4th International DAAAM Conference
"INDUSTRIAL ENGINEERING – INNOVATION AS COMPETITIVE EDGE FOR SME"
29 - 30th April 2004, Tallinn, Estonia

IMPLEMENTATION CBR SYSTEM FOR COMPUTER-AIDED DESIGN OF WORK-HOLDERS

Rein Küttner Grigori Nekrassov

Abstract: *The paper describes an approach for computer aided design of modular work-holders. This approach involves the creation an appropriate design environment, which fosters communication between various knowledge sources involved in work-holder design. The design environment is expected to accommodate different modeling paradigms, knowledge- and databases within a single, integrated framework, a design space (DS). The DS model is based on the use of decomposition and categorization methods. The fact is that much of work-holder designs are not very different from what has been done in the past. The case base reasoning approach (CBR-tool) for a solution of problems is applied. To show how the proposed technique might be useful, we describe a specific instantiation of this approach, in a prototype system for design of work-holders.*

Keywords: *work-holder design, design space, case base reasoning*

1. INTRODUCTION

The paper describes an approach for computer aided design of modular work-holders (CAWD). This approach involves the creation an appropriate design environment, which fosters communication between various knowledge sources involved in work-holder design. The computer-based work-holder design environment is expected to accommodate different modeling paradigms, knowledge- and databases within a single, integrated framework, a design space (DS). In performing a work-holder design task designers use a variety of engineering-related models such as functional, analytical, geometric (solid) models, etc.

The DS model is based on the use of decomposition and categorization methods which represent for the given product family breakdown structure in terms of product functions (abstraction hierarchy) and relations between product components.

The DS model consist of:

- A feature-based representation (database - DB) of components of work-holders (WH) (primitive and complex) which have its own syntax of description and semantic explanations and are described by geometric, dimensional, material etc. properties;
- Indexing method for retrieving from DB of WH's and their components, to stream line the matching process between new design task and existing solutions on the DB;
- A similarity metric to measure the similarity between a new desired work-holder or components and the existing in DB representation of previous designs and used components.

The case base reasoning approach (CBR-tool) for a solution of problems is applied (Aamodt & Plaza, 1994). A CBR system solves new problem by adapting solution that were used to solve previous similar problems.

The DB of the DS (case base) holds for the given product family a number of cases each of which represents a problem description together with its corresponding solution. Once a new problem arises, a possible solution to it is obtained by retrieving similar cases from the DB according to the specification of the new design task and studying and if necessary modifying recorded solutions. Every time that a new design is solved a new experience could be retained and made available for future use. The process of retaining is controlled by the supervisor of the system. A very important feature of CBR is its coupling to learning. It denotes the use of a new computer-learning paradigm in design process. To divide the responsibilities and manage the CBR process for a design environment, the two level decomposition of DS was proposed in (Kuttner et al., 2003) and is used to develop a DS. The tasks of DB management and activities of learning from previous design are tasks of an upper or supervisory level.

The design task of a work-holder starts with a design problem description and includes the specification of the main features of new design problem. In the first step the previous design solutions similar to the new problem specification are searched from DB. If in DB exist no suitable WH model, the problem description is decomposed according to the abstraction hierarchies and the models of suitable components are retrieved. In praxis for most cases of WH design there are always some models suitable for use. In some cases new model must be designed using the capabilities of basic CAD system (Solid Edge in our case).

To show how the proposed technique might be useful, we describe a specific instantiation of this approach, in a prototype system for design of technological equipment (work-holders).

The problems of work-holder design are a complex and highly experience-dependent tasks. A design environment for CAD of work-holders is characterized with the following features:

- A large solution space.
- A variety of input data for many knowledge sources.
- A multiple of company-specific standards, classification schemes etc exist.

The fact is that much of work-holder designs are not very different from what has been done in the past. There are obvious benefits in cost and time saved if the best practices are captured and made available to use.

