DOI: https://doi.org/10.46354/i3m.2019.emss.036

MODELING FOR COMPONENT RELATIONS IN ROBOTIC DISASSMEBLY

Xiang Li[(a)], Yuanjun Laili[(b)], Lin Zhang[(c)], Lei Ren[(d)]

[(a),(b),(c),(d)]School of Automation and Electrical Engineering, Beihang University

[(a)]lllixxxiang@buaa.edu.cn, [(b)]lailiyuanjun@buaa.edu.cn, [(c)]zhanglin@buaa.edu.cn, [(d)]renlei@buaa.edu.cn

ABSTRACT

Robotic disassembly is a critical technology to achieve automatic disassembly in remanufacturing. However, industrial robots cannot recognize component relations of specific products with various unpredictable states. Therefore, a model for component relations is of great necessity for disassembly optimization problems like Disassembly Sequence Planning (DSP) and Disassembly Line Balancing Problems (DLBP). This paper first introduces the most commonly used models of component relations in three categories. The characteristics of different models are analyzed and compared from the aspects of transformational relations and applications. Finally, suggestions are given as a reference for choosing a suitable component relation model.

Keywords: modeling of component relations, robotic disassembly, disassembly sequence planning, disassembly line balancing

1. INTRODUCTION

Remanufacturing refers to the process of rebuilding end-of-life (EoL) products to their original quality standard (Ilgin and Gupta 2016). It has gained great attention due to both the economic and ecological benefits it creates (Guide Jr 2000). Product disassembly is regarded as the core step of remanufacturing (Tang, et al 2004). However, traditional disassembly process relies on many human workers, which results in great labor costs.

Research efforts in robotic disassembly have been made to increasingly replace manual disassembly. They can execute batch disassembly of easy products independently. The introduction of robots raises several disassembly optimization problems, such as disassembly sequence planning (DSP) and disassembly line balancing (DLB), which are aimed at achieving high efficiency and low time consumption by robots. The preliminary of optimizing those problems is to obtain the internal structure information of EoL product with various unpredictable states. Nevertheless, merely a CAD model is insufficient for getting these component relations.

In order to model the inner structure of products for robotic disassembly, researchers have presented several models. One of the most natural way to represent the connection information between components of a product is liaison graph (Bourjault 1984). Each node of the graph defines a component, while an edge between two nodes indicates that the two components contact to each other. Then matrix-based model of liaisons called interference matrix was presented by Dini and Santochi (1992). To simplify the matrix, Huang, et al (2002) introduced disassembly matrix, which is the integration of interference matrices in all disassembly directions. Except these matrix-based models, graph-based models are also used in representing component relations, like joint precedence graph (Riggs, et al. 2015), AND/OR graph (De Mello and Sanderson 1990) and connector-based precedence graph (Tseng, et al. 2004). Matrix-based and graph-based models are combined by some researchers to convey more information of products. Zhang and Kuo (1997) designed a component fastener graph accompanied with two matrices, adjacency matrix and fasten matrix. Disassembly Petri net is also a popular hybrid-based model with many variations in models of component relations (Zhou and Venkatesh 1999; Guo, et al. 2015). A lot of models are provided but a new issue arises that there are so many models that researchers are confused of choosing a more appropriate one for their problems.

Therefore, this paper compares the most frequently used models of component relations and provides a reference for researchers. Section 2 is devoted to introducing three categories of product model. A discussion of these models from the perspective of transformation and application comparison is shown in Section 3. Finally, section 4 concludes this paper and summarizes the future works.

2. TYPICAL MODELS OF COMPONENT RELATIONS

To obtain feasible sequences from a more comprehensive model, researchers have proposed several kinds of matrices and graphs as the preliminaries of disassembly planning. These models can be clarified as three categories, i.e., matrix-based model, graph-based model, hybrid-based model, as shown in Table 1.

