AN AUTOMATED FUNCTION DECOMPOSITION METHOD BASED ON A FORMAL REPRESENTATION OF SOLID MATERIAL'S SHAPE

Yuan, Lin (1); Zhang, Zhinan (2); Liu, Yusheng (1)
1: Zhejiang University, People's Republic of China; 2: Shanghai Jiao Tong University, People's Republic of China

Abstract

The objective of this work is to develop a formal representation of the material flow's shape in the conceptual design stage, specifically to meet the needs of function modelling in order to support the function decomposition of working machines. A hierarchical shape graph (HSG) is used to abstract the structure of shape elements for a material flow's shape. Based on the semantic information contained in the input/output material flow's HSG model, the global shape change of a material flow is decomposed to several local relation changes between shape elements. Moreover, a planning algorithm is proposed to support finding a feasible sub-function execution sequence for achieving the shape change function. Finally, a knowledge representation scheme is developed for action, which is the principle solution to bridge the result of system level function decomposition and the design of mechanism motions. The representation and automated decomposition process have been implemented based on IBM Rhapsody platform.

Keywords: Conceptual design, Functional modelling, Function decomposition, SysML

Contact:
Lin Yuan
Zhejiang University
Computer Science
China, People's Republic of
yuanlin@zjucadcg.cn

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 20th International Conference on Engineering Design (ICED15), Vol. nn: Title of Volume, Milan, Italy, 27.-30.07.2015

1 INTRODUCTION

The purpose of conceptual design is to construct the function structure and find the working principle for each sub-function in the function structure (Pahl and Beitz, 1988). The function of mechanical product can be considered as a process which exchanges, transmits, and stores the input flows (material, signal, energy) to make them output with desired state. The core function of a large number of mechanical systems existing in industry and daily life is to process material flows. These kind of product is called working machine. Moreover, the shape and structure change of solid material is the most common effect that a working machine puts on a material flow. However, the representation of material's shape information in conceptual design phase is difficult. In the existing function representation methods, the input/output flow's attributes are just described in textual format without detailed semantic information. Thus the existing function reasoning methods always deal with energy flow which is easier to be described. How to describe material flow's shape in conceptual design to support the function decomposition for the working machines is still an open problem.

To improve the applicability of function modelling and enhance the automation of function decomposition, we focused on the description of material flow's shape structure and the function decomposition for material's shape change. A function representation method is proposed, which supports describing the input and output material flow's shape information in the early stage of conceptual design based on SysML. The global shape change of a material can be decomposed to several local change effects by reasoning process. Finally, the working principle to implement each of the local change effect will be found. Thus, there are three tasks in this work:

- A computer readable and computable material flow shape representation model.
- An automated reasoning method for decomposing the global shape change to local change effect.
- A working principle retrieval method to find principle solution for implementing shape change effect.

2 RELATED WORK

As conceptual design has been paid more and more attention from academia and industry, there are amount of researches on function representation and function decomposition existed in the last decades. The functional basis (Stone and Wood, 2000) provided standard vocabularies for function description and the classification of functions behaviour (verbs) and objects (nouns). However, this representation is merely based on plain text without semantic content, thus unable to provide information for function decomposition. Sen et al. (2011, 2013) proposed a function verb redefinition protocol to extend the function basis and support conservation based consistencies check. Although this method is novel in enhancing the semantics of function representation, but it mainly focused on the mass conservation of material flow and ignored the important shape information. To support the describing of material flow's structure and shape feature in the early phase of conceptual design, the hierarchical material flow based function representation was proposed (Yuan and Liu, 2014). But the method is more useful to visualization the material flow semantic information, the computability and readability is weak. Li et al. (2014) gave a geometric reasoning approach to divide B-rep model into hierarchical model based on adjacent graph for model retrieval. In the research of motion planning for operating robotic fingers to manipulate complex cartons folding process in packaging industry, Liu and Dai (2003) used graph as the abstract model to describe cartons' shape structure.