2. MAIN PRINCIPLES OF MODELING OF AN INTEGRATED DESIGN SPACE FOR CAD

CAD process can be viewed as automated search in DB for those solutions that satisfy the requirements and are in some sense best among feasible alternatives.

The ability to efficiently save, index, and retrieve alternative models has become critical in a wide range of applications, including CAD systems, indexing schemes for large component inventories, access methods for "smart catalogues," and for performing searches through databases and on the Internet.

In solving the referred tasks we are interested in solving the following related problems:

- Given is a collection of models of WH and their components, it is necessary to estimate which ones are similar to each other and how similar are they?
- Given is a set of characteristics of new design task, what kinds of WH and/or its components we are interested in, how to retrieve suitable models from a system DB?

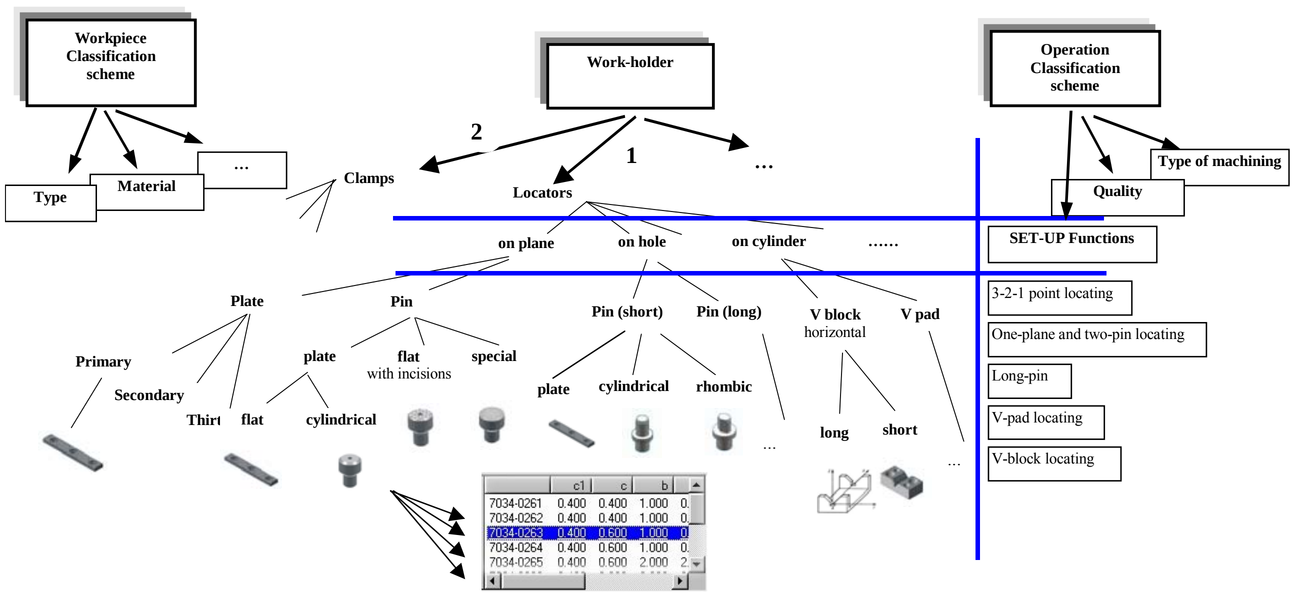


Figure 2. The generic structure of the WH's components and their functional groups.

During last years, a number of efforts have been made to develop algorithms to examine CAD designs and extract features that correspond to functionalities of designed product. Many classification schemes have been developed based on the idea, to capture critical features of a product in an alphanumeric string, for example the GT codes (Mitrofanov, 1966). GT coding was intended to be human interpretable and has caused some difficulties in use these codes in CAD.

There are two main difficulties to solve the proposed task: dealing with the combinatorial explosion of the design space and handling large amount of domain dependent knowledge. To solve the proposed task a hierarchic classification schema of features is proposed.