Table 1: Typical Models of Component Relations

No.	Category	Name	References
1	Matrix	Interference matrix (IM)	Dini and Santochi (1992) Huang and Huang (2002)
2		Disassembly precedence matrix (DPM) (Static DPM & Dynamic DPM)	Güngör and Gupta (2001) Tao, et al. (2018)
3		Immediate preceded matrix (IPM)	Kalayci and Gupta (2013)
4	Graph	Task precedence diagram (TPD)	Prenting and Battaglin (1964)
5		Joint precedence graph (JPG)	Riggs, et al. (2015)
6		AND/OR Graph (AOG)	De Mello and Sanderson (1990)
7		Transformed AND/OR Graph (TAOG)	Koc, et al. (2009)
8		Connector-based precedence graph (CPG)	Tseng, et al. (2004)
9		Disassembly constraint graph (DCG)	Li, et al. (2005)
10	Hybrid	Component-fastener graph (CFG)	Zhang and Kuo (1997)
11		Disassembly hybrid graph model (DHGM)	Zhang and Zhang (2010)
12		Disassembly Petri net (DPN)	Zhou and Venkatesh (1999) Guo, et al. (2015)

2.1. Matrix-based Models

Commonly used matrix-based models include interference matrix, disassembly precedence matrix and immediate preceded matrix.

(1) Interference Matrix

Interference matrix expresses the interference relations between two components along one or more principal directions.

Assume a product composed of n components $E=\{e_1,e_2,\cdots,e_n\}$, the integrated interference matrix in direction (X-, X+, Y-, Y+) is a square matrix of order n as defined in Eq. (1).

The variable $h_{ijk}=1$ means that e_i impedes e_j in direction k, otherwise $h_{ijk}=0$. As no component interferes with itself, h_{mmk}, $m\in[1,n]$ equals 0. The interference matrix can be automatically extracted from CAD model of the product (Zhang, et al. 2017).

Huang, et al. (2002) proposed to use the Boolean OR operator to check each component that if there exists one direction in which no precedent component hinders its removal. An example product and its interference matrix are given in Figure 1.

$$\mathbf{H}_{\text{int}\,eg}=\begin{bmatrix} h_{11x-}h_{11x+}h_{11y-}h_{11y+} & h_{12x-}h_{12x+}h_{12y-}h_{12y+} & \cdots & h_{1nx-}h_{1nx+}h_{1ny-}h_{1ny+} \\ h_{21x-}h_{21x+}h_{21y-}h_{21y+} & h_{22x-}h_{22x+}h_{22y-}h_{22y+} & \cdots & h_{2nx-}h_{2nx+}h_{2ny-}h_{2ny+} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1x-}h_{n1x+}h_{n1y-}h_{n1y+} & h_{n2x-}h_{n2x+}h_{n2y-}h_{n2y+} & \cdots & h_{nnx-}h_{nnx+}h_{nny-}h_{nny+} \end{bmatrix} \quad (1)$$

In addition, interference matrix is usually used together with many complementary matrices such as contact matrix and connection matrix proposed by Dini and Santochi (1992).

(2) Disassembly Precedence Matrix

There are two types of disassembly precedence matrix (DPM) to show the precedence relations, static DPM and dynamic DPM. Static DPM and its variations mentioned in later references (Tseng, et al. 2011; González and Adenso-Díaz, 2006) somehow has more similarities to interference matrix.

As the precedence relations vary during the disassembly process, dynamic DPM is established by Tao, et al (2018) to reflect these changes.

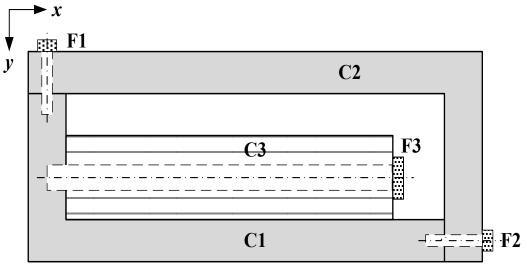
Dynamic DPM consists of four sub-matrices, a connection matrix between components and fasteners (CFM), a motion constraint matrix for fasteners (FCM), a motion constraint matrix for components (CCM), and a precedence matrix between fasteners (FFM). Element in these matrices represents the disassembly precedence between fasteners and components.

If some components and fasteners can be dismantled as a subassembly, dynamic DPM will be divided into several sub-DPMs. Dynamic characteristic here refers to dimensionality reduction of the disassembly sequencing, and therefore reduces computing cost and difficulty of searching feasible solutions.