Researchers also focused on the reusing of design knowledge to facilitate design process. Sun and Liu (2011) thought the role of a function is to lead the changes to flow's properties. He called the changes as functional effects and presented a function decomposition method to decompose an overall function to several sub-functions, each of them related to a functional effect which would correspond to the known physical effect as a specific working principle. Zhang et al. (2012) proposed an ontology framework to reuse knowledge flow in an effective and efficient way. Chen et al. (2011) consider that the essence of a function is the behaviour of object effect by subject's behaviour. The object and its behaviour are the flows and how they be changed respectively. The subject behaviour can be attributed to the mechanical systems mechanism motion. There are several excellent works for the conceptual design of mechanism motion. Such as Chiou and Kota (1999) devised a matrix representation scheme for mechanism's motion to enable automatic decomposition and identified 43 physical building blocks from hundreds of ingenious mechanisms as knowledge for implementing the decomposed functions.

Han and Lee (2006) proposed a case-study approach for synthesis mechanism motion with adaptation rules. However, there still has a gap between system-level function decomposition and mechanism motion design as they are not integrated in the whole conceptual design phase.

3 HIERARCHICAL SHAPE GRAPH

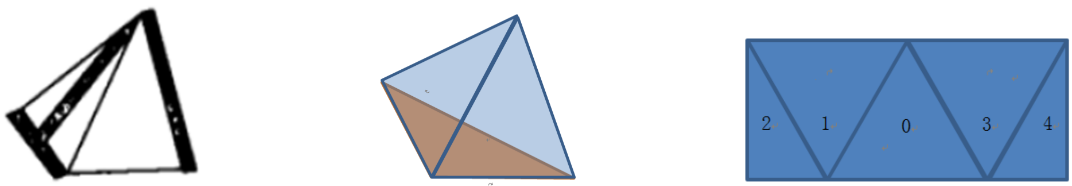
To aid designers to model the material flow's shape structure in the early stage of conceptual design, and make the model readable to a computer, we extend the hierarchical material flow based function representation (Yuan and Liu 2014). For a solid material flow, a *shape element* is used to represent its shape feature. Both *single element* and *composite element* are generalized from shape element. The former refers to a basic geometry element, such as a cube, a rectangle and a line. The latter is used to model the complex shape of material flow which cannot be simply abstract as a single element. Composite element can be subdivided into several sub shape elements. Then the hierarchical partition graph (Li et al. 2014) is adapted to model material flow's shape structural in different levels of abstraction (or resolution). This shape description model is named hierarchical shape graph (HSG), which includes the following two parts:

Hierarchical Structure Tree (HST) is an abstract model for the structure of the shape elements of a material flow. The root of a HST is a shape element which indicates the abstract shape of a material flow. Each child node is a local shape descriptor corresponding to a sub shape element. The child node can be a leaf node (a single element) or a non-leaf node (composite element). An adjacent graph is referenced by a composite element to describe how the children shape elements relate to each other.

Local Adjacent Graph (LAG) is the description of connection relationship between the shape elements owned by a composite element. Thus the local shape feature is modelled by an adjacent graph with the vertex refers to shape element and the edge refers to the connection relationship of a pair of shape elements. If the children nodes of a composite element are all single elements in the HST, the local adjacent graph is exactly the descriptor of the entire sub shape structure. If there are non-leaf nodes (composite element) existed in the children nodes, the local adjacent graph just shows some scattered local connections between the sub shape elements. For example, how a sub composite element connects to other shape elements by a connection relationship between its owned single element and other shape elements.

The *relation constraint* is used to represent the semantic information of a certain edge. A relation constraint is defined as a 4-tuple: <source, target, relation type, value>. Here, "source" and "target" are two single elements. relation type shows the geometric or space relation from source to target. The space relation can be "distance", "overlap", "on", "under", "cross" and so on. And the geometry relation can be "angle", "perpendicular", "parallelism", "symmetry". The "value" is a qualitative or quantitative value of the geometric or space relation.

To explain our model method more clearly, the shape of a pyramid box in Figure 1 is used as an example. Figure 1(a) is a cardboard box with the shape of pyramid as Figure 1(b) shows. And the shape of the cardboard before the folding process is a flat plane with several parts as Figure 1(c).