The top node in the hierarchy is the description of characteristics of a product family. The bottom level entities are the multitude of models of components that could be used to construct the WH. The components of a WH can be primitive (non-decomposable) or they may complex, consisting different subcomponents and corresponding structure.

The total classification tree for a product family could be represented by composing the general classification scheme for main functionalities of a product family and hierarchic classification schemas (abstraction hierarchies) of components. The structure of the functional hierarchy for WH consists (Rong & Zhu, 1999): clamping components, supporting components, fastening components, guiding elements, accessories etc. According to this approach a two levels of abstraction is used: the functional hierarchy and for each functional group of components the hierarchy of geometrical and dimensional information (abstraction hierarchy).

Abstraction hierarchies contain two types of relationships:

- The "***type-of***" (is-a) relationships that specify whether a call is a special case of another class. Abstraction hierarchy is an important issue of modeling the CAD domain data and knowledge. The main purpose of use the abstraction hierarchies, is to generate small groups of components that demonstrate similar characteristics. The type-of relationships help to decompose the DS into the subtasks and to estimate the similarity of components, to determine what data and knowledge should be applied.
- *Decomposition hierarchies* described by the "***part-of***" (consist) relationships that specifies what components or parts are assembled into an assembly (component of higher level). Figure 2. shows example of the substructure of work-holder domain with *Abstraction and Decomposition hierarchies.*

An important difference between these two hierarchies is that abstraction hierarchies do not change if the product family and class of work-holders (manufacturing operation) is not changed, whereas decomposition hierarchies vary with the changes in the situation of in the purchase of the components from different suppliers (related to the management of inventories).

To represent the problem domain we try to avoid the use of multiple inheritances to eliminate the problems of ambiguity related to the meaning of the attribute in a specialized class since it inherits the same attribute from multiple parents.

The most critical factor to measure the similarity in the work-holder design is the locating method. If the locating methods of two work-holder designs are the same, there is a basis for comparing their similarity.

Supporting access to the similar components through different classification schemas raises a number of issues. The classification schemas are specialized, and therefore a new user may not be familiar with the schema or terms that are employed. To solve this problem the corresponding computer support is needed.

The main guidelines for representing the information content of the DB are the following:

- Small number of product families is recommended to use, in order to generate DS model with consistent number of components.
- Assignment to each class of components a few significant features and attributes.
- Adding capability to visualize the component 3D models, to have an immediate idea of its usefulness.
- The of flexible classification structures, to adapt to continuous changes of the situation in a company.

3. THE ESTIMATION OF SIMILARITY OF PRODUCTS AND COMPONENTS

The similarity (semantic closeness) between models is a measure of how closely they are related with the design problem specification. The similarity estimation is based on the used classification schemes.

Let $sim(p_i, p_j)$ is a similarity function that is normalized and captures the closeness between the pair of points of DS $p_i \; and \; p_j$. The function *sim* could be one of the well-known distance metrics (e.g. L_1, L_2) or it could even be *non-*

metric (e.g. distance/similarity function provided by a domain expert). We assume that *sim* assumes values between 0 and 1, with larger values indicating that the points are more similar.
Given a threshold θ between 0 (components are not similar) and 1 (identical component), a pair of points p_i, p_j are defined to be neighbours if the following holds:

$$sim(p_i, p_j) \geq \theta.$$

In the above equation, θ is a user-defined parameter that can be used to control how similar a pair of points of DS must be in order to be considered neighbours.
The hierarchical nature of abstraction and decomposition hierarchies implies that closeness (similarity) is transitive: that is if T_1 *and* T_2 are close, and T_2 *and* T_3 are close, then T_1 *and* T_3 must be also closed.
We need to find the way to measure the similarity of two designs along some numeric scale. A generic method of measuring similarity is based on measuring a distance as a length of the path in classification tree representing the traversal from the first classification term to second. Starting from the closeness value of 1, each traversal on the path of classification tree diminishes the closeness by supplied weighting. For simplified example, with a diminishing factor 0,3 three traversal would represent the extent of similarity after which closeness would diminish to 0 (Fig. 3) (similarity value of less than zero are taken as 0).

