

## **AN INTEGRATED REPRESENTATION SCHEME FOR ASSEMBLY SEQUENCE PLANNING AND ASSEMBLY LINE BALANCING**

Mohd F. Rashid  
Ashutosh Tiwari  
Windo Hutabarat

School of Applied Sciences  
Cranfield University  
Cranfield  
Bedford, MK43 0AL, UK  
m.abrashid@cranfield.ac.uk

### **ABSTRACT**

In a typical assembly optimisation, Assembly Sequence Planning and Assembly Line Balancing are performed independently. However, competition has compelled the manufacturer to innovate by integrating the optimisation of both problems. To incorporate ASP and ALB optimisations into a single integrated optimisation, a clear prerequisite is the availability of integrated ASP and ALB representation. Although many assembly representation works has been proposed, none of them fully meet the requirements of integrated optimisation because they were developed independently from various needs. In this paper, an integrated representation scheme for ASP and ALB that incorporate essential optimisation information is developed. The proposed representation scheme is built based on assembly tasks and represented using precedence graph and data matrix. The outcome from presented example showed that the information for ASP and ALB optimisation can be integrated and represented using task-based precedence graph and data matrix, without discarding important attributes.

### **1 INTRODUCTION**

In manufacturing, assembly optimisation is an important process which allows a company to identify and eliminate any inefficiency in assembly. Assembly Sequence Planning (ASP) and Assembly Line Balancing (ALB) are classified among major topics in assembly optimisation because both are directly related to assembly efficiency. In conventional assembly optimisation, ASP and ALB activities are performed independently since these activities are considered as belonging to different product development and production phases (Marian, 2003). In relation with assembly optimisation, the ASP and ALB optimisation that was previously performed in serial presents a good opportunity to be integrated to reduce time-to-market for the product.

Although there are many works on assembly representation that has been done, most of the papers present an independent representation scheme for ASP or ALB problem that were built on the different basis. Most of the existing representation schemes for ASP are based on assembly parts, while ALB representation schemes are all based on assembly tasks which lead to difficulties in integrating both problems. There are several works that represent ASP problem based on assembly task, but the optimisation objectives that are applicable to this type of representation is very limited. In order to enable the ASP and ALB optimisation to be done concurrently, a single representation scheme that can be used to represent both of the problems is required.

Prior to this work, a survey on published papers in ASP and ALB optimisation from 2000 until 2010 has been conducted. From this survey, it was found that the most frequent ASP optimisation objectives

that have been used are to *minimise assembly direction change* and to *minimise number of tool change*. Meanwhile, in ALB, the dominant optimisation objectives are to *minimise cycle time*, *minimise number of workstation* and *minimise workload variance* (Rashid *et al.*, 2011). These findings show the importance and relevance of these objectives in ASP and ALB optimisation.

Previous researchers have successfully developed various ASP representation schemes to fit with their particular problem characteristics and attributes. One of the common similarities among these schemes is they are based *on assembly part*. On the other hand, the most dominant and successful representation method in ALB is precedence graph that is built *based on assembly task*. An integrated representation for both problems must use similar basis. After considering the available alternatives, their flexibility, and their success in previous works, the assembly task basis is chosen.

In this paper, an integrated representation scheme for ASP and ALB problem based on assembly task will be built. In contrast with existing work, the proposed representation will consider all of the most frequently used optimisation objectives as reported in previous survey (Rashid *et al.*, 2011). An integrated representation scheme for ASP and ALB enable both of the problems to be optimised together, which shorten the time-to-market for the product. Section 2 of this paper presents the proposed representation scheme and example of application. Section 3 discusses the proposed representation, benefits and limitations and finally followed by conclusion in section 4.

## 2 PROPOSED METHOD AND EXAMPLE

In this paper, the proposed model is divided into two main sections; Representation and Evaluation as in Figure 1. The proposed representation scheme will be built based on the assembly task. To represent both ASP and ALB problems, precedence graph and data matrix will be used. The assembly of wall rack in Figure 2 will be used to explain the proposed representation application.

