A new study of assembly scheduling effective constraint method in assembly planning

Meng Yu
School of Computer Science and Engineering
Guilin University of Electronic Technology
Guilin Guangxi 541004, China
mengyucoco@163.com

Abstract—**This paper establishes an assembly planning model combined with the actual assembly process of the complex equipment and based on the equipment characteristic & interactive features. Adding constraints model for complex assembly steps which ensure the assembly validity by the hierarchical constraint assembly drawing built through a detailed analysis of complex constraint relations between the blocks and layers. The fact shows that the model can effectively express the establishment of the assembly sequence based on the equipment level information and hierarchical constraint relations, and in order to achieve the assembly relationship the decomposition and timing, make the interactive process is simple and easy to operate.**

Keywords- assembly modeling; constraint relations; assembly planning

I. INTRODUCTION

As an important part in the training simulation system assembly scheduling process modeling needs to be able to fully express the assembly process. It is the focus of attention of domestic and foreign scholars that how to express the information of the assembly process. Dian-Liang Wu etc.[1 2] put forward an expression method in assembly model expression technology research of virtual assembly environment. The expression method includes hierarchy tree model, composite expression constraint part model, process-oriented restriction model and assembly model. Liu Zhenyu, etc. [3] propose a process-oriented virtual assembly model method which using assembly relationship automatic identification, movement and assembly constraint motion and assembly process information reuse to realize assembly process modeling in the virtual environment of the process-oriented product assembly modeling study. Liu Zhenyu , etc. [4] also developed a virtual assembly system to record virtual assembly modeling process by establishing the assembly historical model and the introducing the concept of assembly tasks sequences. Hakusan, etc. [5 ~ 6] summarize aggregation and constraint dependencies relationship of the assembly structure, and express the assembly model through the interaction process reflected by the timing relationship structure in A Study of The Virtual Environment-oriented Assembly Model.

II. THE SCHEDULING ASSEMBLY FACTORS SELECTION

The scheduling assembly production rule we proposed is based on production steps which means the generation of the scheduling is based on the priority of each assembly step. The key point is how to determine the priority of the each candidate step in the system. Three factors are considered in process priority determination. All these factors reflect different aspects of the urgency degree. The first factor is the latest finish time (LFT) of the step. When the delivery period of a product is determined, the latest finish time and the latest start time (LST) of the product for each step have been identified. As mentioned earlier, if the actual completion time of a step is later than its LFT, products contain this step would inevitably tardiness. LFT is a key factor in step urgency description. This is the reasons for the good performance of LFT rules. The second factor is the remaining number of steps on the path to which this procedure belongs. If the two steps have the same or very close LFT, the procedure with a larger remaining number obviously have a higher priority, because the remaining steps is meant more participation in the machine resources competition and more potential waiting time. The third factor is whether the step is in the critical path, whose remaining processing time is longest in all paths. The critical path performs a direct role in the completion of the entire product. It is noteworthy that the critical path may be changed with the manufacturing process.

The following short forms may be used:

LFTi: the latest finish time of process i;

RPSi: the remaining number of processes on the path of process i;

MPT: the average processing time for all the process;

CPi: a binary variable, 1 indicates that process i is in the critical path, 0 means opposite. Obviously, if CP equals to 1, smaller LFT and larger RPS means a higher priority of the step

A. Build Hierarchical Constraint Assemble model

HCAG (Hierarchical Constraint Assemble Graph) is based on the definition of HCAM (hierarchical constrained assembly model) and subdivision of the constraint relations, as shown in Figure 1.

A = {A1, A2, Ai, An} is a set of vertices. Ai represents an system assembly which can be an assembly group, a member or a part. Vertex represented with a circle. The connection between vertices is called edge which presents constraint relationship of the sub-components. The set of edges is constraint relation C. Each edge contains a marker Di which is the element of direct constraint relations set D. The marker presents the type of constraint relationship, which means that sub-components on both ends are restricted by the constraint relations. The constraint relationship types includes direct constraint D and indirect constraint I. Indirect constraint relations are divided into Time Constraints, State Constraints, Control Constraints, Causal Constraint, Engineering Constraints and Level Hierarchical Constraints, denoted as T, S, Ct, C \ E, E, H. As shown in Figure 1, after assembly behavior, each sub-member of the assembly object from the position posture in the assembly object to final located position posture, and the intermediate behaviors are affected by the edge marker. For example, subassemblies can be divided into three parts A11, A12, A13 through direct constraint relationship D14, D15, D16 A7, but the final position posture of the three parts are also influenced by many other constraints relations. A11 is restricted by Hierarchical Mapping Constraints, A7 & A6 State Constraints and Time Constraints, A7 and A11 is Engineering Constraints, thus form the final position posture. Other vertices analysis is similar with the example.