(3) Immediate Preceded Matrix

Immediate preceded matrix (IPM) is also termed as immediately preceding matrix (Kalayci and Polat 2014). Each element in an IPM defines a disassembly operation/task instead of a component/fastener. Therefore, the task precedence relations are shown by IPM$[y_{ij}]_{n\times n}$, where y_{ij} is equal to 1 if task i should be executed after task j, otherwise it is 0.

Using IPM alone is not enough for disassembly line balancing. It is usually accompanied with an assignment matrix $[x_{ij}]_{n\times m}$ to state the partition of total tasks, where *m* denotes the number of workstations (Kalayci and Polat 2014; Kalayci and Gupta 2013). If part *j* is assigned to station k, corresponding element x_{jk} equals to 1, otherwise it is 0.



(a)

	C1	C2	C3	F1	F2	F3	OR
C1	0000	0110	0110	1111	1111	1111	1111
C2	1001	0000	1001	1111	1111	1001	1111
C3	1001	0110	0000	0000	0000	1111	1111
F1	1101	1101	0000	0000	0000	0000	1101
F2	1011	1011	0000	0000	0000	0000	1011
F3	1011	0110	1011	0000	0000	0000	1111

(b)

Figure 1. A simple product example and its interference matrix

2.2. Graph-based Model

Typical graph-based models are task precedence diagram, joint precedence graph, AND/OR graph, Transformed AND/OR graph, connector-based precedence graph and disassembly constraint graph.

(1) Task Precedence Diagram

Task precedence diagram (TPD) is proposed mainly in solving assembly/disassembly line balancing problem. Each node in the graph represents a task required in assembly/disassembly. Each connecting arc with a specific direction demonstrates the precedence relationship between the two connected nodes.

The advantages of TPD are its simplicity and elimination of redundancies (Lu and Li 2003). As a task-based diagram, it can be regarded as the graph-based model of IPM and derived from other graphs such as AND/OR graph. The topology of TPD for the same product may vary due to different choices of disassembly tasks. Hence, it is used to represent the assembly/disassembly of products with fixed tasks. (Koc, et al. 2009)

(2) Joint Precedence Graph

JPG is originally used for assembly line balancing. Riggs, et al. (2015) validates its usability in disassembly line balancing. It is an acyclic graph which can be seen as a composition of a TPD and a group of node weights representing the weighted average processing time of each task respectively. It is denoted as $G=(V,E,\overline{\mathbf{t}})$ with the definitions that the node set V contains all tasks for disassembling a product, the directed arc set E reflects the precedence relations between each two nodes. A vector $\overline{\mathbf{t}}$ represents the average processing times of these nodes, which can be obtained in Eq. (2).

$$t_i=\sum_{q\in Q}p_q t_{iq} \qquad \forall i\in I \tag{2}$$

There are in total of *Q* models. t_{iq} represents the processing time of the i^{th} task. p_q is defined as a demand portion to represent the weight of the q^{th} model. These weights satisfy $\sum_{q\in Q}p_q=1$. Therefore, t_i is the resultant average processing time of node *i*. The nodes of the resultant graph are the union set of the sub-graphs correspond to different models. An example JPG is shown in Figure 2. The arcs of the resultant graph are obtained by Eq. (3). Redundant arcs are defined as the ones that not exist in the critical path of a process.

$$E=\bigcup_{q\in Q}E_q\setminus\{\text{redundant arcs}\} \tag{3}$$

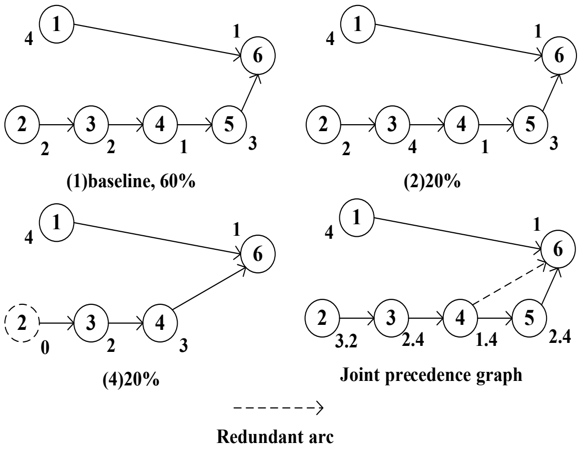


Figure 2: An Example JPG

(3) AND/OR Graph

AND/OR graph (AOG) is introduced by De Mello and Sanderson (1990) to connect disassembly task, components and subassemblies. Nodes in AOG indicate possible subassemblies and components. Parent and children subassemblies are connected by a pair of directed arcs representing disassembly tasks. The arcs are AND-type. If there are multiple optional operations for a parent node, OR-type relation is used between different pairs of AND-type-arcs to show possible disassembly processes. A disassembly sequence will be generated by choosing OR-type-arcs with connected sub-graphs.