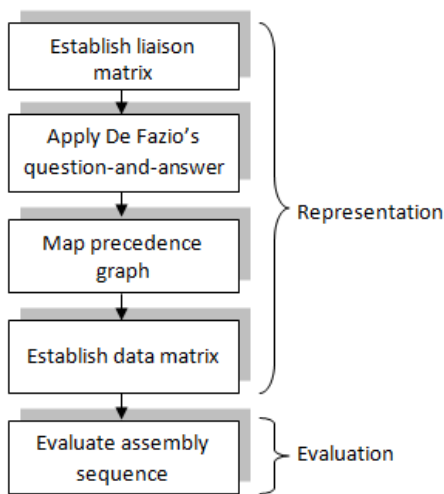


Figure 1: Flowchart of the proposed representation

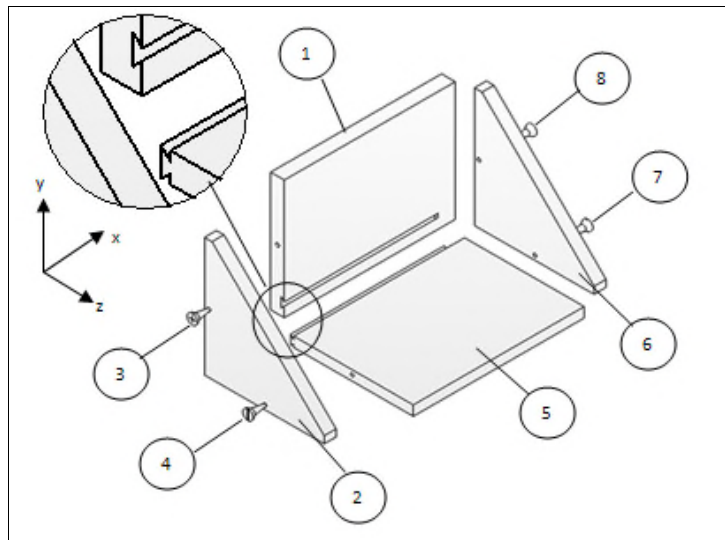


Figure 2: Assembly of wall rack

### 2.1 Basic Assumptions

Before presenting the proposed approach, certain assumptions need to be stated in order to define the representation scheme. The assumptions are listed as follows:

Assumption 1: For an assembly task, only two parts or subassemblies are involved. It might contain a combination of ‘part and part’, ‘part and subassembly’ or ‘subassembly and subassembly’.

Assumption 2: Only one part or subassembly is moved during one assembly task. Therefore, one part or subassembly will be moving part and another one will be fixed part.

## 2.2 Representation

In this approach, the ASP and ALB information such as design variables, constraint and optimisation parameters can be represented using precedence graph and data matrix. Therefore, the main activities in this stage are to establish a precedence graph and data matrix from the assembly drawing.

### 2.2.1 Establish liaison matrix

In the proposed method, the liaison matrix will be expressed using unique numbering system. For product with  $n$  parts, the relation between  $k^{\text{th}}$  and  $l^{\text{th}}$  parts is presented in the liaison matrix. If there exists an assembly relation between  $k$  and  $l$ ,  $L(k, l) = a_i$  ( $i=1,2,\dots,r$ ), otherwise,  $L(k, l)$  is left blank. Here,  $r$  is the number of liaison that exists in the assembly. For assembly example in Figure 2, the liaison matrix is shown in Table 1.

Table 1: Liaison matrix for wall rack assembly

$k \backslash l$	1	2	3	4	5	6	7	8
1	-	$a_1$	$a_2$		$a_3$	$a_4$		$a_5$
2		-			$a_6$			
3			-					
4				-	$a_7$			
5					-	$a_8$	$a_9$	
6						-		
7							-	
8								-

Table 2: Summary of the resulting De Fazio’s Q&A

Task	De Fazio’s Q&A	
	1	2
$a_1$	$a_3$	$a_2$
$a_2$	$a_1, a_3$	-
$a_3$	-	$a_1$
$a_4$	-	-
$a_5$	$a_4$	-
$a_6$	$a_3$	-
$a_7$	$a_6$	-
$a_8$	$a_3$	-
$a_9$	$a_8$	-

### 2.2.2 Apply De Fazio’s question-and-answer

Question-and-answer (Q&A) procedure is applied to determine precedence relations in assembly liaisons. This procedure is adopted from (De Fazio and Whitney, 1987) which consists of two questions for each liaison. For liaison  $i$ ;

Question 1: What liaisons must be done prior to doing liaison  $i$ ?

Question 2: What liaisons must be left to be done after doing liaison  $i$ ?

For example above, assembly task  $a_1$ , task  $a_3$  needs to be done prior to  $a_1$ , while task  $a_2$  must be left to be done after task  $a_1$ . The summary of De Fazio’s question-and-answer (Q&A) is presented in Table 2.

