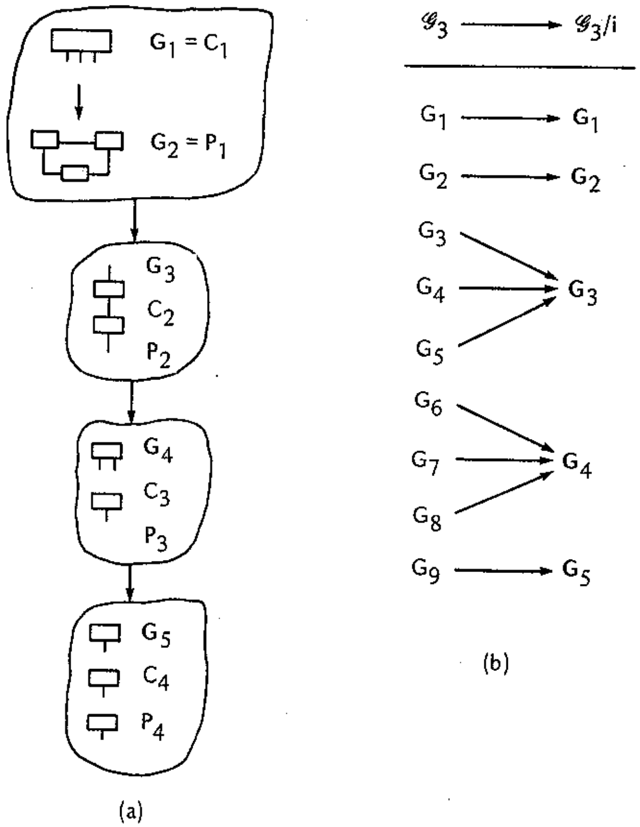

Figure 4.17. Lattice ($\mathscr{G}_3/i, \leq$) and the homomorphic mapping $\mathscr{G}_3 \to \mathscr{G}_3/i$.

C-structures in the permutation equivalence class expressed by that block diagram; the number placed next to each arrow indicates the number of immediate refinements of each *C*-structure of one permutation class in the other class. As explained earlier, this lattice is isomorphic with the lattices defined on $\mathscr{R}_4/i$, $\mathscr{P}_4/i$, and $(\mathscr{G}_4/i)/r$.

While the full lattices $(\mathscr{G}_n, \leq)$ represent the basis for the local level of computation in the reconstruction problem, the lattices $(\mathscr{C}_n, \leq)$ or their isomorphic counterparts are the basis for the global level of computation. To operate at the global computational level, a procedure is required by which all immediate refinements in the lattices $(\mathscr{C}_n, \leq)$ are generated for any given *C*-structure $C_k \in \mathscr{C}_n$ $(n \in \mathbb{N})$. One such procedure, which utilizes the graph representation of *C*-structures, is described as follows.

Refinement Procedure for C-structures (or *RC-Procedure*). Given a *C*-structure $C_k \in \mathscr{C}_n$ and the corresponding graph $r_n(C_k)$, to determine all immediate refinements of C_k in the set $\mathscr{C}_n$:

