

The Refitting of Plans by a Human Expert

Franz Schmalhofer, Christoph Globig, Jörg Thoben

German Research Center for Artificial Intelligence
University Bldg 57
Erwin-Schroedinger Str.
W-6750 Kaiserslautern

email: schmalho@informatik.uni-kl.de

Abstract. During the course of the development of a Case-Oriented Expert System for situated applications additional cases were needed. The required cases were obtained by having a human expert refit old solutions to new problems and the structural relations between source and target cases were analyzed: A higher degree of reuse of the old cases was found when the expert could apply derivational reasoning and a uniform design rationale (i.e. the solution of the source was generated by the expert himself) than when the expert could only analyze structural relationships (i.e. the source solution was constructed by some one else). Except with very obvious cases, it was also found, that different experts perceive different cases as the most similar source to a given target problem. The results also indicate for user-situated applications of expert systems.

1. Introduction

In order to overcome the brittleness of first generation expert systems, it has recently been proposed to develop Case-Oriented Expert Systems (COEx-Systems), which allow situated applications (Schmalhofer & Thoben, 1992). One prerequisite for developing such a system is that a sufficient number of prototypical cases are available for the desired competence of the system. Since originally we had only very few cases, we had an expert generate solutions to additional prototypical problems by having him refit old solutions, so that they would become solutions for those problems.

The current paper first reviews the integrated knowledge acquisition method (Schmalhofer, Kühn & Schmidt, 1991) for COEx-Systems together with their general characteristics. We then present a structural analysis of the refitted plans. Finally several conclusions with respect to the development of expert systems and the situated applications of old cases are drawn.

2. Case-Oriented Expert Systems for Mechanical Engineering Planning Tasks

In the knowledge acquisition phase for such COEx-Systems, model-based abstractions are formed from concrete past experiences, so that they can be reused in novel situations. Human expert judgments concerning the classification and similarities of the concrete past experiences are applied to obtain an abstraction hierarchy of problem classes (Bergmann & Schmalhofer, 1991; Schmalhofer, Reinartz & Tschaitshian, in press) and supplementary knowledge from

written materials is used to obtain explicit operator definitions (Schmidt, 1992) so that associated skeletal plans can be constructed (Bergmann, 1992; Friedland, 1985).

The knowledge acquisition for such systems thus yields an abstraction hierarchy of problem classes with associated skeletal plans which allow for a situated utilization of past experiences in future tasks. During the knowledge acquisition phase, these past experiences have been interpreted by one or several experts within some uniform rationale. More details about such systems can be found in Schmalhofer & Thoben (1992). The respective knowledge acquisition procedures and tools were summarized by Schmalhofer, Bergmann, Kühn & Schmidt (1991). The model of expertise or problem solving model (Breuker & Wielinga, 1989), which underlies COEx-Systems for planning tasks has been described by Kühn & Schmalhofer (1992).

Our research group has recently been developing such a system for production planning problems in mechanical engineering. Without going into any details of this application domain, we can state that production planning is a typical planning problem: For example, the mold of the workpiece defines the given state and the goal workpiece defines the goal state of the manufacturing problem. A number of different types of operations (chucking, unchucking, cutting operations) are available for transforming the mold (given state) to the goal workpiece (goal state). The operations themselves are quite complex requiring the specification of a number of different parameters (such as cutting path specification, specific cutting parameters, toolholders, etc.). It is therefore very useful to classify and abstract operations to different types of macro-operators.

		Workpiece																			
		Drive Shaft g1				Drive Shaft g2				Pinion Shaft g3				Axle Shaft g4				Axle Shaft g5			
Geometry		w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4
Workpiece material																					
Manufacturing machines (lathes)	d1					3	3	1		3	3	1		4	4	2		4	4	2	*
	d2	5	5	7	9	5	5	7	9	m3				6	6	8	10	6	6	8	10
	d3	m1				m2								m4				m5			
		*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Table 1 (after Schmalhofer & Thoben, 1992): A number of specific problems are used in order to delineate the competence of the future expert system. From the factorial combination of three types of manufacturing machines (d_1, d_2 , and d_3), and workpieces with five different types of geometries (g_1, g_2, g_3, g_4 , and g_5) and materials (w_1, w_2, w_3 , and w_4) fifty-two problems were identified as meaningful. The numbers 1 to 10 indicate the abstract problem classes to which a specific prototypical problem belongs. An abstraction hierarchy for these problem classes is shown in Figure 1. See text for further explanation.