(a) A pyramid box *(b) The abstract shape of pyramid box* *(c) The cardboard*

Figure 1. An example of pyramid box

Figure 2 is the HSG model for the shape structure of the input cardboard. The HST of the shape elements are shown in Figure 2(a), there are 5 shape elements named from part0 to part4 integrate together to form the entire shape of this cardboard. Each of these shape elements is a single element. As they are linked in the same plane, the angle between each two of them is 180 degrees. The LAG of the root node is shown as Figure 2(b). Note that each edge corresponds to a relation constraint which contains the semantic information of the relationship of the two end vertices. Such as <part2, part1, angle, 180> to the edge of part2 and part1.

(a) HST of the shape of cardboard *(b) The LAG of the root node*

Figure 2. HSG model of the input cardboard

Figure 3 shows the HSG of the shape structure of the output pyramid box. The HST represents that the overall shape descriptor (the root) of this box has two shape elements in the first level of detailed: the single element part5 as the base and the pyramidal sides. In the second level of detailed, the pyramidal side consists of part1, part0 and part3. Then LAG of the root node and sides show how the shape elements connect to each other.

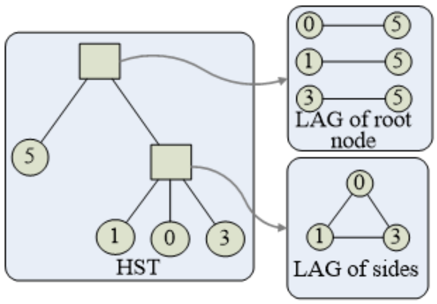


Figure 3. HSG model of the output pyramid box

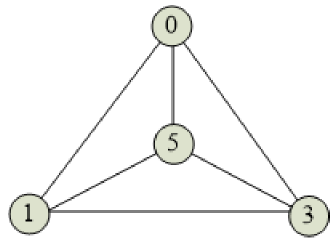


Figure 4. The global adjacent graph

HSG gives the abstract model of a material flow's shape by representing the hierarchical structure of shape elements and describing the local connection between them. Through parsing the HSG of a material's shape model, and extracting all the leaf nodes in the HST and all the edges in each LAG of this HSG, a global adjacent graph can be built. It is the overview of a material's shape which consists of the single shape elements in the HSG, And the detailed relations between these elements are also be represented. For the input cardboard, as its HSG only has leaf nodes besides the root, the global adjacent graph of it is the same as Figure 2(b). And for the output box, after retrieving its HST together with extracting information contained in LAGs, the corresponding global adjacent graph is shown in Figure 4. It shows that the four sides are connecting to each other. And the semantics of the edges are about the acute angles between the sides.

When a human tries to observe the shape of an object, the overall outline or shape will be recognized at the first, then the local features, and finally the minutiae of these features. Thus this level of detailed shape modelling process conforms to designers' cognition processes to material flow's shape structure, and the complexity leaded by describing each detail of the entire shape structure of a material flow directly will be avoided. More importantly, the global adjacent graph generated based on the information in HSG reflects the material flow's shape model explicitly.

4 SHAPE CHANGE REASONING

Compare the material's shape structure in the input state with the output state, the changes of material's shape structure can be identified. As global adjacent graph is a formal abstract descriptor for modelling material flow's shape, it makes possible that the automated decomposition for material's shape change function. With the identification and analysis of the material flow's shape based on the input and output global adjacent graph (GAG), the shape elements which compose the material's shape can be recognized, and the shape change caused by the relation changes between the shape elements can be decomposed out as functional effects of sub functions. This process is an effect decomposition process with each effect function reflects a local relation change between two shape elements. The automated material flow's shape change decomposition process has three stages, the GAG analysis process, the sub function generation process and the sub functions execution sequence planning process.

4.1 GAG analysis

Firstly, the GAG analysis process is shown as follows:

- Collect all the vertices of the input GAG to set V_{in} and all the vertices of the output GAG to set V_{out}. Thus get set $V_{unchange} = V_{in} \cap V_{out}$, which represents the same single shape elements composite the input and out shape of a material flow. The elements that are not in set $V_{unchange}$ are either a shape element generated or removed during the shape change process of the material flow.
- Get $V_{old} = (V_{in} \cup V_{out}) - V_{out}$, which refers to the single shape elements only exist in the input shape model. Then, get $V_{new} = (V_{in} \cup V_{out}) - V_{in}$, which refers to the single shape elements that only exit in the output shape model.
- Collect edges E_{in} and E_{out} from input/output GAG, and get $E_{unchange} = E_{in} \cap E_{out}$ together with E_{old} and E_{new} the same as step 1 and 2.
- Synthesize the information obtained above for the subsequent reasoning process.

