Chapter V

Treatment of Loop in Heat Exchanger Network

5.1 Loop Level and Identification

Su and Motard [21: 67] define the "Nth-level loops" as the loops that involve N source streams and N sink streams. By this definition the first-level loops are those loops that involve one source stream and one sink stream. The second-level loops involve two source streams and two sink streams, and so on. The highest loop level possible in a network is the smaller of either the number of source streams or the number of sink streams.

For example, in Figure 5.1(a) the following loop can be identified.

first-level loop : (3,5)
second-level loop: (1,2,3,4)
(1,2,5,4)
(C1,5,4,C2)
(C1,3,4,C2)
(C1,2,1,C2)

The numbers in parentheses represent the identification numbers of heat exchangers, coolers and heaters, as shown in Figure 5.1. The highest possible loop level in this example is three because there are three source streams and three sink streams. However, no thirdlevel loops are found.

If the heaters and coolers of Figure 5.1(a) are drawn in the manner of Figure 5.1(b), it would be easier to identify the second-level loops that contain heaters and coolers.

Two important features about the loop level are

(1) The existence of higher-level loops does not depend on the existence of lower-level ones.

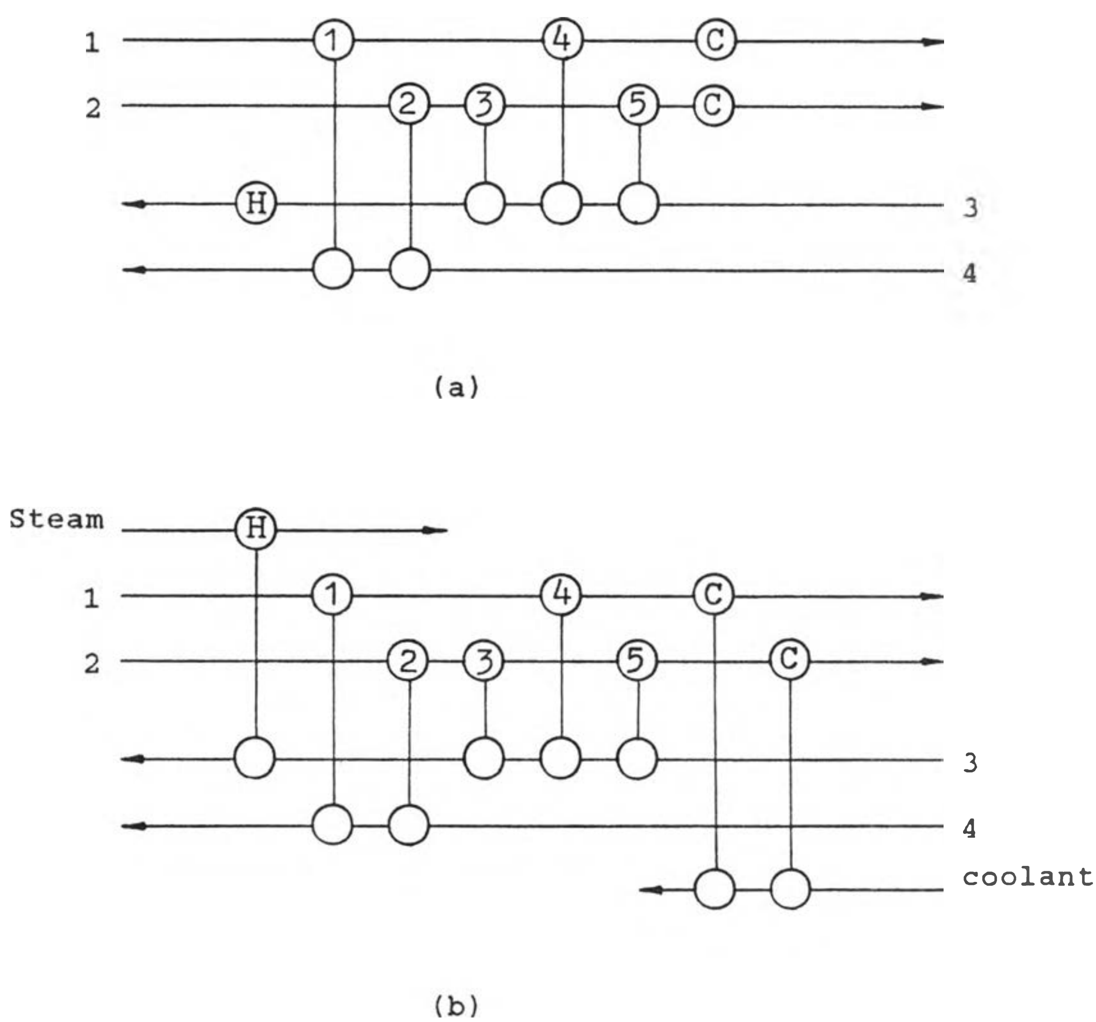


Figure 5.1 Example for identification of loop level

(2) The redistribution of energy in a loop at a specific level always influences the loops in the other levels.

The loop level provides information about how large a loop can be in a certain system. In addition, it also leads to a clear procedure for searching loops.

5.1.1 Loop identification procedure

In this thesis, the searching of strings of heat exchangers that form loops up to the second level is computerized. Any higher-level loop must be searched manually. The loop searching procedure is shown in Figure 5.2. All heat load paths which number of exchangers in the paths corresponding to the specified loop level will be searched. The first-level loop comprises of two exchangers, whereas it is four for the second level. If the first and the last exchanger of a path lie on the same stream, it means that the path causing a loop and the loop has been found.

5.2 Loop Breaking Procedure

Attempts are made to search for and break the loops from the lowest level up. The higher-level loops will not be broken until the lower-level ones have disappeared or cannot be broken any more. Once a loop of any level is broken, the whole procedure has to be restarted from the lowest level because the heat duties or temperature enviroments of lower-level loops have been