Since a production plan strongly depends upon the specific geometry of the workpiece (g), the workpiece material (w), and the particular machine (d), which are to be used when manufacturing the workpiece, we denote production problems with the descriptors g , w , and d . By using different indices with these descriptors we can thus refer to a given manufacturing problem.

In Table 1 sixty production problems are specified through the factorial combination of 3 manufacturing machines (d_1 , d_2 , and d_3), five different geometries (g_1 , g_2 , g_3 , g_4 , and g_5) and four different workpiece materials (w_1 , w_2 , w_3 , and w_4). Fifty-two of these problems (all problems whose cells are marked by a number between 1 and 10) are the prototypical problems, which delineate the desired competence of the future expert system. Problems with the same number were assigned to the same abstract problem class. The abstraction hierarchy of these ten abstract problem classes is shown in Figure 1.

Since only five production plans were originally available for the 52 prototypical problems, i.e. the cases m_1 , m_2 , m_3 , m_4 , and m_5 (see Table 1), an expert refitted these plans (refitting roots) and his subsequently generated plans (refitting children) for 16 of the 52 prototypical problems. He also constructed one production plan from scratch ($g_5w_4d_1$). In Table 1, the problems with associated refitted plans are indicated by the asterisks.

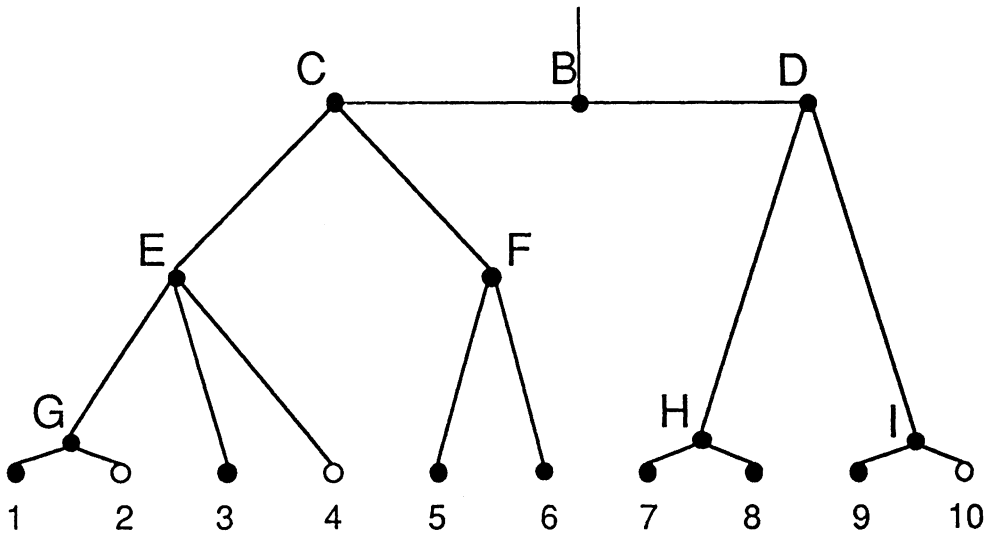


Figure 1: Shown is the problem abstraction hierarchy for 10 abstract problem classes (see Table 1)

3. Plan Refitting

Figure 2 identifies the different source cases which the expert used for finding solution plans for the 16 target problems: The source-target case relation is indicated by an arrow. Whereas case m_3 was five times used as a direct source, the cases m_1 , m_2 , and m_4 were each only used once as a direct source and case m_5 was never used as a source. On 8 occasions one of the cases which had already been tested in the real world (tested source case or refitting root) were used as source and 8 times a solution plan which the expert had generated himself (i.e. a refitting child) was used as source (self-generated source case).