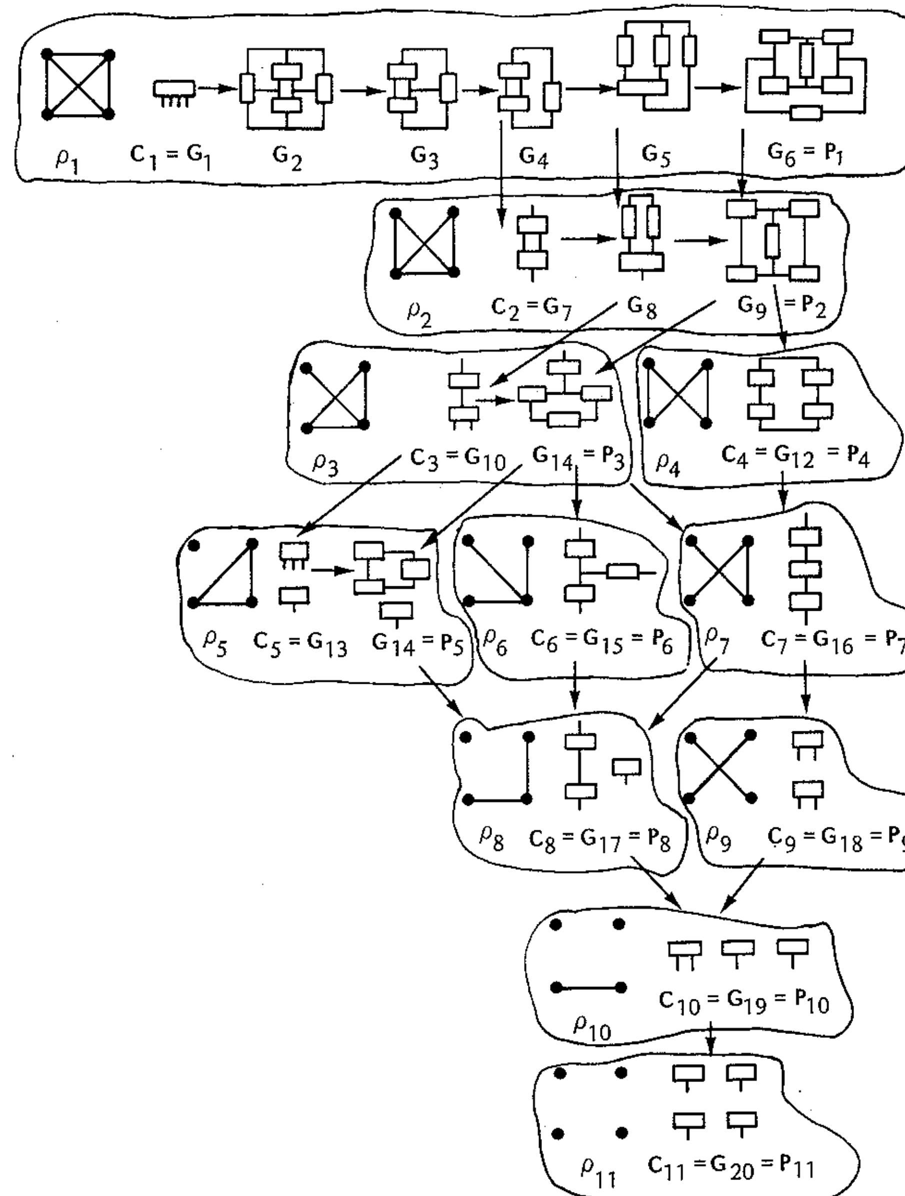

Figure 4.18. Lattice ($\mathscr{G}_4/i, \leq$) with the indication of r-equivalence classes and canonic *C*-structures and *P*-structures.

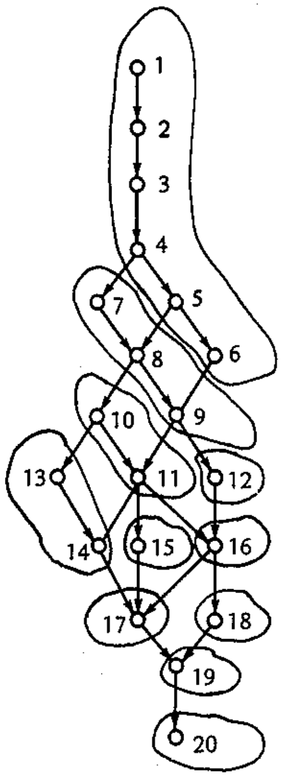

Figure 4.19. Summary of lattice ($\mathscr{G}_4/i, \leq$) described fully in Figure 4.18.

1. Exclude one edge from the graph $r_n(C_k)$, say edge (a, b);
2. split each element x of C_k that contains both a and b into two elements, $x_a = x - \{b\}$ and $x_b = x - \{a\}$, and replace x in C_k with x_a and x_b;
3. exclude all x_a's and x_b's generated in step (2) that are redundant and record the result as an immediate refinement of C_k in the lattice $(\mathscr{C}_n, \leq)$;
4. repeat steps (1)–(3) for all edges of the graph $r_n(C_k)$ and, then, stop.