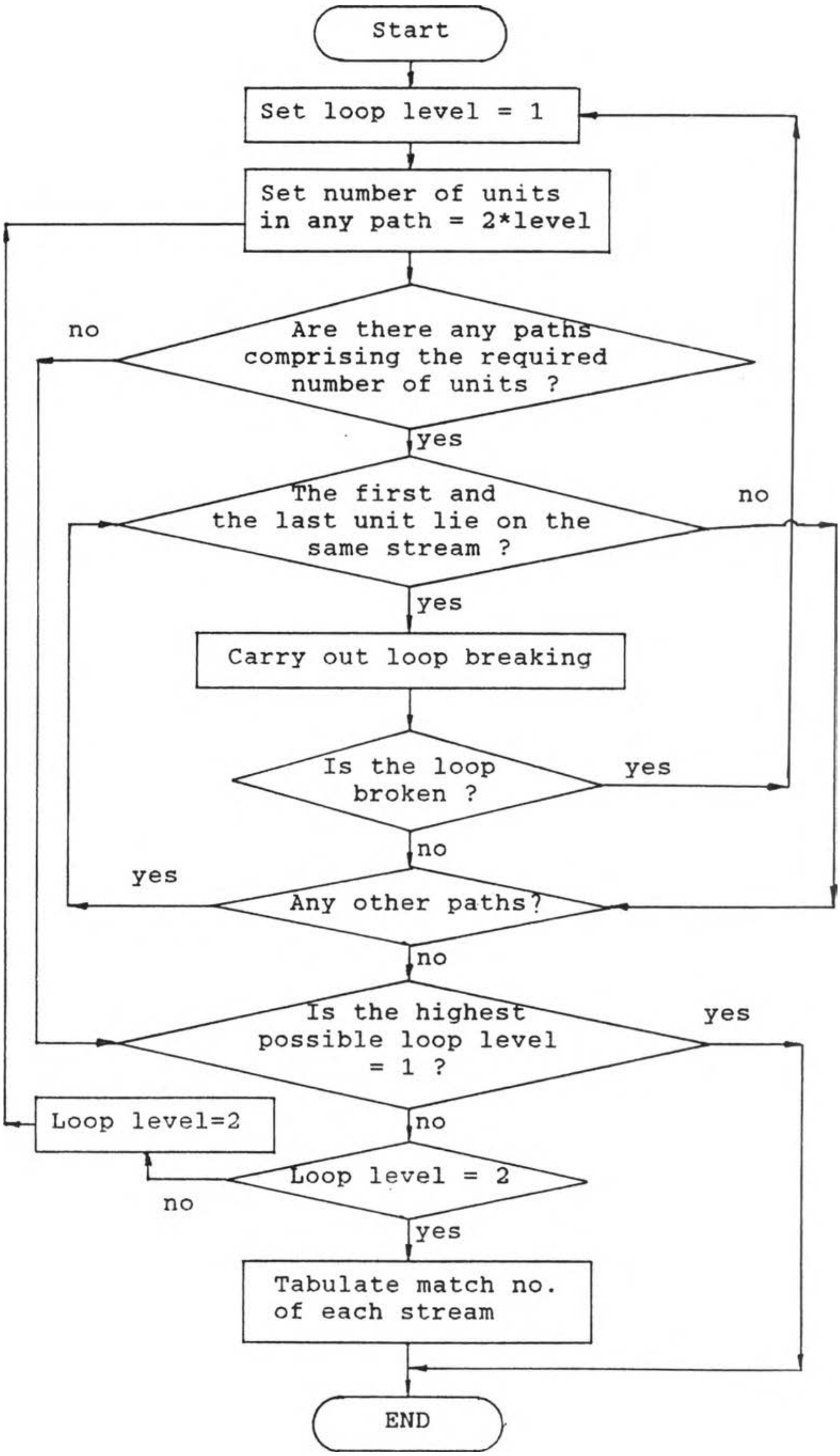


Figure 5.2 Loop searching procedure

changed and they are easier to identify. More importantly, some higher-level loops may disappear at the same time that a lower-level loop is broken. For instance, in Figure 5.1(a), if the first-level loop (3,5) can be broken by merging exchangers 3 and 5, then the second-level loops (1,2,3,4) and (C1,3,4,C2) will automatically disappear.

5.2.1. Primary loop breaking

The function of primary loop breaking is to break the loops, or to reduce the number of units as much as possible without stream splitting.

The primary loop breaking procedure shown in Figure 5.3 can be summarized as follows:

Step 1. Search for loops.

Step 2. Using the first element in the loop string as merging target, add (subtract) the amount of heat load of the merging target to (from) the heat load of each even (odd) element in the loop string consecutively.

Step 3. Check if there is any negative heat load in any unit of the loop string. If a negative heat load is found, the merging is infeasible. The other remining loops, or the same loop but with a different merging target will be searched.

Step 4. Calculate the temperature difference across each unit and compare it with the minimum allowable temperature difference to determine the feasibility of the new structure. Once a new feasible

