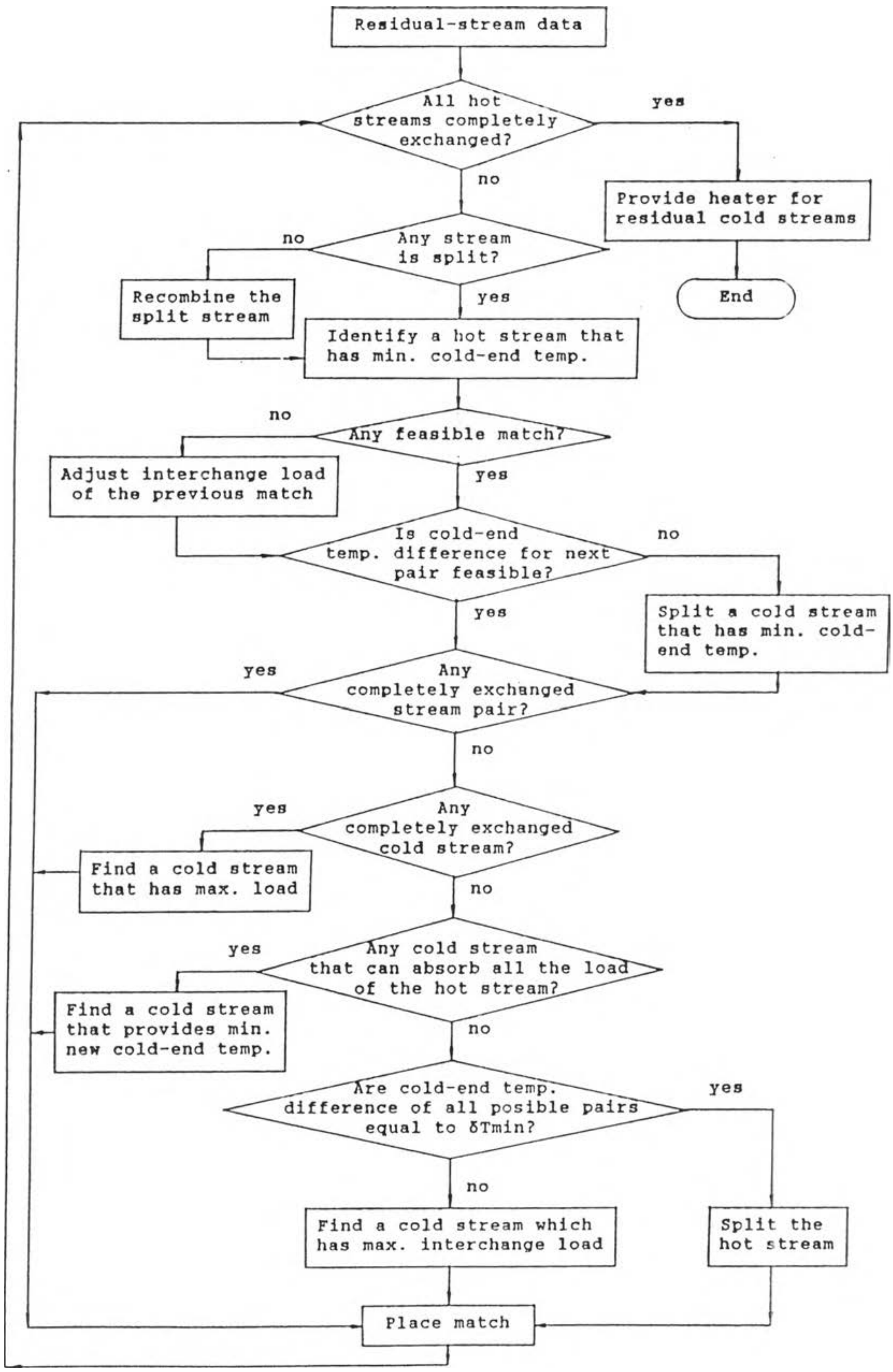

Figure 2.15 Heuristic procedure for above-the-pinch design

closer to the pinch the split point, and the junction farther from the pinch the recombined point, as shown in Figure 7.5. It obvious from Figure 7.5 that if the branches of a hot stream is to be recombined, the hot-end temperature of each branch must be equal. This restriction does not apply to the branches of a cold stream. In other words, the branches of any cold stream may be recombined directly without adjusting the temperature or heat capacity flowrate (CP) of any of them.