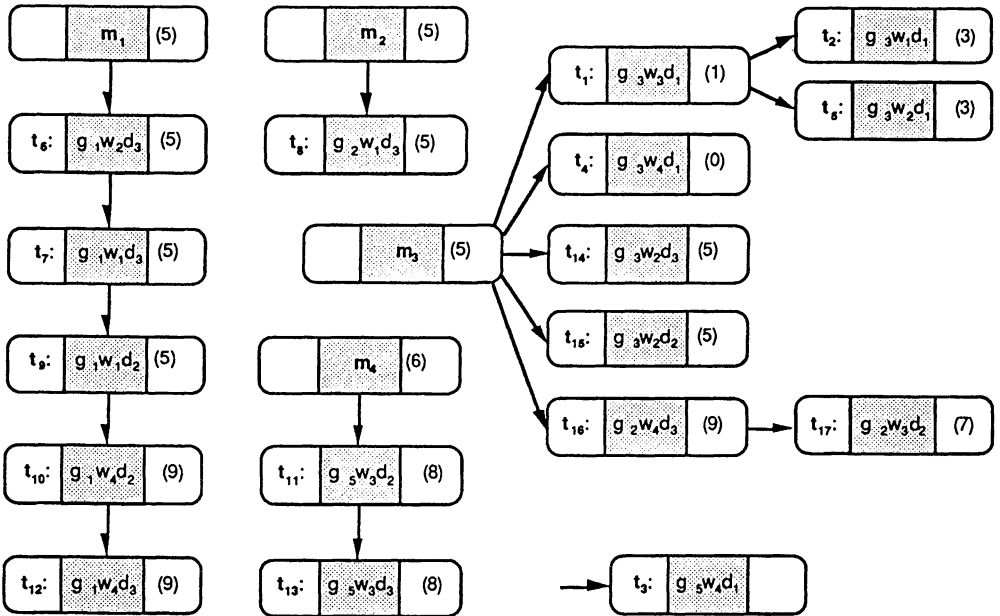


Figure 2 (after Schmalhofer & Thoben, 1992): The source case - target case relation is shown for the 17 tasks (t_1 to t_{17}) which were solved by the expert. In parentheses the abstract problem class that a specific task is associated with (see Figure 1) is noted. Task t_3 was solved from scratch, so that there is no source case associated with it. Whereas the cases m_1 , m_2 , m_3 , and m_4 served as refitting roots, the other cases are denoted as refitting children.

The task numbers t_1 to t_{17} indicate the temporal order in which the 16 refitting and the one plan construction task (t_3) were performed by the expert. These numbers show that the immediately preceding target solution was very often used as the source for the next target problem. For example the solution to task t_6 was used as a source for t_7 and the solution to t_{14} was used as the source for t_{15} . On other occasions somewhat earlier preceding target solutions were used as the source for the current target problem. For example, the last but one target solution was used as the source for task t_{13} . These temporal relationships indicate that for the refitting of old plans, the expert tried to maintain a fresh memory of the modification processes by which he constructed the old plan.

When the expert remembers his reasoning (i.e. the derivations), by which he constructed or modified the old plan, he can perform derivational refitting processes (Carbonell, 1986). When the old cases were generated by somebody else, as for example the tested source cases m_1 , m_2 , m_3 , m_4 and m_5 (i.e. the refitting roots), the expert is more likely to perform only structural refitting processes (Hammond, 1989). Another important observation was: The plans which were obtained by modifying an already existing plan were completed by an order of magnitude faster than the plan which was produced from scratch (t_3).

3.1 Different Types of Modifications between Source and Target Plans

We also analyzed more detailed structural relations between the source and target plans. Thereby it was distinguished between the refitting of tested source cases (i.e. refitting roots), where the expert was very likely to only use structural analogies and the refitting of self-

generated source plans (i.e. refitting children), where the expert could at least to a certain degree also apply derivational analogies.