AOG enumerates all possible operations for disassembling a product. One can traverse the graph to find the optimal disassembly sequence (Ghandi and Masehian 2015), but it is not practical. In order to avoid excessively complex graph search, Lambert (1999) proposed to use hyper-arcs to avoid repeated nodes and simplify the graph. An improved AOG of the sample product of Figure 1(a) is given in Figure 3.

(4) Transformed AND/OR Graph

Transformed AND/OR graph (TAOG) is introduced by Koc, et al. (2009). It is a clearer and more efficient model than AOG.

There are two types of nodes in a TAOG, i.e., artificial node and normal node. Artificial node A_i defines subassembly or component to be disassembled, while normal node B_i represents possible disassembly operation. Therefore, the two kinds of nodes are connected alternately. A group of OR-type arcs points from an artificial node to normal nodes defines optional operations. Only one of these operations can be executed at one time. A set of AND-type arcs that start from a normal node and end at several artificial nodes indicates the resultant parts by the specific operation.

With the precedence relations still satisfied and nodes connected by the two kinds of arcs, a TAOG can be formed or transformed from an AOG.

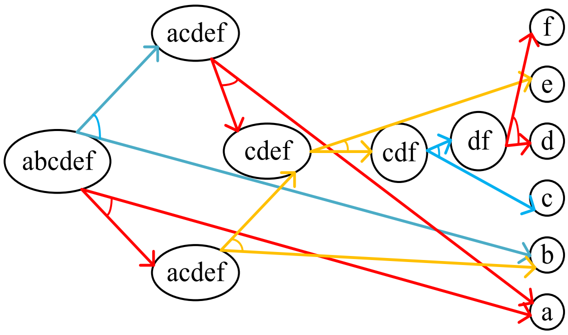


Figure 3: Hyper-arc AND/OR Graph of the Product in Figure 1(a). a to c and d to f Correspond to F1 to F3 and C1 to C3 Respectively

(5) Connector-based Precedence Graph

Connector-based precedence graph (CPG) is originally proposed to deal with assembly sequence planning by Tseng, et al. (2004). Considering the definition of connectors provided in the paper, a product can be decomposed into a set of connector-based elements, thus the graph is also applicable in disassembly problems. A node in CPG refers to a connector or fastener, with its assembly/disassembly information, such as disassembly directions, tools, and precedence constraints. In assembly problems, CPG has a start point representing the state that all connectors are not assembled yet and a terminal point representing full product. Directed arc between two nodes shows their precedence relations.

(6) Disassembly Constraint Graph

Disassembly constraint graph (DCG) introduced by Li, et al. (2005) minimizes model complexity and simplifies the process of generating disassembly sequences for selective disassembly.

A DCG is defined as a triple $M=\{V,E,DE\}$. V is a node set containing components and subassemblies which cannot be dismantled further. E is an undirected edge set containing disassembly contact constraints between components and also represent the disassembly operations. DE is a directed edge set describing precedence information between components. Li, et al. (2005) have developed a method to identify non-contact adjacent nodes and subassemblies in a DCG within a short time.

2.3. Hybrid-based Model

By hybridizing matrix and graphs, models like component-fastener graph, disassembly hybrid graph model and disassembly Petri net, can extract more information from CAD model.

(1) Component-fastener Graph

Component-fastener graph, denoted as $G_c=(V,E)$, is presented by Zhang and Kuo (1997).

It can be retrieved from a CAD model providing required information, like the name of components, disassembly method and connection relationship between components. The n components are labeled as $\{v_1,v_2,\cdots,v_n\}$ in the vertex set V. The relations between these vertices are indicated as a set of undirected edges $E=\{e_1,e_2,\cdots,e_m\}$. All the component and fastener information in need is included in the vertex, such as its name, weight and material type.