B. The generation of assembly sequence

Assembly sequence generation algorithm based on assembly object attribute information model and the hierarchical constraints assembly model is described as follows:

Input: assembly object attribute information model and the hierarchical constraints assembly model.

Output: assembly sequence under complicated constraints.

Step1: divide assembly object into many assembly chunks, indicated as p1, p2, ⋯, pi, pj, ⋯ ,pn. Assembly object is evidently divided into a few paragraphs or blocks. Constraint relationship between pi and pj could be direct constraints D, Hierarchical Mapping Constraints H, State Constraint S or Time Constraints T.

Step2: Move assembly chunks along the axis (space x, y, z axis) to the right position and complete the rotation action.

Step3: Carry out disassembled and assembled analysis to each block from the assembled group layer. Traverse edges in Hierarchical Constraint Assemble Graph. There are two relationships D & H between the assembly groups and the next layer.

Step4: Traverse structure trees with assemble group vertex in blocks, then combine with Hierarchical Constraint Assemble Graph to build blocks node set {Aci}. Determine the constraints of the component current node under, and whether the node need to move.

Step5: Traverse edges in Hierarchical Constraint Assemble Graph. E relation components, which is physically split, but do not need to in engineering, should stay in the original position posture. S relation components, which can be split but do not have to, should wholly move to special position posture. Then create sub-components node and establish the mapping relationship between the S component and the assembly parts layer.

Step6: Take the subassemblies node generated in Step5 as the vertex, combining with Hierarchical Constraint Assemble Graph, sub-components structure and constraint relationships to determine whether do the lower parts move.

Step7: Traverse edges in Hierarchical Constraint Assemble Graph. Split H, T, S, C \ E relation parts or components in sub-components on space axial direction. Ct relation components are retained in situ posture.

Step8: the traverse assembly parts layer the parts node {ei}, to determine whether each part of the assembly path collision and collision detection. If there is interference the several parts assembly path in space moving parts in the assembly path in the intersection, then adjusted. Such as split sub-components can from the the Step6 perform to the Step8.

Traverse parts nodes {ei} on assemble element layer. Carry out collision detection in the assemble path of each part. A collision indicates that involved parts paths are overlapped. In this situation, the path formed later should be adjusted.

Step9: If there are still sub-components after the split, carry out Step6 to Step10.

Step10: end of the algorithm.

III. CASE STUDY

A. The Comparison of proposed rules and others

In order to forcefully prove the efficiency of the dynamic scheduling rules present in this paper, we will compare the rules with other ones which have a good performance in study of assembly produce dynamic scheduling. ECO is the constraint rules proposed in this paper, which presents that the priority of the processes are determined by the property of a product, such as delivery period. Others (including two in this paper) are scheduling rules based on the process. Test these rules in 10 groups of products, compare their tardiness products numbers and time. The test results are shown in Table 1 and Table 2 . In the table L / R / C represent LFT / RPS / CP rules separately.

TABLE I. TARDINESS PRODUCTS NUMBERS COMPARISON

	Based on process				**Based on product**	
	L/R/C	***ELFT***	***LFT***	***LSD***	***ECO***	***EDD***
Group 1	80	74	88	85	97	108
Group 2	51	50	51	68	95	94
Group 3	77	87	91	82	108	102
Group 4	87	92	94	91	100	107
Group 5	77	71	76	78	108	119
Group 6	84	94	91	91	110	104
Group 7	73	81	79	83	109	111
Group 8	58	65	68	78	90	93
Group 9	65	66	64	69	89	116
Group 10	71	69	70	81	99	94
average	72.3	74.9	77.2	80.6	100.5	104.8

From Table1 and Table2, we can infer several results in reducing tardiness products numbers and time in a single assembly production of dynamic scheduling process:

(1) Scheduling rules based on process is significantly better than the one based on product.

