Application and development

Of case based reasoning in fixture design

Abstract: Based on the case based designing (CBD) methodology, the fixture similarity is in two respects: the function and the structure information. Then, the computer aided fixture design system is created on case based reasoning (CBR),in which the attributes of the main features of workpiece and structure of fixture as case index code are designed for the retrieve of the similar cases, and the structure and hierarchical relation of case library are set up for store. Meanwhile, the algorithm based on the knowledge guided in the retrieve of the similar cases, the strategy of case adapt at ion and case storage in which the case ident if cat ion number is used to distinguish from similar cases are presented. The application of the system in some projects improves the design efficiency and gets a good result .

Keywords: case based reasoning ;fixture design; computer aided design(CAD)

Fixtures are devices that serve as the purpose of holding the workpiece securely and accurately, and maintaining a consistent relationship with respect to the tools while machining. Because the fixture structure depends on the feature of the product and the status of the process planning in the enterprise, its design is the bottleneck during manufacturing, which restrains to improve the efficiency and leadtime. And fixture design is a complicated process, based on experience that needs comprehensive qualitative knowledge about a number of design issues including workpiece configuration, manufacturing processes involved, and machining environment. This is also a very time consuming work when using traditional CAD tools (such as Unigraphics, CATIA or Pro/E), which are good at performing detailed design tasks, but provide few benefits for taking advantage of the previous design experience and resources, which are precisely the key factors in improving the efficiency. The methodology of case based reasoning (CBR) adapts the solution of a previously solved case to build a solution for a new problem with the following four steps: retrieve, reuse, revise, and retain [1]. This is a more useful method than the use of an expert system to simulate human thought because proposing a similar case and applying a few modifications seems to be self explanatory and more intuitive to humans .So various case based design support tools have been developed for numerous areas[2-4], such as in injection molding and design, architectural design, die casting die design, process planning, and also in fixture design. Sun used six digitals to compose the index code that included workpiece shape, machine portion, bushing, the 1st locating device, the 2nd locating device and clamping device[5]. But the system cannot be used for other fixture types except for drill fixtures, and cannot solve the problem of storage of the same index code that needs to be retained, which is very important in CBR[6].

1 Construction of a Case Index and

Case Library

1.1 Case index

The case index should be composed of all features of the workpiece, which are distinguished from different fixtures. Using all of them would make the operation in convenient. Because the forms of the parts are diverse, and the technology requirements of manufacture in the enterprise also develop continuously, lots of features used as the case index will make the search rate slow, and the main feature unimportant, for the reason that the relative weight which is allotted to every feature must diminish. And on the other hand, it is hard to include all the features in the case index.

Therefore, considering the practicality and the demand of rapid design, the case index includes both the major feature of the workpiece and the structure of fixture. The case index code is made up of 16 digits: 13 digits for case features and 3 digits for case identification number.

The first 13 digits represent 13 features. Each digit is corresponding to an attribute of the feature, which may be one of“*”, “? ”, “1”, “2”,…, “A”,“B”,…, “Z”,…, etc. In which, “*” means anyone, “?” uncertain, “0” nothing.

The system rules: fixture type, workpiece shape, locating model cannot be “*” or“?” . When the system is designed, the attribute information of the three items does not have these options, which means the certain attribute must be selected.

The last three digits are the case identification number, which means the 13 digits of the case feature are the same, and the number of these three digits is used for distinguishing them.

The system also rules: “000” is a prototype case, which is used for retrieval, and other cases are “001”, “002”,…, which are used for reference cases to be searched by designers. If occasionally one of them needs to be changed as the prototype case, first it must be required to apply to change the one to “000”, and the former is changed to referential case automatically.

The construction of the case index code is shown in Fig.1.

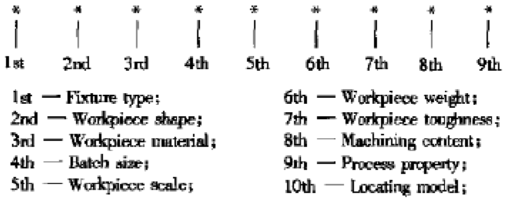

Fig. 1 Construction of case

1.2 Case library

The case library consists of lots of predefined cases. Case representation is one of the most important issues in case based reasoning. So compounding with the index code,.

1.3 Hierarchical form of Case

The structure similarity of the fixture is represented as the whole fixture similarity, components similarity and component similarity. So the whole fixture case library, components case library, component case library of fixture are formed correspondingly. Usually design information of the whole fixture is composed of workpiece information and workpiece procedure information, which represent the fixture satisfying the specifically designing function demand. The whole fixture case is made up of function components, which are described by the function components’ names and numbers. The components case represents the members. (function component and other structure components， main driven parameter, the number, and their constrain relations.) The component case (the lowest layer of the fixture) is the structure of function component and other components. In the modern fixture design there are lots of parametric standard parts and common non standard parts. So the component case library should record the specification parameter and the way in which it keeps them.