Elements in two auxiliary matrix, adjacency matrix E_c and fasten matrix F_c are defined respectively in Eq. (4) and Eq. (5).

$$E_{i,j}=\begin{cases}1, & \textit{if the component i is connected} \\ & \textit{with component j} \\ 0, & \textit{otherwise}\end{cases} \tag{4}$$

$$F_{i,j}=\begin{cases}k, & \textit{if the component i is connected with} \\ & \textit{component j by k fasteners } (k>0) \\ 0, & \textit{otherwize}\end{cases} \tag{5}$$

(2) Disassembly Hybrid Graph Model

Disassembly hybrid graph model (DHGM) is proposed by Zhang, et al (2010) to record both mating contact and noncontact precedence relationships among components. A DHGM is denoted as $G=\{V,E_f,E_{fc},E_c\}$, where node set V contains the minimal disassembly units (components or subassemblies). E_f contains undirected edges that denotes contact constraints. E_{fc} is a directed solid edge set in which the edges represent both contact constraints and precedence relations. Directed dotted edges in E_c define the priority relationships between two units. A DHGM of sample product in Figure 1(a) is shown in Figure 4.

(3) Disassembly Petri Net

Disassembly Petri net (DPN) varies in different definitions. In this paper, an 8-tuple Petri net presented by Guo, et al. (2015) is introduced. DPN can be defined as $\text{DPN} = (S, T, I, O, M, c, \tau, w)$, where defines a place set containing n disassembly parts (the number of components or subassemblies). T records a set of transitions representing disassembly operations. I is an input function that defines a set of directed arcs from T to S. O is an output function defining arcs from S to T. $M = \{M_0, \cdots, M_n\}$ is a mark set. Each element M_i represents the number of tokens in place S_i, $i = 1, 2, \ldots, n$. c is disassembly cost associated with each disassembly operations. τ is corresponding recycling/reuse value of each place. w is a set of weight functions corresponding to the transitions. A DPN of sample product in Figure 2(a) is shown in Figure 5.

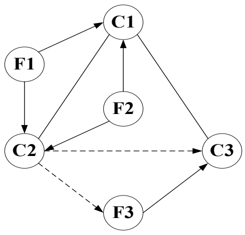


Figure 4: DHGM of Sample Product in Figure 2(a)

Two matrices, precedence matrix $\mathbf{U} = [u_{ij}]$ and disassembly matrix $\mathbf{D} = [d_{ij}]$, are also adopted (Guo, et al. 2015) and specified in Eq.(6) and (7).

$$u_{ij} = \begin{cases} 1, \; \textit{if operation } t_j \textit{ need to be} \\ \quad \textit{performed after } t_i \\ 0, \; \textit{otherwise} \end{cases} \tag{6}$$

$$d_{ij} = \begin{cases} 1, \quad \textit{if disassembly unit } s_i \textit{ is obtained} \\ \qquad \textit{via operation } t_j \\ -1, \; \textit{if } s_i \textit{ is disassembled via } t_j \\ 0, \quad \textit{otherwise} \end{cases} \tag{7}$$

3. TRANSFORMATION AND COMPARISON BETWEEN DIFFERENT MODELS

Twelve frequently used models of component relations are already introduced hereinbefore, although it is possible that there are still some remarkable ones not mentioned. Although those models vary, all of them are able to provide essential information for solving disassembly optimization problems such as DLP and DLBP. This section aims at offering a reference for those confused of choosing the model better for their problems.

A comparison between the models is given in Table 3 and the transformational relations are also included. As shown in the table, it is noticeable that three categories of models can be converted to each other in some ways. However, differences still exist between those models. By contrast, matrix-based models like IM, DPM is easier to extract from others, while hybrid ones like DHGM, DPN are more informative therefore difficult to obtain. Almost all models based on components can be derived from a CAD model. Without definitions of each task and other necessary information, a component-based model cannot be transformed directly to a task-based one. Apart from those based on either components or tasks, models based on both of them, such as AOG and TAOG, are commonly applied to solve disassembly problems as well. The two models can be converted to each other. To replace repeated and long-hour manual work, some automatic generating methods are presented. Zhang, et al. (2016) proposed an algorithm to generate improved IMs.

