Functional-to-form mapping for assembly design automation

Z G Xu[1], W M Liu[1], W D Shen[2], D Y Yang[2], T T Liu[2]

[1]Shenzhen Research Institue of Shandong University, 518057, High-tech Zone A301, Virtual University Part, Shenzhen, China.
[2]School of Mechanical Eng, Shandong University, 250061, No 17923, Jingshilu, Lixiaqu, Jinan, Shandong, China

Abstract. Assembly-level function-to-form mapping is the most effective procedure towards design automation. The research work mainly includes: the assembly-level function definitions, product network model and the two-step mapping mechanisms. The function-to-form mapping is divided into two steps, i.e. mapping of function-to-behavior, called the first-step mapping, and the second-step mapping, i.e. mapping of behavior-to-structure. After the first step mapping, the three dimensional transmission chain (or 3D sketch) is studied, and the feasible design computing tools are developed. The mapping procedure is relatively easy to be implemented interactively, but, it is quite difficult to finish it automatically. So manual, semi-automatic, automatic and interactive modification of the mapping model are studied. A mechanical hand F-F mapping process is illustrated to verify the design methodologies.

1. Introduction

So far, many design automation tools are embedded in present CAD systems, the model creation, modification, maintenance and propagation become more convenient with the introduction of the latest 2D and 3D parametric design technologies [1] (Hoffmann, 2001). ICAD systems are becoming even more powerful, when PDM (product data model), or ERP (enterprise resource planning) and MRP II (materials resource process planning) database as well as advanced HCI (human computer interface), VR (virtual reality), and web-based technologies etc are rapidly incorporated. However, it may be pointed out that it only represents a vision which is almost identical to the earliest visions of CAD systems in view of assembly-level design automation. The major difference now is that we have the opportunity through our improved knowledge of AI and cognitive science to take important strides towards delivering CAD systems closer to these visions.

Some warned that the opportunity may be missed if we concentrate on the wrong issues [2]. Many others may suggest the appropriate approaches [3]. In this paper, several assembly-level "enhancement procedures" are testified.

A specific function-to-form mapping model (F-F mapping model for short) is proposed to cope with divergent exploration, automatic transformation and convergent exploitation. Decomposition of design domains in computational design context helps to set up the domain specific knowledge base (KB).

Automatic mapping among design domains could dramatically increase exploration proficiency. And reconstitution algorithms evolving default logic is to cope with ambiguous information occurred in conceptual design stage, which is also effective exploration tool for concept creation and visualization, called exploitation. The F-F mapping model cooperating axiomatic design theory [4] is effective to integrate requirement, function and form domains for a creative design tool.

2. Mathematical model for assembly stage mapping

Let U be a functional domain (Function Requirements, referred to as **FRs**), V is the domain (Design Properties, referred to as **DPs**), $P(U)$, $P(V)$ is the power set of U, V, $FRs \subset P(U)$, $DPs \subset P(V)$. $f_{FRs \to DPs}$, where, **FRs** is called the definition domain,

Definition 1, functional carrier, if there exist transitional domains, between functional domain and form domain, called **CRs**, make $x \subset CRs$, if and only when x satisfies the followed two terms:

- A simple, intuitive, deterministic correspondence that satisfies **FRs** to **CRs**
- At the same time, mapping from **CRs** to **DPs** satisfies the 'deterministic' criterion, where 'deterministic' means configurability.

Non-manifold entities are the "carrier" from the function-to-form mapping. For example, in a gearbox, the shifting fork is to "shift the direction of the two gears", the Non-manifold entities of the shifting fork is composed of three functional faces, as in figure 1.
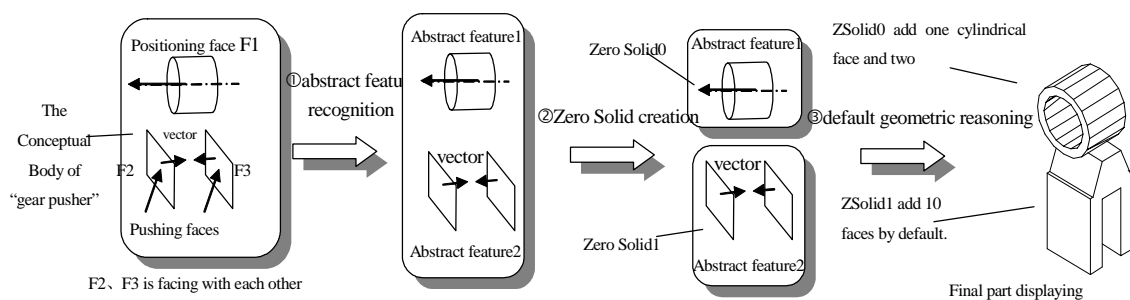

Figure 1. Function-to-form mapping of the shifting fork.