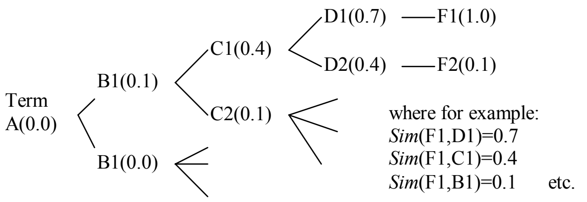


Figure 3. The extent of semantic closeness for the term F1.

In the real world, random structural alterations of components occur that measurement of similarity could have some random noise. This leads to the concept where we seek matches, which are not necessarily perfect, and are only good enough. Beside fixed value a *method of weighed similarity coefficients* determined by the inductive learning form previous examples is proposed.
The similarity between the two designs is estimated by the sum of similarity measures of its functional groups (or groups of decomposition hierarchy of DS such as: workpiece classification scheme, operation classification scheme, decomposition hierarchy of functional groups of work-holder):

$$S_{ij} = \frac{\sum_{n=1}^{N2} \left\{ S_{ij_n}, W_{Pn} \right\}}{\sum_{i=1}^{N2} W_{Pn}} + \frac{\sum_{n=1}^{N3} \left\{ S_{ij_n}, W_{On} \right\}}{\sum_{i=1}^{N3} W_{On}} + \frac{\sum_{n=1}^{N1} \left\{ S_{ij_n}, W_{Fn} \right\}}{\sum_{i=1}^{N1} W_{Fn}}$$

where:

i and j - are indexes of two designs to be compared;
n - is a functional feature index;
S_{ijn} - is a similarity measure between designs i and j at the functional group n;
W_{Pn} , W_{On} , W_{Fn} - is a weight factor assigned to the functional feature n according in workpiece classification scheme; operation classification scheme, workholder domain

The choice of attributes for feature description is one the most critical phases in component classification and retrieval system development, to develop it properly is necessary to follow the following criteria:

- To assign to each class of components the minimal number of attributes.
- Standardizing as much as possible the attribute names.
- Linking attributes and CAD parameters of 3D models.

For purposes of use the CBR the component's descriptions in DB must be enriched with additional attributes, such as the statistics of use, performance data, costing information, manufacturer or delivery data, etc. (Dieter, 2000).

4. DESIGN WORK-HOLDER USING CBR METHODOLOGY

A modular case-based design to support the hybrid generative/variant design, generating alternatives and evaluating them is considered as basic techniques for modeling of DS. A case-based design involves finding from database cases similar to the current problem and if necessary adapting them to match to the requirements.
The quality of design based on the case-base reasoning depends on the knowledge that is hold in DB, and the use of inductive learning abilities from previous successful solutions. The problems of support the case-base design and optimization of the list of feasible alternatives (cases in DB) of components for adapting a DS to the product family is described in (Küttner & Nekrassov, 2002).
By using the CBR approach in CAWD, the fixturing features of the given workpiece can be extracted, the similar workholder design can be identified, and the existing workholder design can be retrieved as is showed in Figure 4.
The analysis of design experience of similar work-holders and the development of recommendations are based on the identification of the similarity between the new design problem and the problems stored in the design database. The corresponding computer supported tools for the design feature extraction and similar solution retrieval must be developed and included into the system.
The workholder requirement to design must be identified through a key features of DS (CASE-BASE), where the workpiece, operational, and fixturing information is recognized. The workpiece, operational and work-holder information is represented by using indexing methodology and tree structure of the Design Space. In Figure 5. the tree ***structure workpiece domain*** DS with index and weight factor are presented.
The workholder-design similarity can be examined between an incoming work-holder design requirements and the one existing in CB work-holder models.
The key features of the given workpiece can be extracted, the similar work-holder design can be identified, and the existing work-holder design can be retrieved.
For example in Figure 4 shows a workpiece of *a lever fork*. The side surfaces of big hole on this workpiece are to be machined. The three overall dimensions are to be known. The locating mode (one plane and two holes) is used. Machine - a horizontal milling machine. There are three key features to be identified from ***workpiece domain*** DS: *shape types, material, dimension (see figure 5)*.
These key-features with index (*Type of work-piece, Material, etc.*) are presented in Table 1, which describe the information for design.
Step 1. Task specification and search the index of the *key features* from tree structure of the DS. The results are represented in table 1.
The values of the weight factors of the *key features* are assigned according to its importance in the workholder design (on supervisory level by administrator of DS).