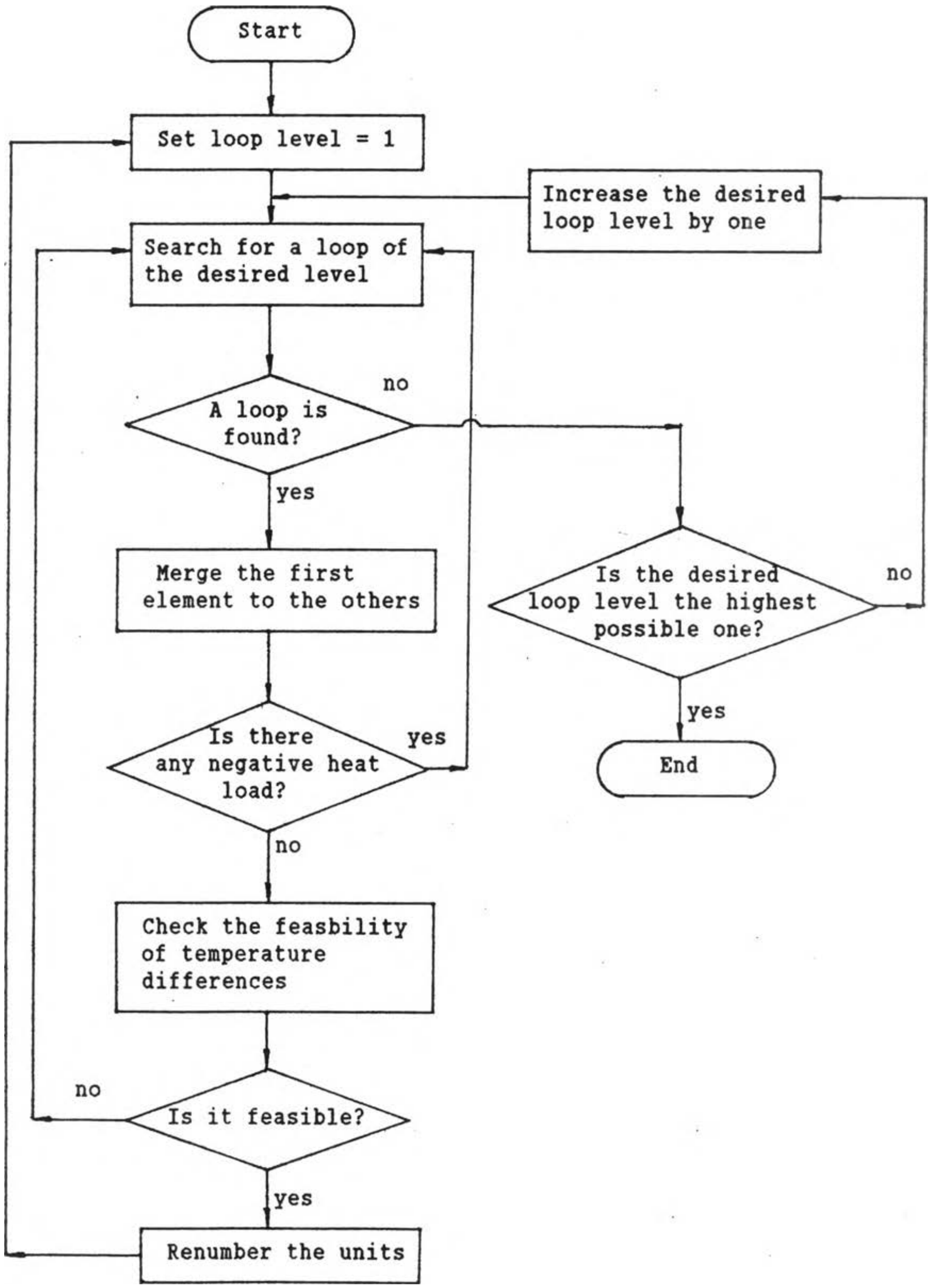
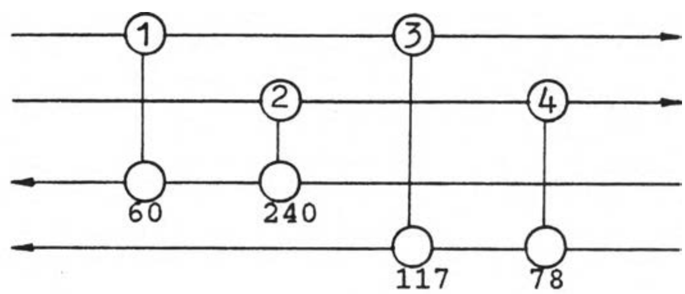


Figure 5.3 Primary loop breaking

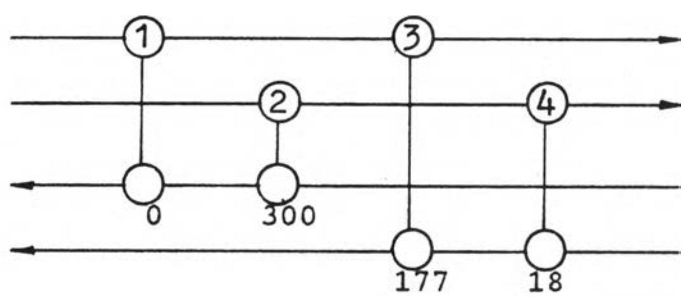
structure is evolved, the loop searching and breaking procedure must restart from the lowest level loop.