### 2.2.3 Map precedence graph

After answering these questions, all of the assembly precedence will be determined. The precedence constraint for this problem is given as  $C[(a_3, a_1), (a_1, a_2), (a_3, a_2), (a_4, a_5), (a_3, a_6), (a_6, a_7), (a_3, a_8), (a_8, a_9)]$ . In this set,  $(a_3, a_1)$  bring information that  $a_3$  must be done prior to  $a_1$ . The predecessor task is represented with outgoing arc and successor task is shown by incoming arc. According to the transitivity of the precedence constraints, the shortest paths between two generic nodes are removed (Fouda *et al.*,

2001). Once the precedence graph is established (Figure 3), the design space that contains design variables and constraints are defined.

### 2.2.4 Establish data matrix, $DM$

The assembly data is represented by matrix  $r \times 3$  ( $r$  is number of assembly liaison). The first column,  $D$  represents assembly direction. In this case, six major direction (+x, -x, +y, -y, +z, -z) are considered. Meanwhile the second and third column shows assembly tool,  $T$  and assembly time,  $M$  respectively. The assembly direction,  $D$  is obtained by determining fix and moving part in each assembly task. For example, the assembly direction for task  $a_1$  is the direction of bringing part 2 to be assembled with part 1, which is in +x direction. Meanwhile,  $T$  is assembly tool that involve in the assembly task. Then, the assembly time,  $M$  is acquired by performing time study of the product. The detail of assembly data for example in Figure 2 is shown in Table 3.

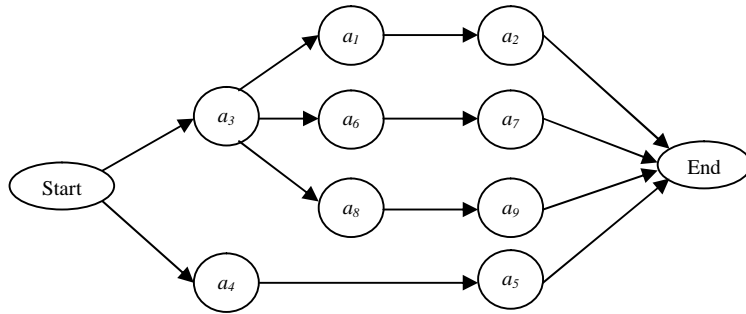


Figure 3: Precedence graph for wall rack assembly

Table 3: Data matrix,  $DM$  for wall rack assembly

Task	$D$	$T$	$M$ (time unit)
$a_1$	+x	-	4
$a_2$	+x	$T1$	12
$a_3$	+x	-	7
$a_4$	-x	-	4
$a_5$	-x	$T1$	12
$a_6$	+x	-	5
$a_7$	+x	$T2$	12
$a_8$	-x	-	5
$a_9$	-x	$T2$	12

### 2.3 Evaluation

The main purpose of assembly representation scheme is to enable assembly sequence to be evaluated for optimisation process. In this work, five optimisation objectives are considered according to frequently used objectives in previous survey: (i) Minimise number of assembly direction change, (ii) Minimise number of assembly tool change, (iii) Minimise cycle time, (iv) Minimise number of workstation, and (v) Minimise workload variation

#### 2.3.1 Assembly sequence evaluation

For a feasible assembly sequence, number of assembly direction change is counted when the next assembly task requires a different assembly direction compared with the present assembly task. In this case, similar approach also goes to second objective in determining number of assembly tool change.

Meanwhile, to evaluate third and fourth objectives, the maximum allowable cycle time,  $ct_{max}$  is required. Normally,  $ct_{max}$  is determined from number of demand or required output in assignment period. Once  $ct_{max}$  is determined, the assembly tasks can be assigned into workstations. The cycle time ( $ct$ ) for particular assembly sequence is the highest processing time among all workstations. Processing time ( $pt$ ) refers to total assembly time in a particular workstation. Once the total processing time for the current workstation is larger than  $ct_{max}$ , the present assembly task will be assigned into the next workstation.

Then, the workload variation ( $v$ ) for the fifth objective is calculated using the following formula. Here,  $ws$  refer to number of workstations in assembly line.

$$v = \frac{\sum_{i=1}^{ws} (ct - pt_i)}{ws}$$

To evaluate an assembly sequence, a feasible assembly sequence based on precedence graph needs to be established. As an example, a feasible assembly sequences  $F_1[a_4, a_5, a_3, a_1, a_6, a_8, a_9, a_7, a_2]$  is considered. In this case, number of assembly direction change ( $D$ ) and tool change ( $T$ ): Sequence  $F_1$  has  $D = 3$  and  $T = 4$ .