An algorithm for recombining above-the-pinch branches of both hot and cold streams is shown in Figure 7.6. Split streams are cannot be recombined unless the hot-end temperature of each branches is made equal to some predetermined temperature. Since the temperature of each recombined branch must be made equal without changing the interchange load of each branch, the only way out is to adjust the CP of each branch accordingly. The new CP of a branch is computed as the ratio of interchange load on that branch to the new temperature difference. If both terminal temperatures of every match obey the temperature constraint, the recombination is successful. On the other hand, if a cold stream cannot be recombined under this condition, it may still be recombined directly without adjusting the temperature or CP.

(b) Checking for a feasible match.

The purpose of this step is to check the cold-end temperature difference between a given hot

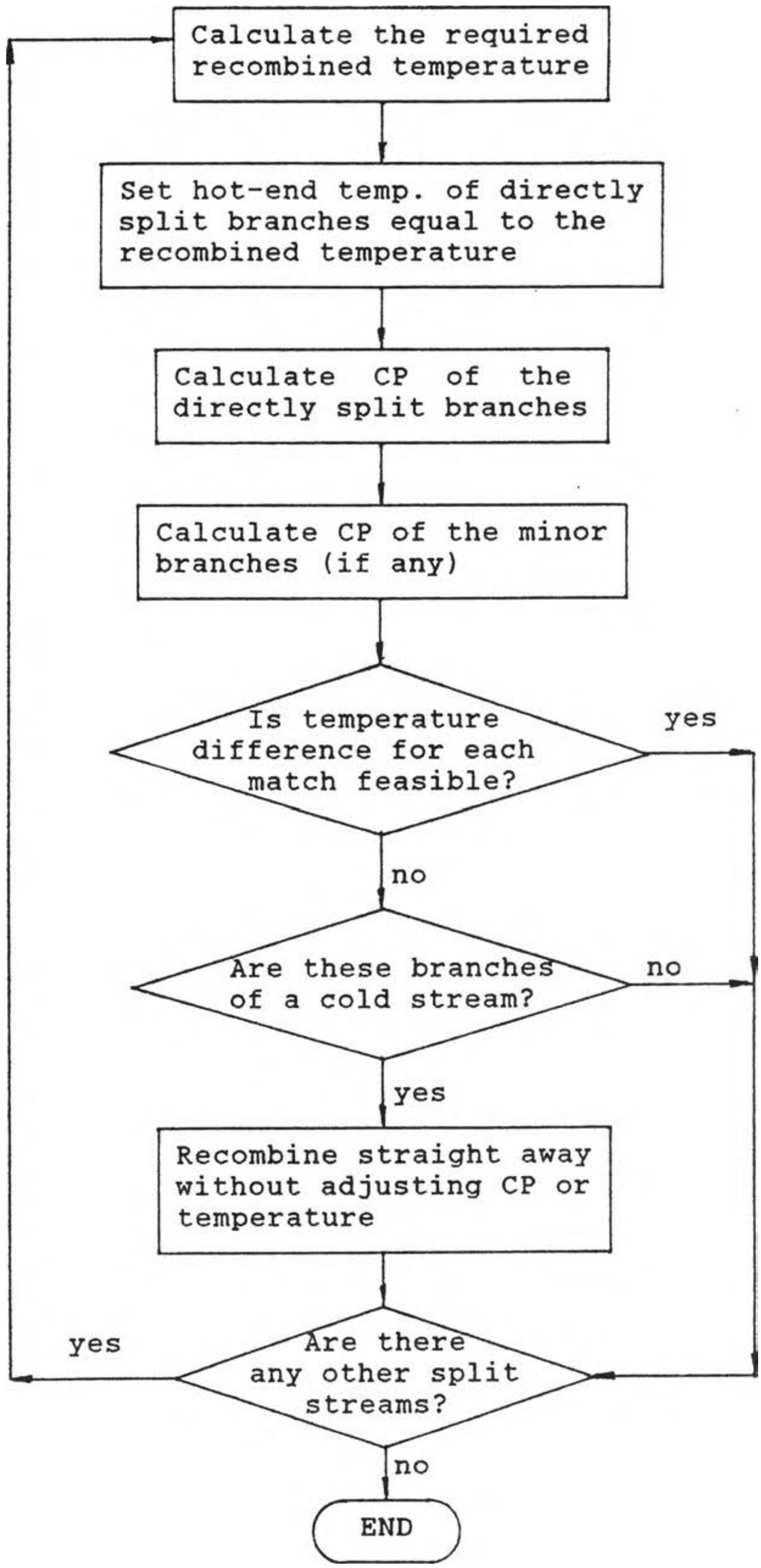

Figure 7.6 Recombination of split streams (above the pinch)

stream and any of the residual cold streams to see if there exists a suitable cold streams which can match hot stream. If all the pairing violate the given δTmin, the earlier matching results will be adjusted so that they yield a modified cold stream which makes a feasible matching.

(c) Check the cold-end temperature difference of the following pair.

This step is to check whether there is any cold stream that can match with the second least hot stream (judged by its cold-end temperature) after the present stream pair has been selected. This is checked by assuming that the least hot stream is already matched with the most cold stream, then the cold-end temperature difference between the second least hot stream and any of the residual cold streams. If the creation of such a match is infeasible, then the most cold stream will be split prior to the present matching.

7.4.3 The RMATCHB.BAS module

Below-the-pinch design is likewise composed of two design strategies, namely, near-the-pinch design and heuristic away-from-the-pinch design. Figure 7.7 showns a typical pinch design algorithm. First the hot streams as well as cold streams are sorted out in the other of their CP. Then starting from the top, a cold and a hot stream is matched, pair by pair, while making the interchange load for each pair the maximum possible. Splitting of a