So compare Figure 2(b) to Figure 4, these related information can be obtained as follows:

- The unchanged vertices: part0, part1 and part3.
- The newly generated vertices: part5.
- The disappeared (or removed) vertices: part2 and part4.
- The unchanged edges: <part0, part1> and <part0, part3>.
- The newly generated edges: <part1, part3>, <part0, part5>, <part1, part5> and <part3, part5>.
- The edges have been removed: <part1, part2> and <part3, part4>.

4.2 Sub-function generation process

The knowledge listed above is based on the analysis of the topology of input and output GAG and which reflects the shape change of a material in a coarse-grained level. Since each shape element of the input/output shape model of a material flow is defined in the input or output HST, the reason for the disappearance of V_{old} is the decomposition (remove connections between shape elements) of corresponding shape elements' and recombination (generate new connections between shape elements) for new generated shape elements (V_{new}) in output. These processes together with the change of connection relationship between $V_{unchange}$ reflect the essence of material's shape change. Thus the decomposition of a shape change from input state to output state is to find out the three types of local connection relationship change between single elements.

The *relationChange*<source, target, change mode, relation type, change trend> is used to represent the connection relationship changes. The value of change mode can be "remove", "generate" or "change". When the change mode is "change", the change trend would obtained by comparing the value in the relation constraint of the edge in input and output state. The relation change can be the functional effect for a sub-function of the material flow's shape change function. To reason out the relation changes, the finer-grained knowledge should be considered, which includes the correspondence between V_{new} and V_{old}, the correspondence between E_{old} and E_{new}, and semantic information contained in each edge.

For instance, from the semantic information contained in the edges belong to the GAG of cardboard and pyramid box, these finer-grained knowledge and relation changes shown in Table 1 can be obtained. Based on each relation change effect, a corresponding sub-function can be created. Then a relation change graph can be built as Figure 5, with red edges refer to relation change with change mode "generate" and black edges refer to the relation change relation with change mode "change".

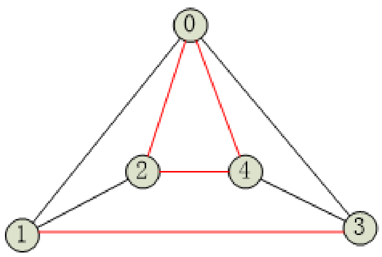


Figure 5. The adjacent graph of relation change

Table 1. Relation changes reasoned out by semantic information

Vertices correspondence:	part2 and part4 compose part5.
Edge correspondence:	<part1, part5> corresponds to <part1, part2>
	<part3, part5> corresponds to <part3, part4>
RelationChanges:	<part0, part1, change, angle, decrease>
	<part0, part3, change, angle, decrease >
	<part1, part2, change, angle, decrease >
	<part3, part4, change, angle, decrease >
	<part0, part2, generate, joint, null >
	<part0,part 4, generate, joint, null >
	<part2, part4, generate, joint, null >
	<part1, part3, generate, joint, null>

4.3 Sub-function execution sequence planning process

In Figure 5, an edge not only refers to a relation change between two shape elements, but also can be a path for determining the execution sequence for the sub-functions of the relation change effect. If an edge exists between two vertices and all of them are in a loop of this graph, the edge is coupling with other edges in the loop. For example, there is an edge between part1 and part 3 in Figure 5, and the part0, part1 and part3 compose a loop together with the edges between each two of them. Thus when the angle between part0 and part1, and the angle between part0 and part3 have changed to the desired degree in output state (the two sub functions corresponding to the two relation change have executed), the relationship between part1 and part3 also has changed to the output state. In other word, the two functional effects' performing leads to three relation changes obtained. Even if an additional function (such as adhesive bonding) may be required to keep the relation change, the coupling functional effect should be down first. In most case, a new connection relationship between single elements is generated as a resulted of the coupling relation changes. Therefore, the coupling between relation changes should be considered when planning the sub-functions execution sequence process to help the designers to find out a feasible order of function execution.