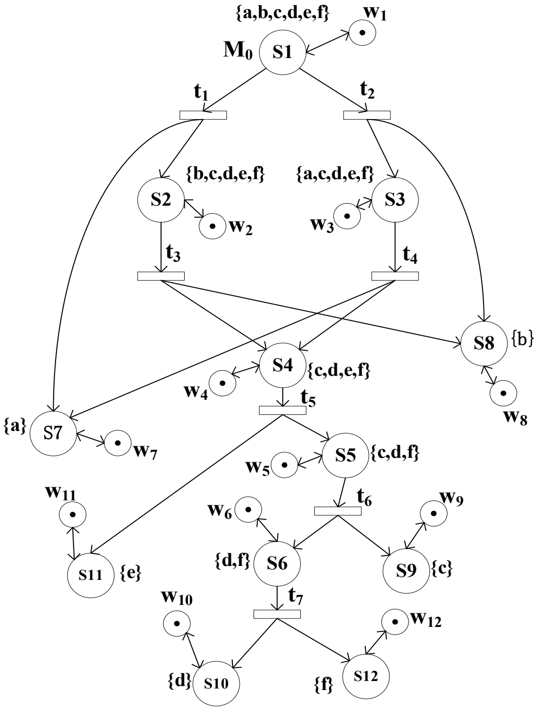


Figure 5: Disassembly Petri Net of the Sample Product Shown in Figure 1(a)

To clarify the application of each model, the conclusion is shown in the following Table 4. N, T, D and C are assumed as the number of indecomposable elements, tasks, operation directions and connectors respectively.

Combined with the former table, it can be concluded that task-based models are mostly used on solving DLBP while component-based ones on solving DSP. Therefore task-and-component-based model, AOG and TAOG, can be applied in both cases (Altekin 2016; Ren, et al. 2017; Bentaha, et al. 2014). Space complexity refers to the storage that the model takes up after formation. The complexity of hybrid-based models is relatively high due to abundant information they contain. That of AOG and DPN is uncertain because it depends on total number of disassembly choices during sequence generation.

Table 2: Transformation

No.	Category	Name	Task-based model or component-based	Transformable from CAD model?	Can be transformed to	Can be generated from
1	Matrix	IM	Component-based	YES	DPM	DPM, CPG, DCG, DHGM, DPN
2		DPM	Component-based	YES	IM	IM, CPG, DCG, DHGM, DPN
3		IPM	Task-based	NO	TPD	TPD, JPG, AOG, DPN
4	Graph	TPD	Task-based	NO	IPM	IPM, JPG, AOG, DPN
5		JPG	Task-based	NO	IPM, TPD	/
6		AOG	Task-and-component based	YES	TAOG	TAOG, DPN
7		TAOG	Task-and-component based	YES	AOG	AOG, DPN
8		CPG	Component-based	YES	IM, DPM, CFG	CFG
9		DCG	Component-based	YES	IM, DPM	DHGM
10	Hybrid	CFG	Component-based	YES	CPG	CPG
11		DHGM	Component-based	YES	IM, DPM, DCG	/
12		DPN	Task-and-component based	YES	IM, DPM, IPM, TPD, AOG, TAOG	/

Furthermore, all of the above models are capable of recording disassembly state and assisting with real-time updates, which is not mentioned in the table. Some of them works by modifying the model to eliminate the dismantled part immediately, like IM, dynamic DPM and so on. Some label the current separated progress at the node of component or task, instead of changing the original model of component relations.

Table 3: Applications

No.	Category	Name	Space(model) complexity	Disassembly sequence planning	Disassembly line balancing
1	Matrix	IM	O(N^{2}D)	√	
2		DPM	O(N^2 D)	√	
3		IPM	O(T^2)		√
4	Graph	TPD	O(T^2)		√
5		JPG	O(T^2)		√
6		AOG	Uncertain	√	√
7		TAOG	O[(2N)^2]	√	√
8		CPG	O(C^2)	√	
9		DCG	O(N^2)	√	
10	Hybrid	CFG	O(N^2)	√	
11		DHGM	O(N^2)	√	
12		DPN	Uncertain	√	

From the perspective of disassembly completion, there are three types of disassembly, complete disassembly, selective disassembly and partial disassembly. Most researches about modeling for component relations are devoted to completely disassembly. Only a few of them have been applied to solve two incomplete separations, such as IM and DCG. Nevertheless, not all parts of EoL products are valuable and reusable. What really matters for manufacturers is how to collect those components of high recovery value within the least time duration. Therefore, the latter two disassembly problems are supposed to be paid more attention.