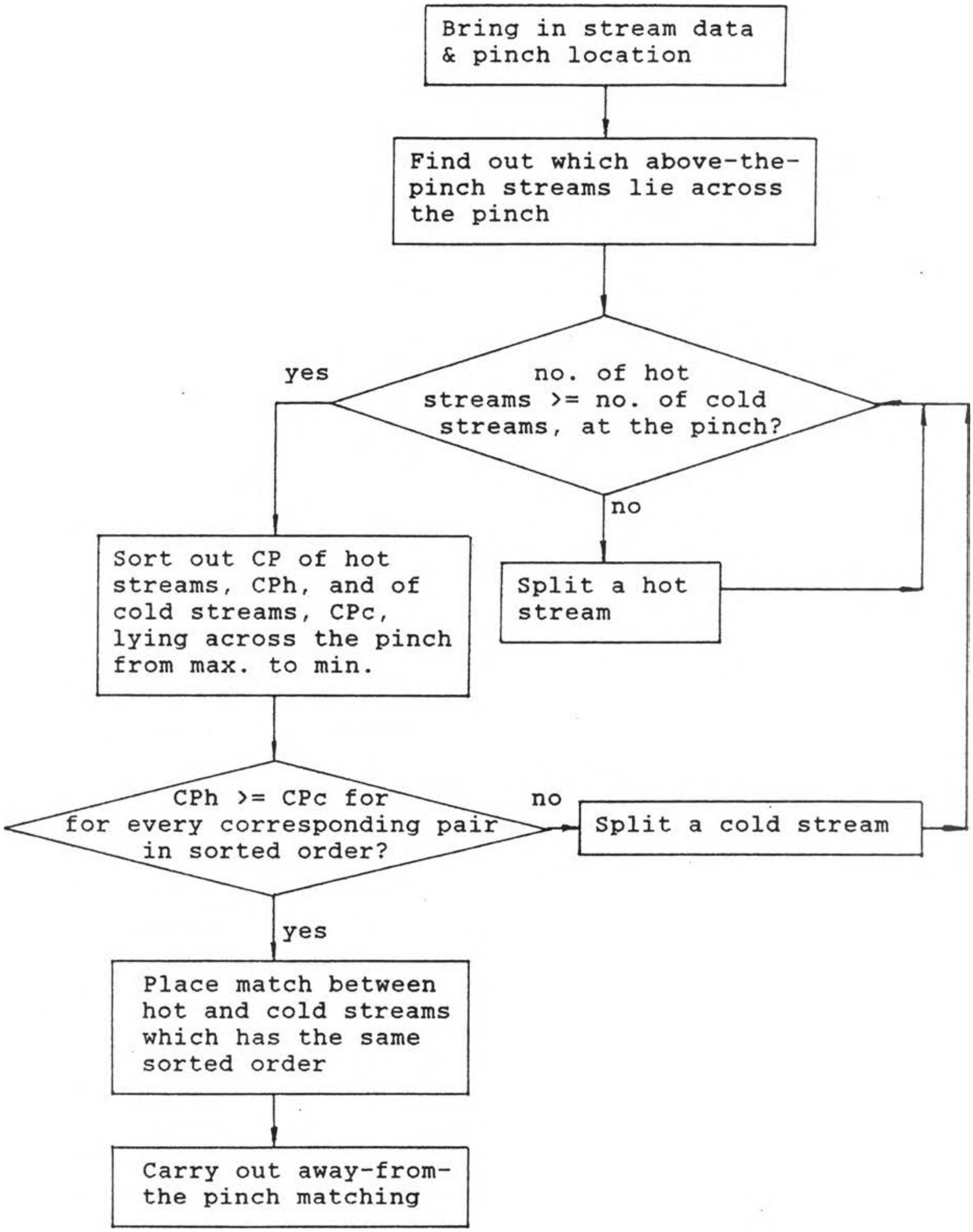

Figure 7.7 Design at pinch (below the pinch)