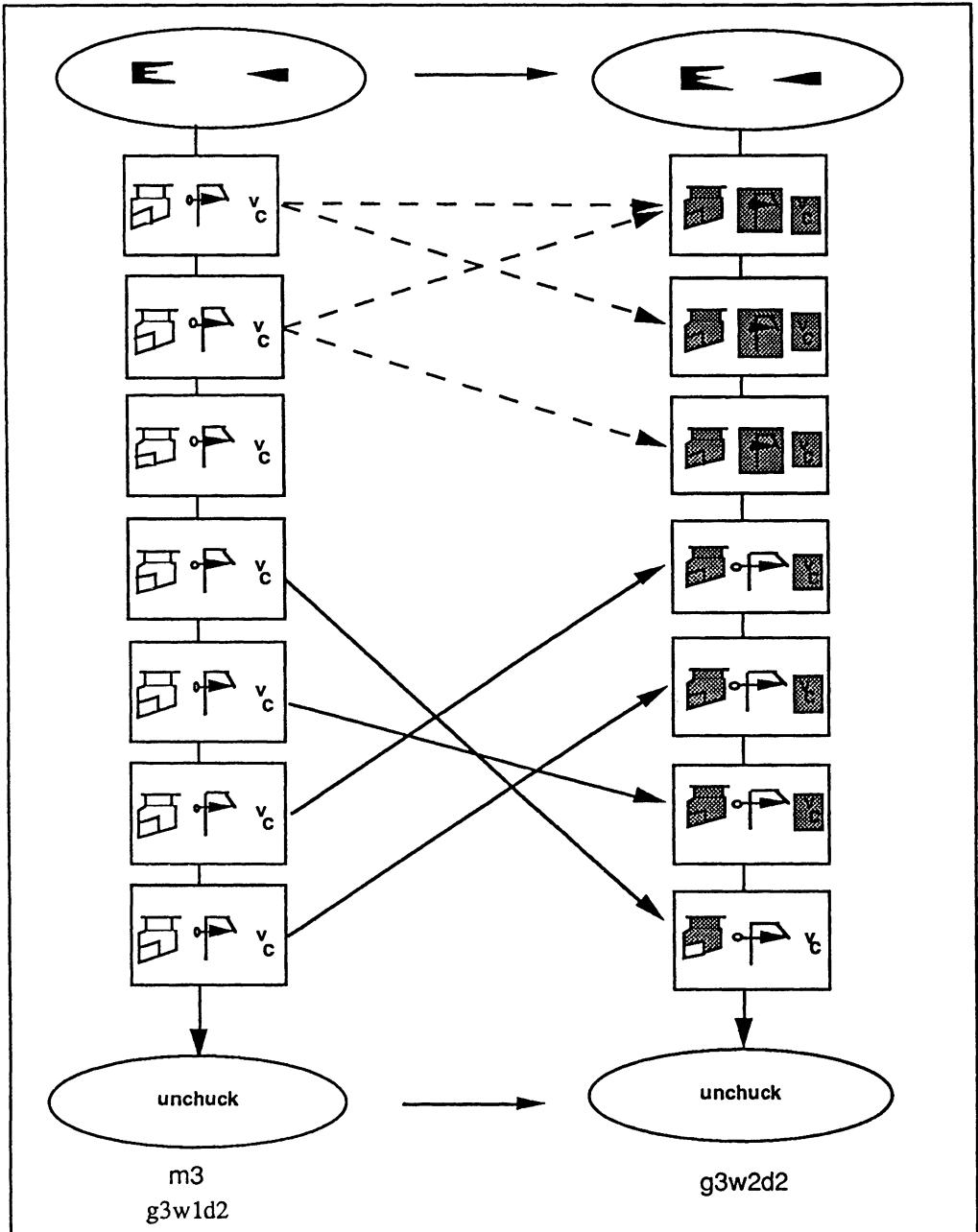


Figure 3: Shown are the structural relations between the tested source case m₃ for problem g_{3w1d2} and the resulting target plan for problem g_{3w2d2}. See text for further explanation.

Figure 3 shows the structural relationships between the operators of the tested source case m₃ and the refitted plan for the manufacturing problem g_{3w2d2}. The Figure shows the structural

relationships between corresponding operators of the source plan m_3 for problem $g_3w_1d_2$ and the target plan for problem $g_3w_2d_2$ at the macro level. The ovals represent chucking and unchucking operations. All cutting (macro-)operations are indicated by rectangles. Within these rectangles, 1) the toolholder together with cutting tool, 2) the cutting path, and 3) the cutting parameter v_c are symbolically represented from left to right. Shaded symbols in the target plan indicate changes from the source to the target. The solid lines with arrows indicate which operations of the source were reused in the target plan. The dashed lines indicate substantial changes in the individual operations themselves.

The first two cutting operations of the source plan (see left side of Figure 3) were splitted apart and the resulting components were rejoined across the original operations of the source plan. Two new operations were thus created, which differ in all three parameters from the operations in the source case (see right side of Figure 3). As a consequence, the third cutting operation of the source was completely eliminated from the target.