4. CONCLUSIONS

Rapid development of remanufacturing brings challenges and opportunities for product manufacturing industry. Compared with those other new fields, there is no doubt that robotic disassembly attracts more attention. Most optimization problems in this field start with modeling for component relations. Therefore, this paper concludes three typical categories of component relation models, matrix-based model, graph-based model and hybrid-based model. Model comparison from perspectives of transformation and application were also discussed, which provides a reference for choosing more suitable component relation models in robotic disassembly optimization problems.

So far, most of the existing models of component relations are designed for completely disassembly. However, it usually takes plenty of time for robot to dismantle unnecessary parts in this kind of disassembly process. Modeling for selective and partial disassembly would be a fruitful area for future work.

CAD model of an EoL product contains various information of products. Only several models of component relations can be obtained with automatic generating algorithms if CAD models are provided. To make use of other models, manual work is still required, which undoubtably will take a lot of time. Future research can be conducted to reduce human involvement in model generation process.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (NSFC) (Grant No. 61703015) and the Fundamental Research Funds for the Central Universities.

REFERENCES

Ilgin M.A. and Gupta S.M., 2016. Remanufacturing modeling and analysis. CRC Press.

Guide Jr V.D.R., 2000. Production planning and control for remanufacturing: industry practice and research needs. Journal of operations Management, 18(4), 467-483.

Tang O., Grubbström R.W. and Zanoni S., 2004. Economic evaluation of disassembly processes in remanufacturing systems. International Journal of Production Research, 42(17), 3603-3617.

Bourjault A., 1984. Contribution à une approche méthodologique de l'assemblage automatisé: élaboration automatique des séquences opératoires. Thesis (PhD). Université de Franche-Comté.

Dini G. and Santochi M., 1992. Automated sequencing and subassembly detection in assembly planning. CIRP annals, 41(1), 1-4.

Huang Y.M. and Huang C.T., 2002. Disassembly matrix for disassembly processes of products. International Journal of Production Research, 40(2), 255-273.

Riggs R.J., Battaïa O. and Hu S.J., 2015. Disassembly line balancing under high variety of end of life states using a joint precedence graph approach. Journal of Manufacturing Systems, 37, 638-648.

De Mello L.S.H. and Sanderson A.C., 1990. AND/OR graph representation of assembly plans. IEEE Transactions on robotics and automation, 6(2), 188-199.

Tseng H.E., Li J.D. and Chang Y.H., 2004. Connector-based approach to assembly planning using a genetic algorithm. International Journal of Production Research, 42(11), 2243-2261.

Zhang H.C. and Kuo T.C., 1997. A graph-based disassembly sequence planning for EOL product recycling. Twenty First IEEE/CPMT International Electronics Manufacturing Technology Symposium Proceedings 1997 IEMT Symposium. 140-151. Oct 13-15, Austin (Texas, USA).

Zhou M.C. and Venkatesh K., 1999. Modeling, simulation, and control of flexible manufacturing systems: A Petri net approach. World Scientific.

Guo X., Liu S., Zhou M.C. and Tian G., 2016. Disassembly sequence optimization for large-scale products with multiresource constraints using scatter search and Petri nets.. IEEE transactions on cybernetics, .46(11), 2435-2446.

Güngör A., Gupta S.M., 2001. Disassembly sequence plan generation using a branch-and-bound algorithm. International Journal of Production Research, 39(3), 481-509.

Tao F., Bi L., Zuo Y. and Nee A.T.C., 2018. Partial/parallel disassembly sequence planning for complex products. Journal of Manufacturing Science and Engineering, 140(1), 011016.

Kalayci C.B., Gupta S.M., 2013. Ant colony optimization for sequence-dependent disassembly line balancing problem. Journal of Manufacturing Technology Management, 24(3), 413-427.

Prenting T.O., Battaglin R.M., 1964. The precedence diagram: A tool for analysis in assembly line balancing. Journal of Industrial Engineering, 15(4), 208-213.