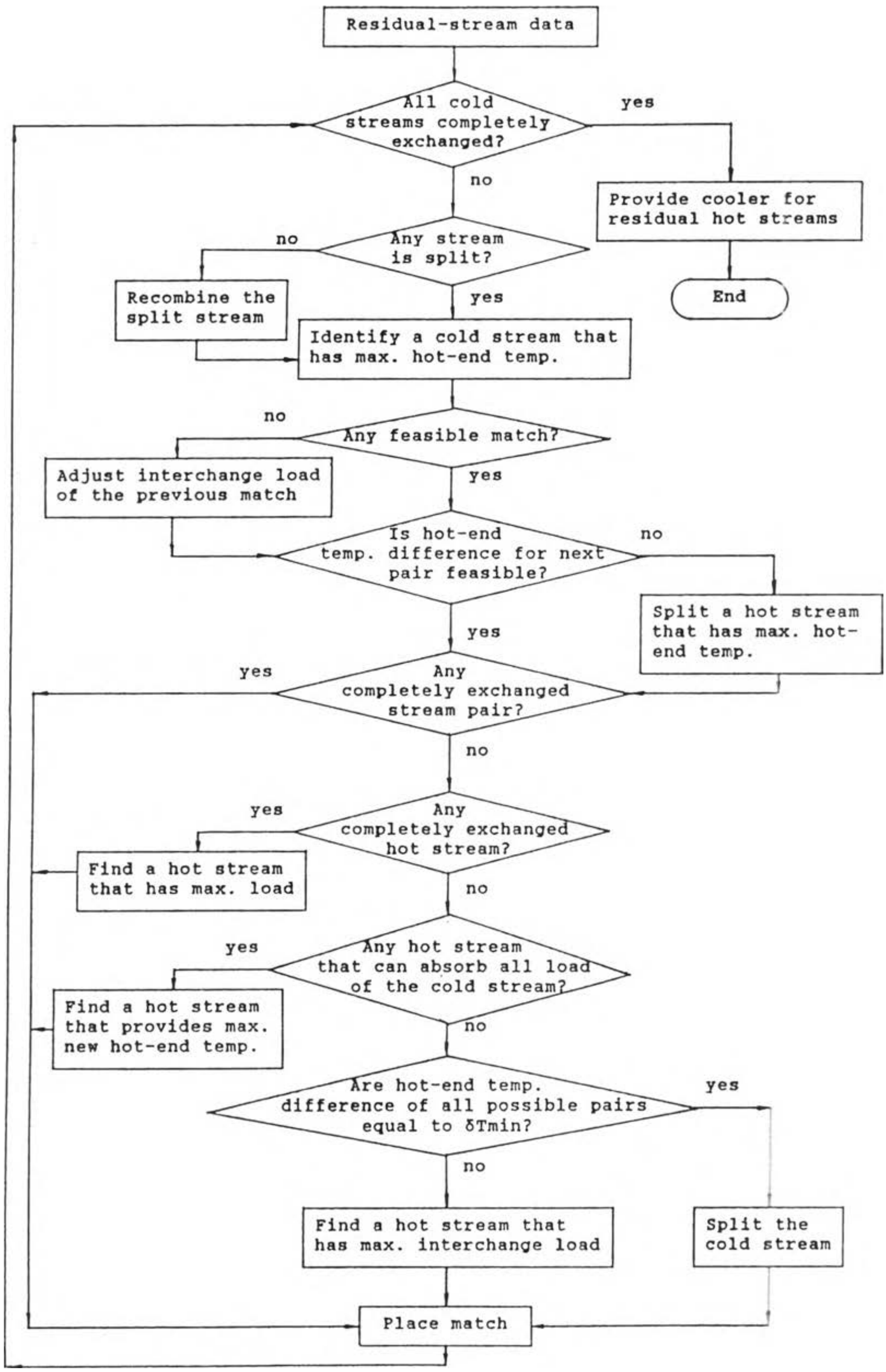

Figure 2.16 Heuristic procedure for below-the-pinch design

cold stream is to be made only when the CP criterion is violated.

Away-from-the-pinch design algorithm is shown in Figure 2.16. The selection of a stream pair for placing match has been described in Chapter 2, and the concept of each design step in the flowchart is a mirror image of that described in the RMATCHA.BAS module.

7.4.4 The BLOOP.BAS module

The flowchart of this module is shown in Figure 7.8. The module will automatically search for and break all loops up to the second-level. If the maxinum possible loop level is higher than 2, the program will tabulate the identification numbers of all units which lie on each stream. This should facilitate the identification of any higher-level loops by the user. The module will handle an the identified loop interactively and try to break it. If the loop is successfully broken it will automatically restart from the first level before proceeding further.

7.4.5 The ECONOMIC.BAS module

The flowchart of this module is shown in Figure 7.9. The required user-supplied input data are:

(1) Unit costs of the hot and cold utilities.

(2) Overall heat transfer coefficient of each exchanger unit.

(3) Present equipment cost index.