Figure 5.4 is used to illustrate the merging process. In structure (a), if the loop string (1,2,4,3) is identifed, the network will evolve to structure (b) after the merging process. Note that both the heat loads of unit 2 and unit 3 increases 60 units, respectively, because they are the elements in even positions in the

Structure (a)



Structure (b)



Structure (c)

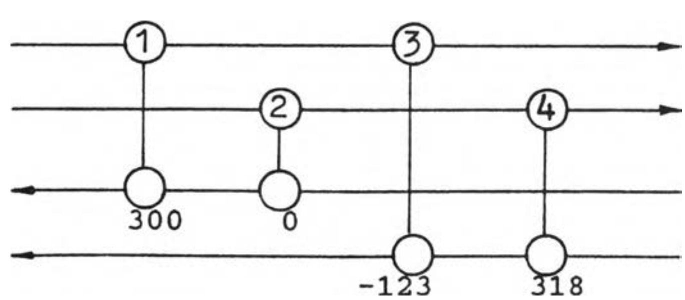


Figure 5.4 Example of the merging process

string. The heat loads of the odd elements (unit 1 and 4) decrease 60, and unit 1 disappears finally because its heat load becomes zero.

However, if the loop string (2,1,3,4) is identified, unit 2 becomes the merging target. The heat load of unit 3 turns out to be negative after the merging process, as shown in structure (c) of Figure 5.4. The merging is infeasible and loop string (2,1,3,4) has to be given up. We may search the other loops or the same loop but with a different merging target.

The whole primary loop-breaking procedure can be illustrated through the following example.

An initial structure that features maximum energy recovery is shown in Figure 5.5(a). The exchangers, heaters can be renumbered as structure (b).

There are no first-level loops to be found. Hence, the goal is set on the second-level loops. The evolution procedure works as follows

Merging target: 5

Loop: (5,1,3,6)

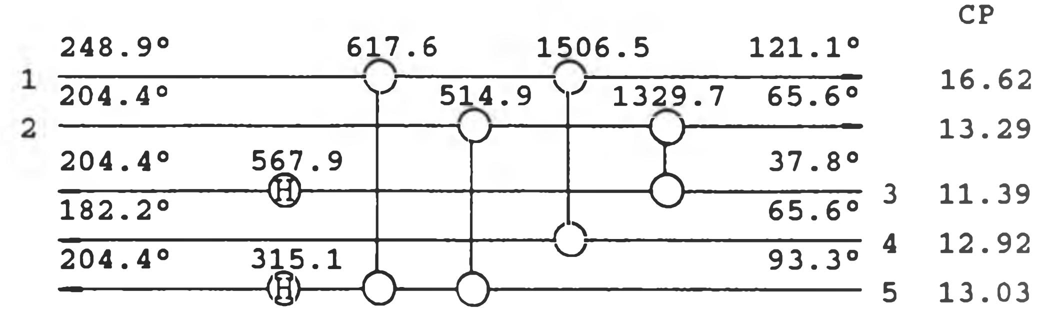
Merge: infeasible, negative heat load on unit 3.

Merging target: 1

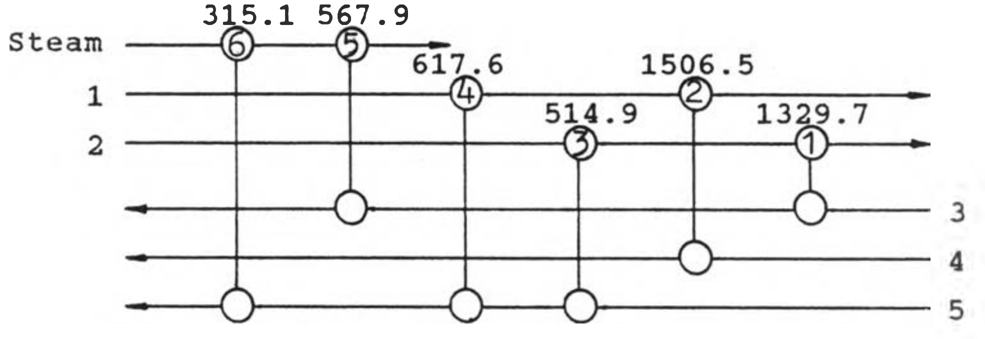
Loop: (1,3,6,5)

Merge: infeasible, negative heat load on unit 6.