TABLE II. TARDINESS TIME COMPARISON

	Based on process				**Based on product**	
	L/R/C	***ELFT***	***LFT***	***LSD***	***ECO***	***EDD***
Group 1	1830	1885	2161	2100	2785	3227
Group 2	1132	987	1139	1699	2844	2325
Group 3	2024	2087	2259	2118	3016	2908
Group 4	2408	2720	2721	2906	3368	3655
Group 5	1468	1326	1438	1540	2780	3424
Group 6	1912	2246	2202	2238	2943	3597
Group 7	1677	2288	2228	2262	3479	3153
Group 8	1134	1395	1357	1497	2266	2341
Group 9	1815	1786	1726	1681	2121	3208
Group 10	1860	1478	1779	1781	2667	2428
average	1726	1820	1901	1982	2826	3025

（2）The proposed rules are more effective in reducing the product tardiness than dynamic scheduling rules (LSD, LFT) based on process. Schedule tardiness of two LFT + RPS + CP rules is 6.34% less than the LFT's and 10.3% less than the LSD's. And also the rules perform better in reducing total tardiness: Schedule tardiness of the constraints rules presented in this paper is 9.2% and 12.91% less than that of LFT and LSD. It is clear that both dynamic scheduling rules in the paper can be more effective in reducing the number and total tardiness in single assembly production process.

IV. CONCLUSION

In this paper, the author establishes an assembly planning model combined with the actual assembly process of the complex equipment and based on the equipment characteristic & interactive features. Adding constraints model for complex assembly steps which ensure the assembly validity by the hierarchical constraint assembly drawing built through a detailed analysis of complex constraint relations between the blocks and layers. The fact shows that the model can effectively express the establishment of the assembly sequence based on the equipment level information and hierarchical constraint relations, and in order to achieve the assembly relationship the decomposition and timing, make the interactive process is simple and easy to operate.

REFERENCES

[1] Dominic P D D, Kaliyamoorthy S and Kumar M S. Efficient dispatching rules for dynamic job shop scheduling [J]. International Journal of Advanced Manufacturing Technology, 2004, 24(1-2): 70-75.

[2] Chiang T C and Fu L C. Solving the FMS scheduling problem by critical ratio-based heuristics and the genetic algorithm [C]. Proceeding of the 2004 IEEE Conference on Robotics and Automation, New Orleans, LA, USA, April 26 - May 1, 2004, vol.3: 3131-3136.

[3] Reeja M K, Rajendran C. Dispatching rules for scheduling in assembly jobshops - part 1 [J]. International Journal of Production Research, 2000, 38(9), 2051-2066.

[4] Mohanasundaram K M, Natarajan K, Viswanathkumar G, Radhakrishnan P and Rajendran C. Scheduling rules for dynamic shops that manufacture multi-level jobs [J]. Computers and Industrial Engineering 2002, 44(1): 119-131.

[5] Choi B K and You N K. Dispatching rules for dynamic scheduling of one-of-a-kind production [J]. International Journal of Computer Integrated Manufacturing, 2006, 19(4):383-392.

[6] Hicks C, Song D P and Earl C F. Product due date assignment for complex assemblies [J]. International Journal of Production Economics, 2002, 76(3): 243-256.

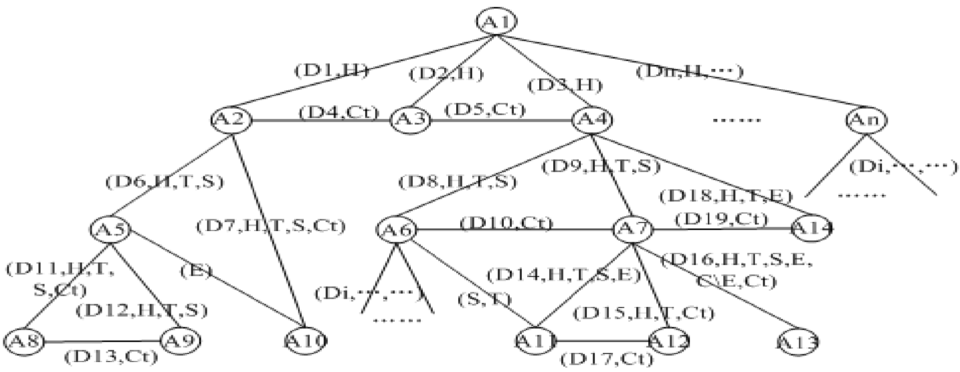

Figure 1. Hierarchical Constraint Assemble Graph