Before the execution sequence planning, we search and store the shortest loop for each relation change with change mode "generate" (for a Graph with *N* nodes, the theoretical maximum number of loops is: $C_n^n + C_n^{n-1} + \cdots + C_n^3 = 2^n - 1 - \frac{n\times(n-1)}{2} - n$. With the consideration of efficiency, we won't find all the loops in a graph but only to find the shortest one for each edge). We also use a bool state bit for each edge to record if the relation change corresponds to this edge has achieved. The sub-functions execution sequence planning will follow two rules:

- The sub-function of a relation change effect with mode "change" will have higher priority to be performed. The sub-function for "generate" a new connection relation will not to be performed initiatively.
- Each time a sub-function is executed (a relation change is achieved). If there are coupling relation change together with the relation change of that sub-function exist in a loop, the loop will be checked to determine if one of the coupling relation can be achieved indirectly (for the state bits of other relation changes are all true). If true, this relation change's corresponding sub-function is add to the execution sequence.

Thus, the planning algorithm for obtain the sub functions execution sequence is as follows:

1. Build the relation change graph based on the result of sub-function determination process.
2. Get the shortest loop of the edges whose change mode of its relation change is "generate".
3. Transform the relation change graph to a directed graph. The direction of each edge is determined by the semantic information of its relation change, from the source to the target.
4. Compute the indegree and outdegree for each vertex, choose the one with indegree = 0 as the start. Create a queue to record the execution order of sub-functions.
5. Get all the neighbour vertices of this vertex.
6. Select one of the neighbour vertices and get the edge between them.
   6.1 If it is a relation change with mode "change", add it to the execution queue.
   6.2 Otherwise, check the shortest loop of the edge to determine if all of its coupling relation change has been achieved. If true, add it to the execution queue.

7. Check if this vertex has neighbour vertices with the relation change between them which have not been achieved yet. If so, repeat steps 5 to step 7.
8. After all the size of the execution queue equal to the number of sub-functions, finish the planning process.

5 ACTION KNOWLEDGE SCHEME

After the effect decomposition by shape change reasoning, each of the sub-function has a functional effect which implies a local relation change of material's shape elements. For the essence of a mechanical system, function is the objects' behaviour result by mechanism's motion. The objects refer to the flows that the function effects on, and the mechanism is the subject of mechanical system's behaviour. Since the purpose of conceptual design is to find the working principle, which is a principle solution or a component to implement this function, the working principle searching step is followed.

The mechanism's motion is considered as a principle solution for achieve the object's behaviour. The motion of mechanism can result in different behaviours of an object under different scenarios. For example, a plastic object would be distorted in different manners because of different external forces. The tension makes it extended while the pressure makes it compacted. And whether the external force is a pressure or tensile depends on the subject of this force. The subject's motion direction and the contact position on the object will result in different effects. We use the term *action* to refer to the mechanism's motion forms in a specific application scenario and provide the certain functional effects. Thus, the knowledge contained in an action can be seemed as a working principle.

For developing a general representation schema for action knowledge, the subject behaviour to generate the action, the specific application scenarios, as well as the corresponding object behaviour, should be considered. The subject behaviour is the mechanism's motion form which corresponding to the working motion of working principle, the object behaviour is the functional effect acting on flow, and the application scenario can be seem as the working surface (e.g., the direction of the force, the position the force act on the material flow). The general schema for representing action knowledge is as shown in Table 2. Therefore an action can be the principle solution for a given functional effect, and which connect the system-level functional analysis to the mechanism-level motion design to integrate the whole conceptual design process.

Table 2. The knowledge representation scheme for an action

Action
Name Description
The knowledge of object and object's behaviour： • Object: Type //The type of a flow that a function effects on characteristic //the shape, the material characteristic and other properties of the object • Function effect : shapeChange: <inShape outShape> relationChange: <source,target,relationType,changeMode[generate|remove|increase|decrease|keep]> typeChange: <inType， outType> valueChange: <property， changeTrend>
Application scenario: //The knowledge for work head design • working surface type: point | line | surface| body • direction: axial | radial | tangential | perpendicular | parallel • Position : Settled | sequential
Working motions: //the knowledge for mechanism design • Type: translation | rotation | Helical | trajectory • Continuity: continuous | intermitted • Direction: unidirectional | bidirectional
Physical Effect: //The corresponding physical principle • Principle name • Physical low