The procedure is justified by the following facts: (i) there is a one-to-one correspondence between sets $\mathscr{R}_n$ and $\mathscr{C}_n$ and, hence, each change in a graph is reflected by a change in the corresponding C-structure; (ii) the smaller the number of edges in a graph, the more refined the corresponding C-structure is; (iii) since no loop on a vertex may be excluded from a graph without violating the covering requirement for the corresponding C-structure, the smallest possible reduction of the graph is to exclude one of its edges. The number of edges in this graph indicates thus the number of immediate refinements of the corresponding C-structure.

Example 4.18. Consider the graph ρ_1 and the corresponding C-structure C_1 specified in Figure 4.21a. The graph has six edges and, hence, there are six immediate

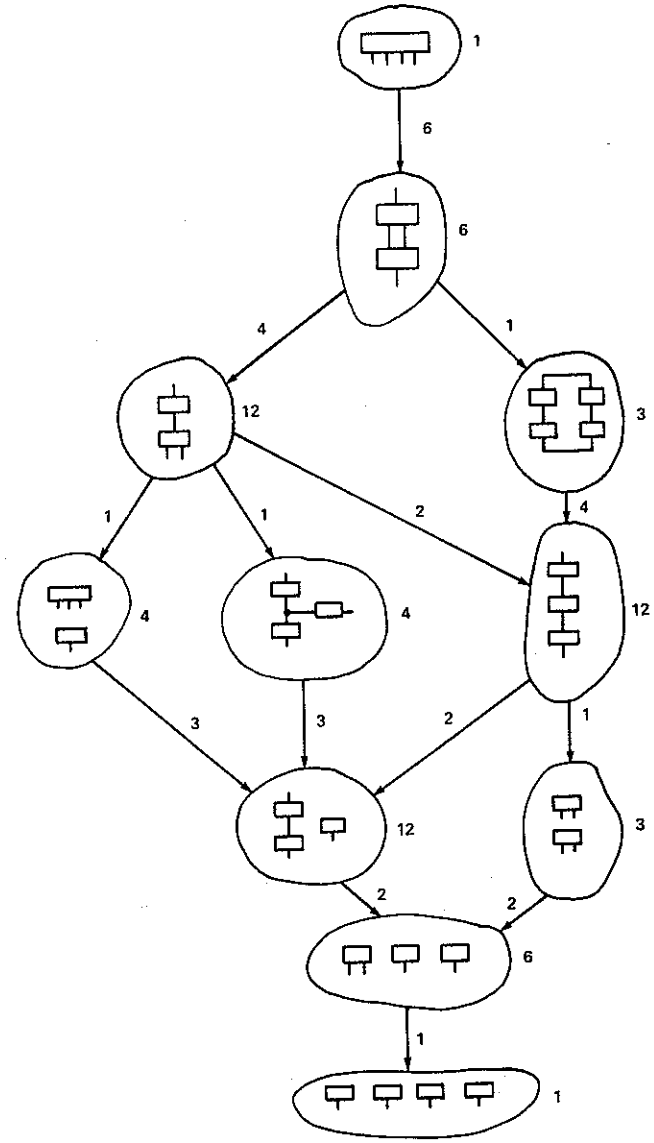

Figure 4.20. Lattice ($\mathscr{C}_4/i, \leq$).

refinements of the C-structure. They are shown in Figure 4.21b. The refinement C_7, for example, is derived by the RC-procedure as follows: (1) edge (4, 5) is excluded from ρ_1 so that graph ρ_7 is obtained; (2) element $\{2, 4, 5\}$ of C_1 (the only element of C_1 that contains both 4 and 5) is split into elements $\{2, 5\}$ and $\{2, 4\}$; (3) since the element $\{2, 4\}$ is the only redundant element $\{2, 4\} \subset \{2, 3, 4\}$), it is excluded and the result $C_7 = \{\{1, 2\}, \{2, 5\}, \{2, 3, 4\}\}$ is recorded as an immediate refinement of C_1.