Koc A., Sabuncuoglu I., Erel E., 2009. Two exact formulations for disassembly line balancing problems with task precedence diagram construction using an AND/OR graph. Iie Transactions, 41(10), 866-881.

Li J.R., Khoo L.P., Tor S.B., 2005. An object-oriented intelligent disassembly sequence planner for maintenance. Computers in Industry, 56(7), 699-718.

Zhang X.F., Zhang S.Y., 2010. Product cooperative disassembly sequence planning based on branch-and-bound algorithm. The International Journal of Advanced Manufacturing Technology, 51(9-12), 1139-1147.

Zhang W., Ma M., Li H., Yu J., 2017. Generating interference matrices for automatic assembly sequence planning. The International Journal of Advanced Manufacturing Technology, 90(1-4), 1187-1201.

Lu M., Li H., 2003. Resource-activity critical-path method for construction planning. Journal of construction engineering and management, 129(4), 412-420.

Ghandi S., Masehian E., 2014. Review and taxonomies of assembly and disassembly path planning problems and approaches. Computer-Aided Design, 67, 58-86.

Lambert A.J.D., 1999. Linear programming in disassembly/clustering sequence generation. Computers & Industrial Engineering, 36(4), 723-738.

Altekin F.T., 2016. A Piecewise Linear Model for Stochastic Disassembly Line Balancing. IFAC-PapersOnLine, 49(12), 932-937.

Ren Y., Yu D., Zhang C., et al, 2017. An improved gravitational search algorithm for profit-oriented partial disassembly line balancing problem. International Journal of Production Research, 55(24), 7302-7316.

Bentaha M.L., Battaïa O., Dolgui A., 2014. Disassembly line balancing and sequencing under uncertainty. Procedia CIRP, 15, 239-244.

AUTHORS BIOGRAPHY

Xiang Li received the BS Degree from the School of Automation Science and Electrical Engineering at Beihang University. Now she is studying as a postgraduate in the School of Automation Science and Electrical Engineering at Beihang University. Her research interests include evolutionary optimization for large-scale cloud computing system and robotic disassembly for remanufacturing.

Yuanjun Laili received the BS, MS, and Ph.D Degree from the School of Automation Science and Electrical Engineering at Beihang University. She is an assistant Professor of the School of Automation Science and Electrical Engineering in Beihang University. She is also a member of SCS (The Society For Modeling and Simulation International) and an Associate Editor of "*International Journal of Modeling, Simulation, and Scientific Computing*". Her main research interests are in the areas of intelligent optimization, modeling and simulation of manufacturing.

Lin Zhang received the B.S. degree in 1986 from the Department of Computer and System Science at Nankai University, China. He received the M.S. degree and the Ph.D. degree in 1989 and 1992 from the Department of Automation at Tsinghua University, China, where he worked as an associate professor from 1994. He served as the director of CIMS Office, National 863 Program, China Ministry of Science and Technology, from December 1997 to August 2001. From 2002 to 2005 he worked at the US Naval Postgraduate School as a senior research associate of the US National Research Council. Now he is a full professor in Beihang University. He is an associate Editor-in-Chief of "*International Journal of Modeling, Simulation, and Scientific Computing*". His research interests include integrated manufacturing systems, system modeling and simulation, and software engineering. Prof. Zhang is an IEEE senior member and a director of baord of SCS.

Lei Ren is an associate professor and the deputy head of Cloud Manufacturing Research Center at School of Automation Science and Electrical Engineering in Beihang University, and a senior research scientist at Engineering Researching Center of Complex Product Advanced Manufacturing Systems, Ministry of Education, China. He received a Ph.D degree in computer science from the Institute of Software, Chinese Academy of Sciences in 2009. His research interests include big data analytics and applications. He has published 50+ papers and got 2000+ citations according to Google Scholar. He edited a book entitled "Challenges and Opportunity with Big Data" published by Springer LNCS. He served as an associate editor of SIMULATION: Transactions of the Society for Modeling and Simulation International, and reviewers for journals such as IEEE TII, TSMC, ACM TCC. He also served as a TPC co-chair for the 19th Monterey Workshop on Big Data, 5th International Conference on Enterprise Systems, etc. He is a member of IEEE, ACM, ASME and SCS.