6 IMPLEMENTATION AND CASE STUDY

The proposed shape representation approach and shape change function decomposition process is implemented in IBM Rhapsody platform as a plug-in. First, in order to model the shape of participate input/output material flows, the meta classes of SysML/UML are extended. A set of stereotypes are established in a profile as the meta model elements. The BDD (block definition diagram) of SysML is used to define the overall function together with input and output flows. The HSTs of input/output flow's shape are also defined in a BDD respectively to model the structure of shape elements which compose the material's shape. Then the IBD (internal block diagram) is used to describe the local adjacent graph for abstracting local connection relations between the shape elements in a certain level of detailed. Therefore, designers can use metamodel elements in SysML to depict the abstractive model of material's shape in the early stage of conceptual design, and these shape models are readable with the APIs provided by Rhapsody. After parsing the information contained in input and output material shape's HSG model in Rhapsody, the corresponding GAG can be generated automatically. Moreover, the sub-functions of relation changes for the global shape change can be reasoned out based on analysing semantic information contained in GAGs. Then the planning algorithm will be performed to obtain the execution order of each sub-function.

We use a complex articulated carton box folding function in (Liu and Dai, 2003) as a case study to validate our material flow shape modelling and shape change reasoning method. Figure 6 shows the definition of the overall function of the box folding. The input flow is the cardboard of the carton box shown in Figure 7. The HSG of it is defined in Figure 8, which shows that the shape of the main board consists of five shape elements: base part, part3, right part, up part and the down part. Except that the base and part3 is single element, the remaining shape elements are all composite elements. So the single elements part4 to part14 are defined as the leaf node of this HST, they compose the composite elements in level 1. The local adjacent graph of the root node is model as an IBD shown is Figure 9.