2 Strategy of Case Retrieval

In the case based design of fixtures ,the most important thing is the retrieval of the similarity, which can help to obtain the most similar case, and to cut down the time of adaptation. According to the requirement of fixture design, the strategy of case retrieval combines the way of the nearest neighbor and knowledge guided. That is, first search on depth, then on breadth; the knowledge guided strategy means to search on the knowledge rule from root to the object, which is firstly searched by the fixture type, then by the shape of the workpiece, thirdly by the locating method. For example, if the case index code includes the milling fixture of fixture type, the search is just for all milling fixtures, then for box of workpiece shape, the third for 1plane+ 2pine of locating method. If there is no match of it, then the search stops on depth, and returns to the upper layer, and retrieves all the relative cases on breadth.

Retrieval algorithms:

1)According to the case index information of fixture case library, search the relevant case library;

2)Match the case index code with the code of each case of the case library, and calculate the value of the similarity measure;

3)Sort the order of similarity measure, the biggest value, which is the most analogical case.

Similarity between two cases is based on the similarity between the two cases. features. The calculation of similarity measure depends on the type of the feature. The value of similarity can be calculated for numerical values, for example, compareWorkpiece with the weight of 50kg and 20kg. The value can also be calculated between non numerical values, for example, now the first 13 digits index code is all non numerical values. The similarity measure of a fixture is calculated as follows:

$$S = \frac{\sum_{i=1}^{n}(w_i s_i(fa_i, fb_{ji}))}{\sum_{i=1}^{n} w_i} \qquad i = 1, 2,$$

where S is the similarity measure of current fixture, n is the number of the index feature, w_i is the weight of each feature, $s_i(fa_i, fb_{ji})$ is the similarity measure of the attribute fa_i of the i2th feature with the attribute fb_{ji} of relative feature of the j-th case in the case library. At the same time, $0 \leqslant s_i(fa_i, fb_{ji}) \leqslant 1$, the value counts as follows:

```
If ($fa_i$ and $fb_{ji}$ are non-numerical values)
If ($fa_i = fb_{ji}$)
$s_i(fa_i, fb_{ji}) = 1$;
Else
$s_i(fa_i, fb_{ji}) = 0$;
If ($fa_i$ and $fb_{ji}$ are numerical values)
$s_i(fa_i, fb_{ji}) = 1 - \left| \frac{fa_i - fb_{ji}}{\max(fa_i, fb_{ji})} \right|$.
```

.

Where fa_i is the value of the index attribute of the i-th feature, and fb_{ji} is the value of attribute of the relative i-th feature of the j-th case in case library.

So there are two methods to select the analogical fixture. One is to set the value. If the values of similarity measure of current cases were less than a given value, those cases would not be selected as analogical cases. When the case library is initially set up, and there are only a few cases, the value can be set smaller. If there are lots of analogical cases, the value should get larger. The other is just to set the number of the analogical cases (such as10), which is the largest value of similarity measure from the sorted order.

3 Case adaptation and Case Storage

3.1 Case adaptation

The modification of the analogical case in the fixture design includes the following three cases:

1) The substitution of components and the component;

2) Adjusting the dimension of components and the component while the form remains;

3) The redesign of the model.

If the components and component of the fixture are common objects, they can be edited, substituted and deleted with tools, which have been designed.

3.2 Case storage

Before saving a new fixture case in the case library, the designer must consider whether the saving is valuable. If the case does not increase the knowledge of the system, it is not necessary to store it in the case library. If it is valuable, then the designer must analyze it before saving it to see whether the case is stored as a prototype case or as reference case. A prototype case is a representation that can describe the main features of a case family. A case family consists of those cases whose index codes have the same first 13 digits and different last three digits in the case library. The last three digits of a prototype case are always “000”. A reference case belongs to the same family as the prototype case and is distinguished by the different last three digits.

From the concept that has been explained, the following strategies are adopted:

1) If a new case matches any existing case family, it has the same first 13 digits as an existing prototype case, so the case is not saved because it is represented well by the prototype case. Or is just saved as a reference case (the last 3 digits are not “000”, and not the same with others) in the case library.

2) If a new case matches any existing case family and is thought to be better at representing this case family than the previous prototype case, then the prototype case is substituted by this new case, and the previous prototype case is saved as a reference case.

3) If a new case does not match any existing case family, a new case family will be generated automatically and the case is stored as the prototype case in the case library.