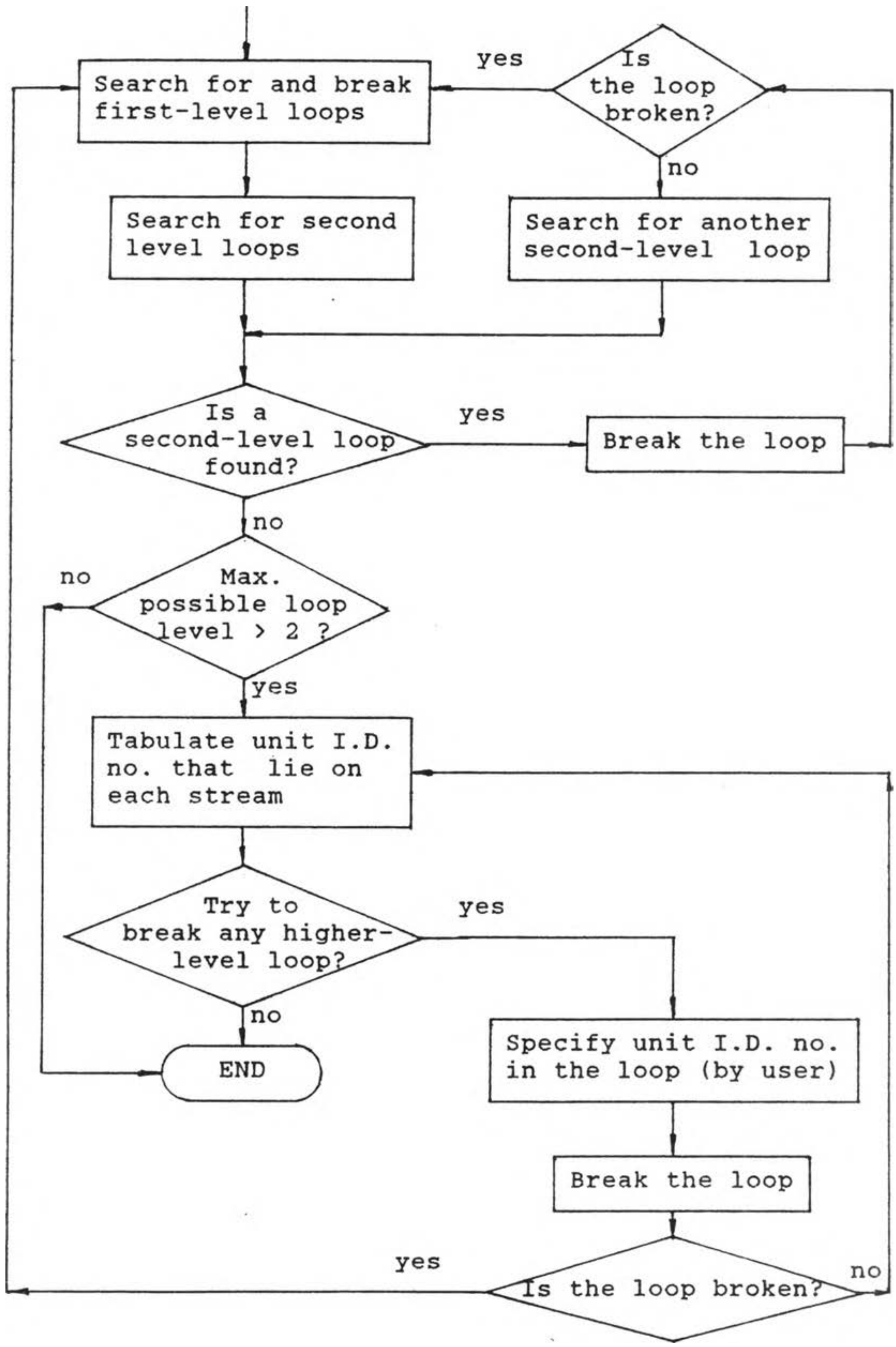

Figure 7.8 Flowchart of the BLOOP.BAS module

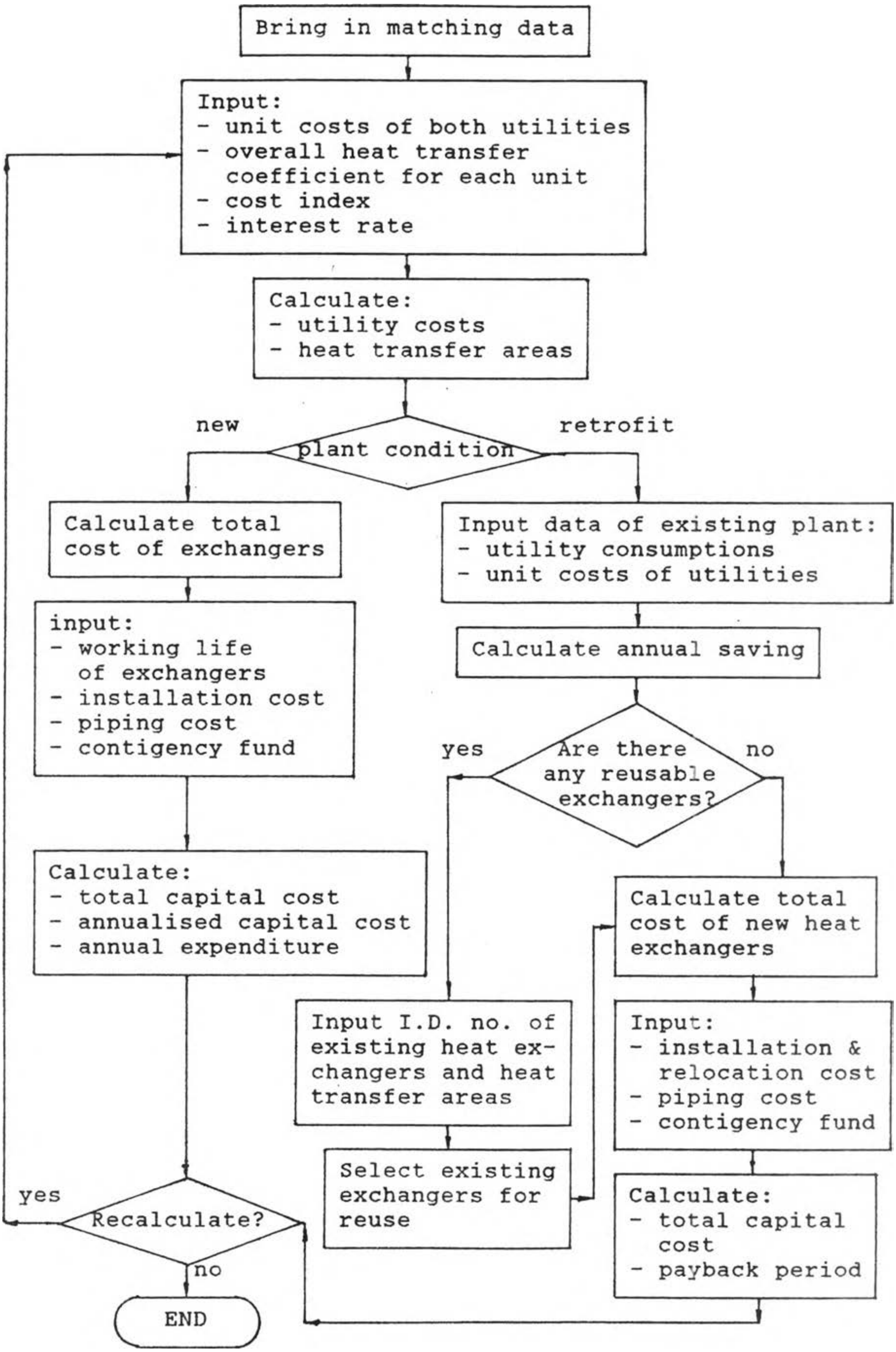

Figure 7-9 Flowchart of the ECONOMIC.BAS module

(4) Interest rate.

(5) Installation and piping costs as a fixed percentage of the exchanger cost or as a fixed amount.

(6) Working life of the exchangers (in case of a new plant only).

In case of a revamped network the following additional data are required.

(7) Energy consumptions and costs of utilities in the existing plant.

(8) Heat tranfer areas of reusable exchangers (if any).

The program will first calculate the required heat transfer area of each exchanger unit then calculate its cost and come up with the total investment cost. Finally the annualized cost, or payback period, is calculated.

For a revamped network, the net annual saving is the difference between the annual utility costs of the existing network and that of the revamped design. Only existing exchangers which have larger heat transfer areas than those required after revamping are considered for reuse.