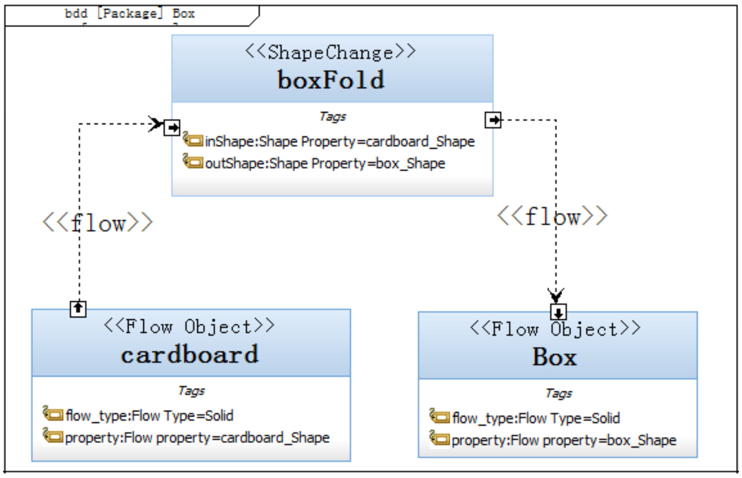


Figure 6. The overall function

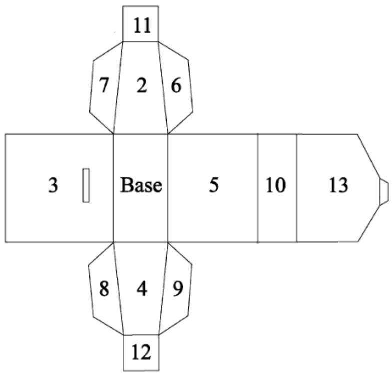


Figure 7. The cardboard

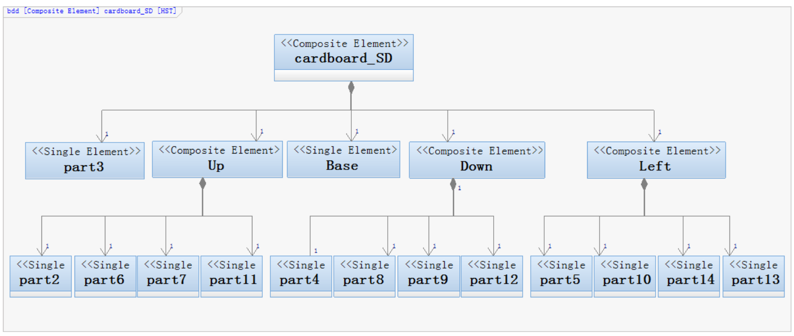


Figure 8. The HST for the shape of the cardboard

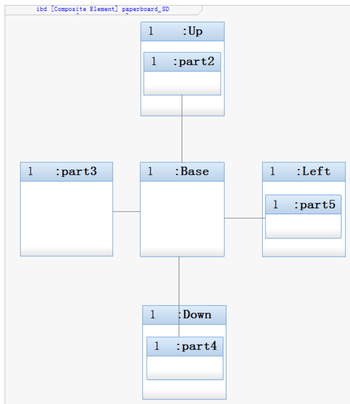


Figure 9. LAG of the root node

Not that each connection in Figure 8 and each block in Figure 8 and Figure 9 contained the semantic information of the connection relationship knowledge or the information of the shape element, they cannot be presented for the limit of space. As all the LAG of the cardboard's composite elements and the HSG of the output box's shape have been modelled in Rhapsody similarly, we execute the shape change function decomposition process to find out the sub function execution sequence to achieve the shape change for folding the cardboard to box. The result is shown in Figure 10, the sub functions of relation change with their execution sequence are created by function decomposition process.

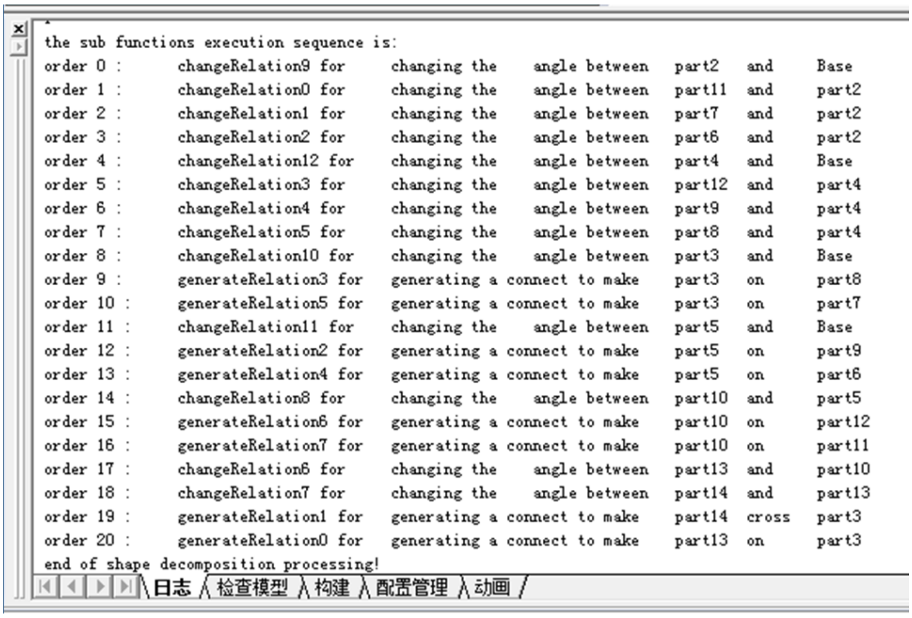


Figure 10. The result of function decomposition for shape change

After the shape change reasoning and sub-function execution sequence planning process, the function decomposition for a material's shape change is finished. For the further support of determining the action to implement the functional effect, the action knowledge representation scheme can be used to build a knowledge repository for action retrieving. The knowledge contained in a certain action will used to inspire designers to design mechanism motion with the particular working surface to implement the required function effects.