The execution order of the fourth and the fifth cutting operation of the source was also changed in the target. While the cutting path remained identical, cutting tool and cutting parameters were adjusted to the new workpiece material. The same modification was performed for the sixth and the seventh operation, except that these operations were not reordered in the target plan.

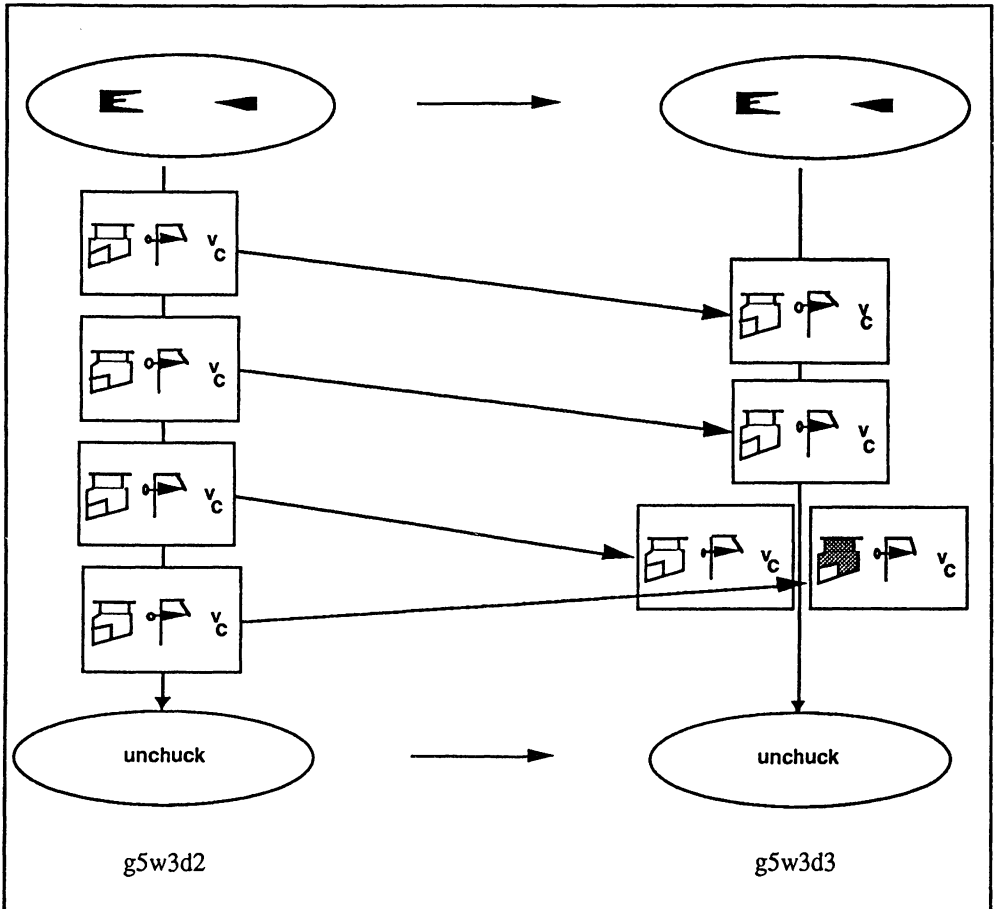


Figure 4: Shown are the structural relations between a self-generated source and the resulting target case. See text for further explanation.

Figure 4 shows the structural relationships between the self-generated source plan g_{5w3d2} and a target plan which is refitted for a machine which allows parallel processing. Whereas the chucking operations as well as the first two cutting operations remain identical, the third and fourth operations of the source are now executed in parallel in the target plan. In addition, one of the toolholders is changed. This source-target pair thus shows a large degree of reuse of the operations and the execution sequence of the old plan.

3.2 Comparison of Structural Relations among Four Different Plan Pair Groups

We compared the structural relations among four different groups of plan pairs. The first group consisted of the 8 pairs, which contained tested source cases ($t_1, t_4, t_6, t_8, t_{11}, t_{14}, t_{15}$, and t_{16}). The second group of plan pairs contained the 8 pairs with self-generated source plans (t_2 ,

type of change	The 16 actual modification tasks		The 11 most similar case pairs	
	source is refitting root	source is refitting child	source is refitting root	source is refitting child
additional chuckings	0.25	0.13	0.00	0.14
eliminated chuckings	0.00	0.13	0.00	0.14
new parallel executions	0.13	0.25	0.25	0.29
new serial executions	0.13	0.25	0.25	0.14
splitted operations	0.75	0.00	0.50	0.00
joined operations	1.13	0.00	1.25	0.00
reordering of operations	0.75	0.50	1.00	0.43
cutting path changes	3.75	0.25	6.75	1.43
cutting parameter changes	5.63	4.25	8.75	2.86
toolholder changes	4.88	2.50	8.50	1.57
cutting tool changes	5.25	4.00	8.00	2.86
total number of case pairs	8	8	4	7
total number of cuts in source	61	57	34	46
total number of cuts in target	53	55	36	41