Step 2. Retrieve similar case in CASE-BASE (work-holder with defined key features and index). The results are represented in Table 2.

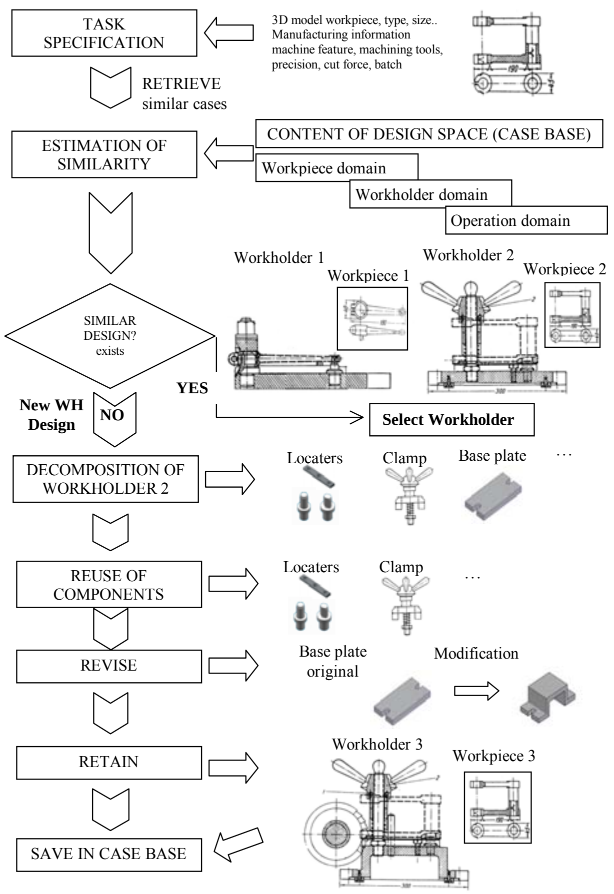


Figure 4. The framework of workholder designing process.

Table 1. The results of the first step (problem description).

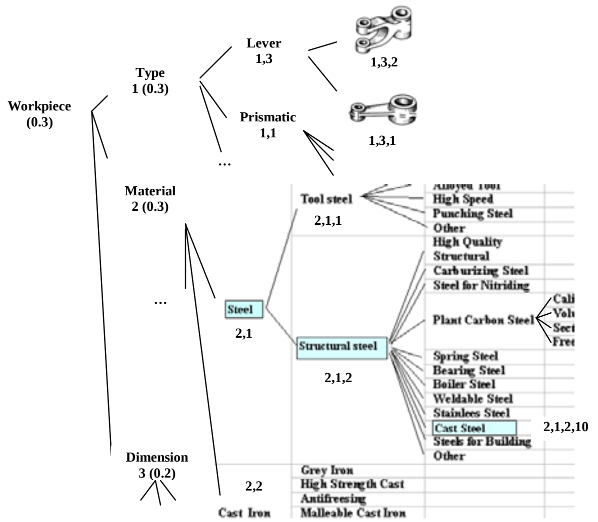
Key Feature	Value	Index from DS	Weigh factor	
Type of work-piece	Lever fork	1,3,2	0.3	*workpiece domain*
Material	steel	2,1,2,10	0.2	
Type operation	milling	1,1	0.5	*operation domain*
Quality	medium	2,2	0.3	
Batch	small-batch	3,1	0.2	
Location mode	one plane, two pin locating	1,2	0.3	*WH domain*
Clamping mode	manual, screw	2,1,2	0.3	
Type of base plate	rectangular	4,1	0.1	