7 CONCLUSION

Conceptual design is such an important design phase that more than 70% innovation generated in this stage. Thus the function modelling and decomposition in conceptual design has been paid more and more attention by researchers. However, there still has not a representation method support modelling material flow's shape structure in this stage. Thus it is hard to do the function decomposition for working machines. Therefore, in this work, we focused on the function modelling and decomposition for working machines. A representation method for material flow's shape structure in conceptual design is proposed, and an automated function decomposition process for shape change. The main contribution of this paper is as follows:

1. A representation method is proposed for modelling material flow's shape structure in the early stage of conceptual design. This representation is independent of specific geometric data structures (such as B-rep). The shape of material flow is abstracted by the topology relations among its shape elements as adjacent graphs. And based on the stereotype extension mechanism of SysML/UML, a profile of metamodel elements are created on IBM Rhapsody platform, so designers can represent function and material flow on the platform easily.
2. An adjacent graph based reasoning method is present for decomposing the complex shape change of the material flow into multiple functional effects. Each of the functional effect corresponds to a single local shape change (such as the relation change between two shape elements) of the material.
3. A general scheme for represent action knowledge is developed. The action knowledge includes

the subject behaviour, the object behaviour and the application scenario. It can be used for retrieving principle solutions for a specific functional effect, and provides the knowledge for the mechanism design after function decomposition. Therefore the action becomes a bridge that connects the conceptual design in function level of a mechanical system to the conceptual design in an embodiment level for mechanism motion.

There are still some shortcomings of this work. Other factors lead to a shape change and different type of shape change other than relation change should be taken into account. In the current, the shape representation model and function decomposition method is only applicable for cardboard folding, the shape descriptor should be improved and more strategies should be unutilized in function decomposition to make this method more general. In order to support the whole conceptual design phase, the different function decomposition strategies, includes effect decomposition, shape reasoning and action searching, should be integrated together. Besides, to aid the conceptual design in practice, a large number of general action and domain specific action should be defined in an action repository.

ACKNOWLEDGMENTS

The authors are appreciative of the support received from the NSF of China (61173126, 51205247), the 863 High-Technology Project of China (2011AA100804) and the Zhejiang Provincial Natural Science Foundation of China (R1110377, LZ12F02001).

REFERENCES

Pahl, G. and Beitz, W. (1988) Engineering Design. Berlin:Springer.

Stone, R.B., Wood, K.L. (2000) Development of a functional basis for design. Journal of Mechanical Design, Vol.122, No. 4, pp. 359-370.

Sen, C., Summers, D.J., and Mocko, G.M. (2011) A protocol to formalise function verbs to support conservation-based model checking. Journal of Engineering Design, Vol. 22, No. 11-12, pp. 765-788.

Sen, C., Summers, D.J., and Mocko, G.M. (2013) A Formal Representation of Function Structure Graphs for Physics-Based Reasoning. Journal of Computing and Information Science in Engineering. Vol. 13, No. 2, pp.021001

Yuan, L. and Liu, Y. A. (2014) Hierarchical Material Flow Based Automated Functional Decomposition for Conceptual Design of Working Machines. Proceedings of the ASME 2014 IDETC/CIE 2014.August 17-20, 2014, Buffalo, New York, USA

Li, Z., Zhou, X., and Liu, W. (2014) A geometric reasoning approach to hierarchical representation for B-rep model retrieval. Computer-Aided Design. In Press, Accepted Manuscript, www.ejournal.com (http://dx.doi.org/10.1016/j.cad.2014.05.008) accessed on (2014.11.8).

Liu H. and Dai J. (2003) An approach to carton-folding trajectory planning using dual robotic fingers. Robotics and Autonomous Systems, Vol. 42, No.1, pp. 47-63.

Sun,Z., Liu, Y. (2011) An Automated functional Decomposition Approach for System Design of Mechatronic Products. Proceedings of ACDDE 2011,Shanghai,China

Zhang, Z., Liu,Z., Chen,Y., Xie,Y., Knowledge flow in engineering design:an ontological framework. Processdings of the Institution of Mechanical Engineers, Part c: Journal of Mechanical Engineering Science, 2013, 227(4):222-232.

Chen,Y., Zhang,Z., Huang, J., Xie, Y. (2011) Toward a scientific model of function-behavior transformation. International Conference on Engineering design,ICED11.

Chiou, S.J., Kota, Sridhar. (1999) Automated conceptual design of mechanisms. Mechanism and Machine Theory, Vol.34, No.3, pp.467-495

Han, Y.H., Lee, K. (2006) A case-based framework for reuse of previous design concepts in conceptual synthesis of mechanisms. Computers in Industry, Vol.57, No.4, pp. 305-318.