3. Definition of assembly level function

Functional domain is classified into three groups, quantitative, abstract, and assembly. We believe that functional definition could be correlated to the problems concerned; product-level, assembly-level or component-level functions are to be taken into considerations.

As long as the assembly structural design is concerned, the assembly level function should be summarized, classified and analyzed. As in figure 2, five levels of functional definitions are arranged in parallel rows, where the assembly-level function is summarized as a connection link model, each node of which is an information unit to express the prerequisite assembly requirement, like positioning, transmission, supporting and lubrication etc in a network structure.

Table 1. Hierarchy of FRs.

Hierarchical function definitions	Function contents	Source of function requirements
Product level functions	Product design requirements	Market analysis, user requirements, etc.
Mechanism level sub function	The performance requirements of the product, mainly refers to the relative translation or rotation between parts	Behavioral performance requirements of products produced by demand analysis
Component assembly stage sub function	Functions such as movement, transmission, blocking, positioning, clamping, sealing, lubrication, connecting and fastening between parts	Obtained by the construction principle of the mechanism

The information unit could be correlated with many alternatives of functional carriers, what-ever it might be, a conceptual face, functional feature, component or mechanism etc. The hierarchical definition of the overall functional requirements is in table 1.

Prior to the first phase of design, there exists a requirement management and mapping process. Requirement domain is dealing with product specifications, user's intent, market information, product life cycle analysis/assessment (LCA) inventory, as well as information about the Production Company etc.

It would provide an information bulk necessary to start the development of a new product. Requirement domain is associated six research areas.

1) capturing of customer's need effectively [5]; 2) creation of product specifications elicited from customers need [6]; 3) computer-based representation of product specifications; 4) testing of product performance against the product specification. 5) design intent management with LCA inventories, where design intent management and the interface for acquisition, representation, analyzing, recording and retrieving of the user's intent information from the product life cycle design perspectives are outlined.
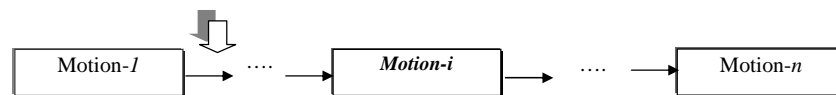

Figure 2. Function definitions.

After the Design requirement is captured, recorded and analyzed. The next step is to propagate the factors to the downstream design activities.

4. Generalized positioning for mechanism

Generalized positioning: Traditionally, a part is thought to be totally positioned by six-point, i.e. translations in X, Y, Z, and rotations in A (around X), B (around Y), C (around Z). However the concept of twelve-point positioning is introduced for better understanding of part dynamics as well as static positioning. One part is thought to be totally positioned by 12 degrees of freedom, i.e. translations in X, -X, Y, -Y, Z, -Z, and rotations in A, -A, B, -B, C, -C,

is called generalized positioning, and the degrees of freedom is called generalized DOF (GDOF for short).

The reason of the proposal of 12 GDOF is that it is easier to set up the mapping correlation of the positioning requirement i.e., the positioning requirement of the GDOF number of the component with respect to a certain feature.

For example, the following slot feature could be thought of being restricted in the user coordinate system (UCS) of the –Y, +X and –X , +A and –A, +B and –B directions, and the other GDOFs are free, which will be encoded in a string of hex code like in table 2.
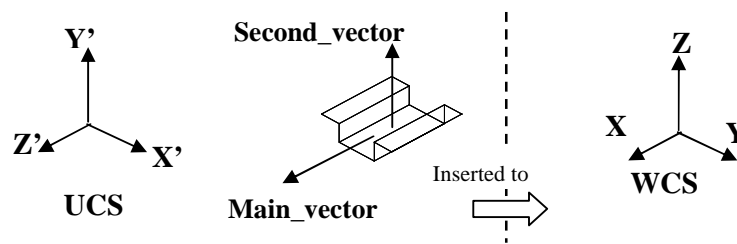

Figure 3. Slot feature in UCS with respect to WCS.

Table 2 gives the GDOF representation of the slot feature (figure 3). The GDOF code in UCS is F73, while the GDOF code in world coordinate system (WCS) should be modified according to the inserting orientation of the UCS with respect to WCS, at this insertion point, the GDOF code in WCS should be 3F7. Table 2 is the GDOF representation of slot feature with respect to the UCS and WCS.

Table 2. The GDOF representation of slot feature with respect to the UCS and WCS.

WCS	Y		B		Z		C		X		A	
UCS	X'		A'		Y'		B'		Z'		C'	
	+	-	+	-	+	-	+	-	+	-	+	-
Binary	1	1	1	1	0	1	1	1	0	0	1	1
Hex	F				7				3			

✴ Hex coding is accepted by GPAL_KN due to its simple and explicit format.