Table 2. Results search similar workholders from CB

Key Feature	Value	problem description	WH 1	WH 2
Type of work-piece	lever fork	1,3,2	1,3,1	1,3,2
Material	steel	2,1,2,10	2,1,2,10	2,1,2,10
Type operation	milling	1,1	1,1	1,1
Quality	medium	2,2	2,2	2,2
Batch	small-batch	3,1	3,1	3,1
Location mode	one plane, two pin	1,2	1,2	1,2
Clamping mode	manual, screw	2,1,2	2,1,2	2,1,2
Type of base plate	rectangular	4,1	4,1	4,1

The ***workpiece*** similarity coefficients can be calculated between the workpieces in two examples described in Figure 4 and Table 2.

$$S_{ij} = \frac{\sum_{n=1}^{N2} \left\{ S_{ij_n}, W_{Pn} \right\}}{\sum_{i=1}^{N2} W_{Pn}} = \frac{1,0*0,3+1,0*0,3+0,7*0,2}{0,3+0,3+0,2} = 0,925$$



$$S_{ij} = \frac{\sum_{n=1}^{N2} \left\{ S_{ij_n}, W_{Pn} \right\}}{\sum_{i=1}^{N2} W_{Pn}} = \frac{0,7*0,3+1,0*0,3+1,0*0,2}{0,3+0,3+0,2} = 0,888$$

Figure 5. The structure workpiece domain DS.

The other ***operational*** similarity feature contains four factors: *machine type, operation type, quality, batch.* Same way the ***operational, workholder*** and common similarity coefficients can be calculated between the two examples described in fig. 4 and table 2 according function 1.

Finally, the similarity coefficient between the two workholder designs for two variants will be sorted by the value of similarity and the case with higher similarity will be got.

Otherwise (if in DB exist no suitable WH model), the problem description is decomposed according to the abstraction hierarchies and the models of suitable components are retrieved. In some cases new model must be designed using the capabilities of basic CAD system.

5. CONCLUSION

This paper has described our approach to use of case-based reasoning in CAD based on assessing the similarity of CAD DS models. The approach is intended to be general in the sense that the same basic ideas and could be used in different application domains. However, the information represented for design, as well as the criteria for judging the similarity of design is heavily domain-specific. As an example of a particular application domain, we have focused on work-holder design.

The work reported has been supported by the Estonian Science Foundation.

6. REFERENCES

Aamodt A. & Plaza E. (1994) Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Com-Artificial Intelligence Communications, 7(1), pp. 39–59, March 1994

Dieter, G. (2000). *Engineering Design,* McGraw-Hill International Editions, Mechanical Engineering Series, ISBN: 0-07-366136-8, USA

Küttner, R. & Nekrassov, G. (2002) Development of an intelligent integrated environment for computer-aided design of work-holders. *Proceedings NordDesign 2002*, Per Boelskifte & Johannes B. Sigurjonsson (Ed.), 185-193 pp, ISBN 82-91917-13-2, August 14-16 2002, NTNU, Trondheim, Norway

Kuttner,R.; Nekrassov,G.; & Sutt, A. (2003) Design Space for Collaborative Concurrent CAD of Technological Equipment. DAAAM International Scientific Book 2003, ed. Branko Katalinic, pp. 347-358. Vienna 2003

Mitrofanov S. P. (1966). The Scientific Principles of Group Technology. National Lending Library, Translation 1966

Nee, J. (1998). *Fundamentals of Tool Design,* Society of Manufacturing Engineers, ISBN: 0-87263-490-6, Dearborn, USA

Rong, Y. & Zhu, Y. (1999). *Computer-Aided Fixture Design,* Marcel Dekker, Inc. ISBN: 0-8247-9961-5, USA