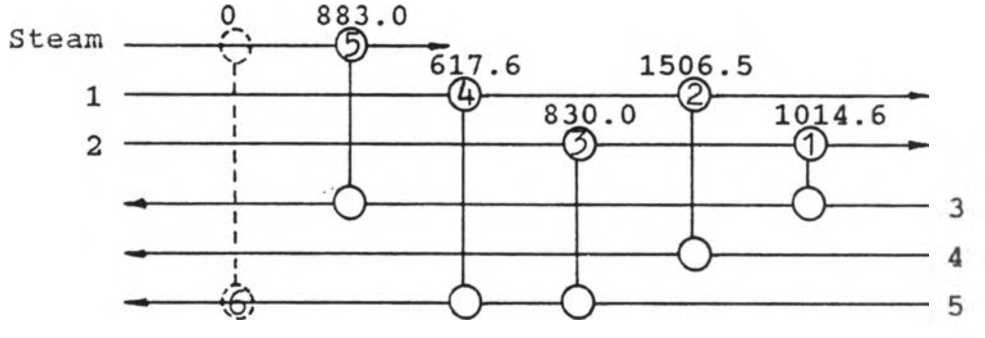
Structure (a)



Structure (b)



Structure (c)



Structure (d)

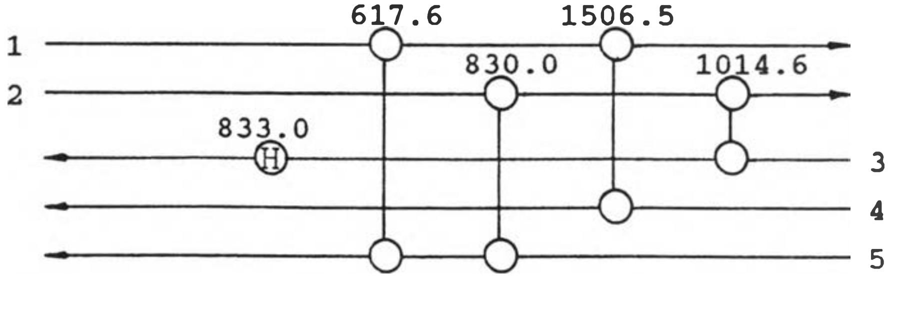


Figure 5.5 Example for primary loop breaking

Merging target: 3

Loop: (3,6,5,1)

Merge: feasible, no nagative heat load.

Temp. difference: violate the given δTmin on unit 1

Merging target: 6

Loop: (6,5,1,3)

Merge: feasible, no nagative heat load.

Temp. difference: feasible

Since both merging and temperature differences are feasible in the last instance, the network is thus evolved to structure (c) of Figure 5.5. Since one loop has now been broken, the loop-breaking procedure has to be restarted from the first-level. However, further manipulation shows that sturcutre (c) cannot be altered any more. Hence, the final result after primary loop-breaking is structure (c) of Figure 5.5, which can replotted as structure (d) of Figure 5.5.

For a large number of problems, the goal of the quasi-minimum number of units can be achieved by primary loop-breaking alone. If this fails, it often means that there are some loops that cannot be broken without stream splitting. Hence, a secondary loop-breaking is required. In the secondary loop-breaking process, the loop identification method is the same as before, but the merging process will incorporate stream splitting procedures.

Anyway, secondary loop-breaking is neither included in the developed computer program nor in the scope of this thesis. Interested researchers should refer to the literature [21: 67].