5. Product assembly model based on functional features

After function-to-form mapping (**F-F mapping** for short), the product conceptual model has been established, the conceptual product model is composed of functional faces and their correlations. There are three levels in a assembly model of one product.

According to the functional requirement of support, stop, fixing or positioning etc, different features are mapped to the corresponding parts, so one specific feature is mapped to two or more corresponding parts, and the relationship between parts in a product are naturally established, as in figure 4.

The Sub-functions in figure 4 mainly refers to the functions of drive, location, clamping, sealing, blocking, etc. for motion function motioni, after **F-F** mapping, a meta_element is formed, defined as meta_elementArray:
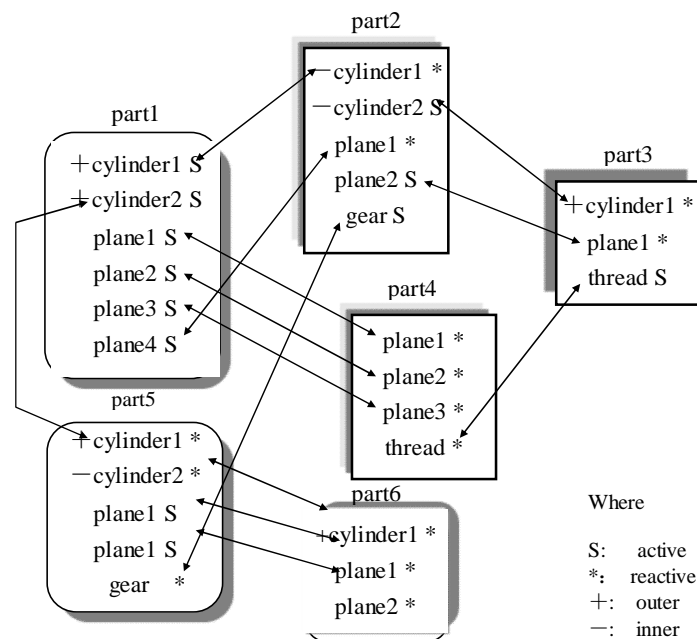
Figure 4. Product assembly model.

```
class meta_element
{  ……
    enum Function_Type           //  assembly level functional definition
    enum element_type;           // include:  FuncFace,  FuncFaceArray
                                 //  and standard_part .
    MfuncFace * facept;          //  functional faces
    AcDbVoidPtrArray face_array; //   functional faces sets
    Mpart * standard_part        //   pointer to a standard part
     ………
}
AcDbVoidPtrArray meta_elementArray  //  meta_element.
```
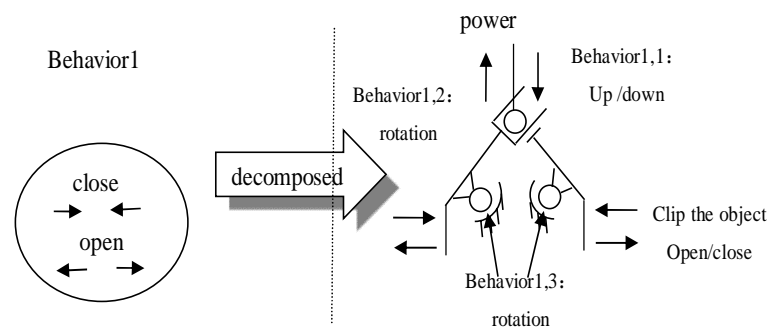

Figure 5. Functional requirement of the mechanical hand.

Based on the product assembly model, the features are expressed in the assembly, and the relationship between features and features, parts and features, and parts and parts etc, with the information of location, size, constraints etc, are defined in a data structure to describe the three types of relations, as in figure 4.

6. Case study

To clamp one work-piece, the mechanical hand have the opening and closing movement, the Behavior1 can be furtherly decomposed into a transmission principle diagram as shown in figure 5.