4 Process of CBR in Fixture Design

According to the characteristics of fixture design, the basic information of the fixture design such as the name of fixture, part, product and the designer, etc. must be input first. Then the fixture file is set up automatically, in which all components of the fixture are put together. Then the model of the workpiece is input or designed. The detailed information about the workpiece is input, the case index code is set up, and then the CBR begins to search the analogical cases, relying on the similarity measure, and the most analogical case is selected out. If needed, the case is adapted to satisfy the current design, and restored into the case library. The flowchart of the process is shown in Fig.3.

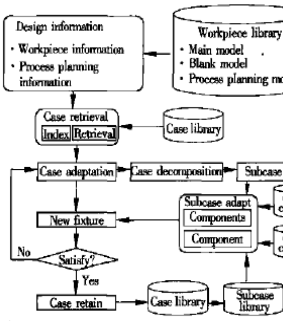

Fig. 3 Flowchart of case-based reasoning for f

5 Illustrating for Fixture Design by CBR

This is a workpiece (seeFig.4). Its material is 45# steel. Its name is seat. Its shape is block, and the product batch size is middle, etc. A fixture is turning fixture that serves to turn the hole, which needs to be designed.

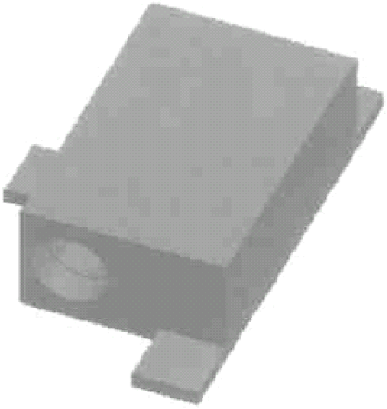

Fig. 4 A workpiece requested to design a fixture (Maximum size is 80 mm × 49 mm × 22 mm)

The value of feature, attribute, case index code and weight of the workpiece is show n in Tab.2.

Tab. 2 The case index code

Feature name	Attribut
Fixture type	Lathe
Workpiece shape	Blo
Workpiece material	Middle ca
Batch size	Mi
Workpiece scale	Sm
Workpiece weight	Lig
Workpiece toughness	Much
Machining content	Ho
Process property	Finish m
Locating method	3pl
Clamping method	Unce
Fixture body	Compo
Others	Notl

Through searching, and calculating the similarity, the case index code of the most similar case is 19325513321402000, and the detailed information is show n in Tab. 3.

Tab. 3 Case index code of

Feature name	At
Fixture type	I
Workpiece shape	
Workpiece material	Mid
Batch size	
Workpiece scale	
Workpiece weight	
Workpiece toughness	M
Machining content	
Process property	Fir
Locating method	
Clamping method	
Fixture body	C
Others	

The similarity is calculated as follows:

$$S = \frac{1 \times 100+ 1 \times 90+ 2 \times 1 \times 70+ 5 \times 1 \times 60+ 2 \times}{100+ 90+ 70+ 60+ 60+ 60+ 60+ 80+ 70} \frac{790}{980} = 0.806$$

So the value of similarity measure of the fixture which needs to be designed with the most analogical case in case library is 0.806, and the structure of the most analogical case is shown in Fig.5.

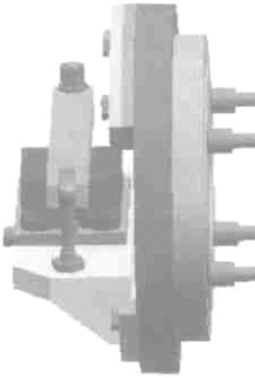

Fig. 5 Most analogical fixt

After having been substituted the component, modified the locating model and clamp model, and adjusted the relative dimension, the new fixture is designed, and the figure is show n in

Fig.6.

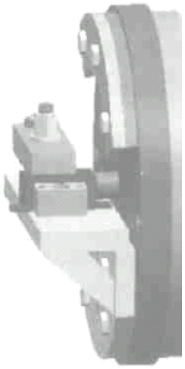

Fig. 6 New fixture that nee

As there is not the analogical fixture in the case library, the new fixture is restored in to the case library. The case index code is 19325513311402000.

6 Conclusion

CBR, as a problem solving methodology, is a more efficient method than an expert system to simulate human thought, and has been developed in many domains where knowledge is difficult to acquire. The advantages of the CBR are as follows: it resembles human thought more closely; the building of a case library which has self learning ability by saving new cases is easier and faster than the building of a rule library; and it supports a better transfer and explanation of new knowledge that is more different than the rule library. A proposed fixture design framework on the CBR has been implemented by using Visual C ++, UG/Open API in U n graphics with Oracle as database support, which also has been integrated with the 32D parametric common component library, common components library and typical fixture library. The prototype system, developed here, is used for the aviation project, and aids the fixture designers to improve the design efficiency and reuse previous design resources.