Table 2: Average number of different types of changes from the source to the target case for the 16 performed modification tasks (see Figure 2) and 11 most similar case pairs from the abstraction hierarchy.

t5, t7, t9, t10, t12, t13, and t17). The abstraction hierarchy of problem classes (see Figure 1 and for more details Schmalhofer & Thoben (1992)) was used for defining the third and fourth group of plan pairs. More specifically, for each of the 16 target plans, the most similar plan according to the abstraction hierarchy was selected as a hypothetical source case and the structural relations of these case pairs were analyzed. Group 3 contains the plan pairs, where the source plan was m_1 , m_2 , m_3 , or m_4 (i.e. the refitting roots): $m_3 - g_1w_1d_3$, $m_1 - g_1w_1d_3$, $m_2 - g_2w_1d_3$, $m_3 - g_3w_2d_2$. Group 4 contains the plan pairs, where the source plan was a self-generated plan: $g_3w_3d_1 - g_3w_1d_1$, $g_3w_3d_1 - g_3w_2d_1$, $g_1w_1d_3 - g_1w_1d_2$, $g_1w_4d_2 - g_1w_4d_3$, $g_5w_3d_2 - g_5w_3d_3$, $g_5w_3d_2 - g_2w_3d_2$, $g_1w_4d_3 - g_2w_4d_3$.

The results of this analysis are shown in Table 2. In general, fewer structural changes were observed between the (real or hypothetical) source and the target case, when the source case was also self-generated (i.e. a refitting child) than when the source case was generated by somebody else (i.e. a refitting root). And as expected, changes of the operations themselves occurred less frequently than parameter changes (e.g. cutting parameter changes).

3.3 Assessing the Expert's Consistency in the Source-Case Selections

In order to assess the expert's consistency in selecting the same source case as the most similar one to a given target problem, further data were collected from the expert who had performed the 16 refitting tasks (HW). In addition an additional expert (RL) had to perform the same task. The task consisted in selecting the most similar source from the cases m_1 , m_2 , m_3 , m_4 and m_5 to each of the 16 target problems, for which a plan modification was performed. In addition, the similarity between the source and target problem had to be estimated by a number between 1 and 7. Whereas 1 meant the lowest similarity, 7 indicated the highest possible similarity. Table 3 shows the results in comparison to the actually used source case. For self-generated source cases, the refitting roots (see Figure 2) were also determined.

From Table 3 it can be seen that the cases which were identified as most similar by HW correspond in only 50 percent to the actually selected source case or root of the source case (i.e. the refitting root) in the refitting task. There is also only a 47 percent consistency between the two experts. However, when only those cases, which were identified as most similar with a similarity rating of 7 are considered, the two experts agreed in 100 percent of the cases. More details have been reported by Thoben, Schmalhofer & Reinartz.

4. Conclusion

Our main purpose for having an expert refit old plans to new problems was to obtain a sufficient number of cases for developing a Case Oriented Expert System for production planning in mechanical engineering. Although there is now a sufficient number of cases available for constructing skeletal plans for the important set of medium level problem classes (i.e. for all classes with a solid node in Figure 1), further prerequisites must be satisfied. Unlike case-based reasoning which does not make such strong prerequisites, Case Oriented Expert Systems require that all prototypical cases follow the same design rationale. This requirement arises from the fact, that several layers of more and more abstract skeletal plans are to be constructed from these cases, so that deductive justifications will exist for the resulting state and operator sequence abstraction mappings (Bergmann & Schmalhofer, 1992). We will consequently have to test, whether the cases of the refitting roots (m_1 , m_2 , m_3 , m_4 , and m_5) follow the same design rationale as the cases generated by the expert HW.