Stepping motors is firstly selected interactively. The stepper motor is related to several parts as, drive parts, nuts, screw drive components. In combination with figure 5, the 3-D transmission chain of the mechanical hand is deduced, as in figure 6, and the corresponding parts of the product are created at the same time.
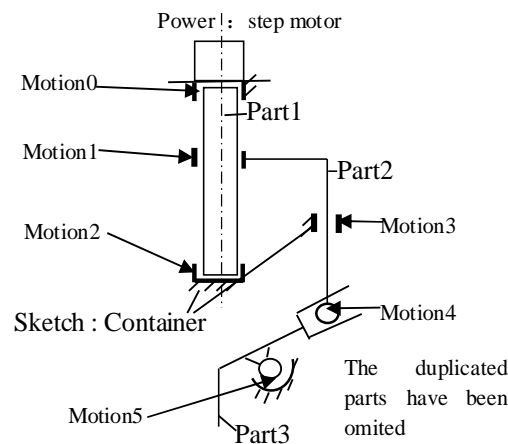
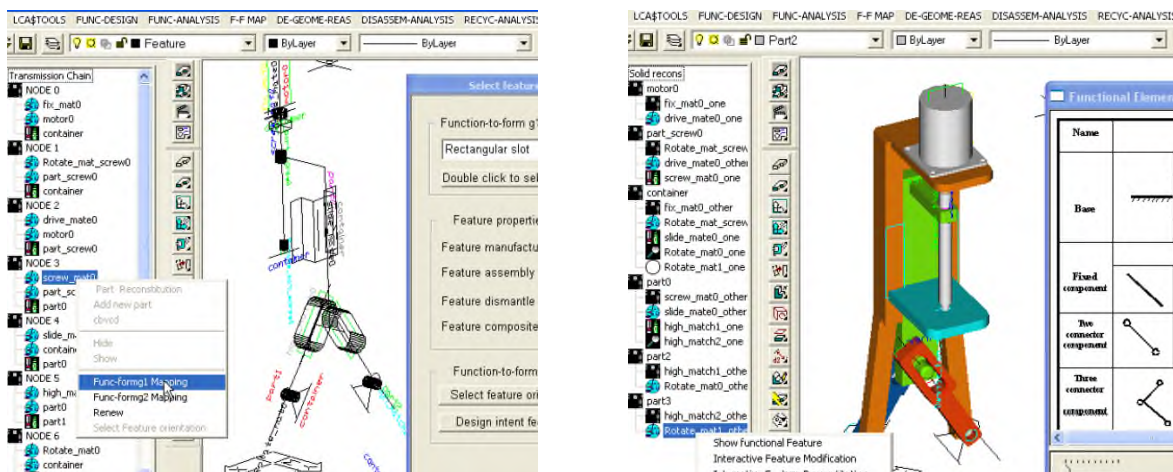**Figure 6.** 2-D Functional mode of the mechanical hand.

As shown in figure 6, from the driving principle of the mechanical hand, four correlated parts created, namely, Part1, Part2, Part3 and Container.

Table 3. The relationship between the created parts and the resulting meta bodies.

function	meta_element	Parts involved
Motion0	F01(Cylinder),F02 (plane)	Part1, Container
Motion1	F11(thread)	Part1, part2
Motion2	F21(Cylinder),F22(plane)	Part1, container
Motion3	F31,F32 F33,F34(plane)	Part2, container
Motion4	F41(double),F42、 F43(Plane)	Part2(F411,F412(plane)) part3(F413(cylinder))
Motion5	F51(cylinder),F52、 F53(plane)	Part3, container

Table 3 illustrates the five motion in the mechanical hand, meta_elements are firstly deducted. For example, the meta_element F41 is deduced by the motion4, F41 is mapped on Part2, and the corresponding feature, F411 and F412 are created, in the mean while F412 is created on Part3.

And the relationship between the two parts, i.e. Part2, Part3, and the corresponding features F411, F412 and F413 are established. The functional surface sets of Part1, part2, and part3 formed by the function structure mapping are shown in figure6. And the mapping process and the mapping results are shown in figure 7.

(a) 3-D transmission chain of the mechanical hand   (b) 3-D Model of the mechanical hand

Figure 7. F-F mapping of the mechanical hand and the result.

7. Conclusion

An effective assembly level function-to-form mapping model is proposed in this paper. It could be concluded that, the **decomposition**, F-F **mapping** and **reconstitution** model could be a common exploration, transformation and exploitation procedure for management of computational design tools, therefore helpful for creative work, specifically the synthesizing of form domain is not simply re-arrangement of physical elements, rather it needs geometrical as well as algebraic reasoning on un-manifold polyhedral. Novel mathematical as well as AI technologies were introduced. Several packages of design tools have been developed to testify the effectiveness of the design methodology.

8. References

[1]   Hoffmann C M and Kim K J 2001 *Comput. Aid. Des.* **33(1)** 81-90.
[2]   MacCallum K J 1990 *Artific. Intell. Eng.* **5(2)** 55-64.
[3]   Faltings B and Sun K 1996 *Comput. Aid. Des.* **28(3)** 207-16.
[4]   Suh N P 1990 *The principles of Design*. Oxford University Express, New York.
[5]   Stauffer L and Morris L 1992 *A new program to enhance the development of product requirements*. Presented at NSF Design and Manufacturing Systems Conference, Atlanta, Georgia,
[6]   Grothe M T 1998 *Visual modeling of products and life-cycle activities*. Presented at DETC'98 ASME Design Engineering Technical Conference, Atlanta, Georgia,

Acknowledgment