Table 4: Assembly direction and tool change evaluation

$F_1$	$a_4$	$a_5$	$a_3$	$a_1$	$a_6$	$a_8$	$a_9$	$a_7$	$a_2$	Total
$D$	-x	-x	+x	+x	+x	-x	-x	+x	+x	
Change	0	0	1	0	0	1	0	1	0	3
$T$	-	$T1$	-	-	-	-	$T2$	$T2$	$T1$	
Change	0	1	1	0	0	0	1	0	1	4

Assembly time ( $M$ ): Let  $ct_{max} = 20$  time unit, which is the maximum allowable cycle time for a workstation. As an example, for workstation 1 ( $ws_1$ ), the total assembly time for  $a_4$  and  $a_5$  is 16 time unit. If the assembly task  $a_3$  is also included in  $ws_1$ , the total assembly time will become 23 time unit which is exceeding the  $ct_{max}$ . Therefore, the assembly task  $a_3$  is assigned into  $ws_2$ . Similar procedure is also applied to the subsequent workstations.

Table 5: Assembly tasks allocation for  $F_1$

$ws$	$ws_1$		$ws_2$			$ws_3$		$ws_4$	$ws_5$
$F_1$	$a_4$	$a_5$	$a_3$	$a_1$	$a_6$	$a_8$	$a_9$	$a_7$	$a_2$
$M$	4	12	7	4	5	5	12	12	12
$pt$	16		16			17		12	12

From the assembly assignment for sequence  $F_1$ , the highest processing time,  $pt$  in all workstations is 17 time unit. Therefore, the cycle time,  $ct = 17$  time unit. Meanwhile to assign all of assembly task with  $ct_{max}$  constraint, five workstations are required. From there, the workload variance,  $v$  for sequence  $F_1$  can be calculated using Equation 1, which is equal to 2.4 time unit/workstation.

### 3 DISCUSSION

In this paper, an integrated representation and evaluation scheme for ASP and ALB had been proposed. The proposed representation scheme is built on the basis of assembly task as widely used in representing ALB problem. In this case, ASP representation that usually built based on assembly part is transformed into assembly task. In transforming ASP problem into task-based representation, the main problem is to determine the assembly direction because no previous work had represented assembly direction based on assembly task. This problem has been solved by redefining assembly direction according to Assumptions 1 and 2 (Section 2.1).

Previously, there are researchers that integrate ASP and ALB representation using a single representation scheme. In contrast with (Chen et al., 2002) works that only consider assembly tool for ASP problem, the proposed representation also considered the assembly direction that brings new definition of this parameter to associate with assembly task representation. Therefore, the proposed representation scheme provided more assembly parameters that can cover all important optimisation objectives as stated in Section 2.3.

Another integrated representation scheme proposed in (Tseng et al., 2008) was based on assembly connectors. In this approach, the assembly parameters refer to the connectors. For example, the assembly direction and assembly time are referred to connector direction and time to assemble the connector respectively. In contrast, the proposed representation considered the assembly parameters in terms of assembly task, which is closely linked to assembly process rather than connector, because not all

assembly components needs for a connector as shown in assembling part 1 and 5 in Figure 2. In this example, this operation does not utilise any connector.

In section 2, an example of wall rack assembly has been presented in demonstrating how to transform the assembly product to the proposed representation. This example shows that the ASP and ALB problem can be represented in a single representation scheme. In the proposed representation scheme, all information needed for ASP and ALB problems are represented using precedence graph and data matrix. The design variable that is assembly sequence can be generated from the precedence graph. Meanwhile, the optimisation constraint which is precedence constraint is presented by directed arc in the precedence graph. Then, the optimisation parameter is presented in the data matrix.

There are some limitations of the proposed approach. The first limitation is that the proposed approach cannot deal with an assembly task for more than two parts. To apply the proposed approach, the assembly task needs to be redefined by assembling only two parts at one time. The second limitation is the difficulty of generating the precedence graph. The proposed approach to generate a precedence graph might be messy for manual processing of a large assembly task. In the future, to simplify the process, it is highly recommended that the precedence graph can be generated automatically using the input of assembly task and precedence constraint.

#### 4 CONCLUSION

The best approach to ensure simultaneous optimisation of ASP and ALB problem is by employing an integrated representation of solutions. A successful integrated approach will be able to combine and retains the important characteristics of the problem such as optimisation objectives that usually used in independent works. This work shows that the proposed representation scheme is able to integrate ASP and ALB representation and at the same time consider all important optimisation objectives as been used in individual ASP and ALB.

Although there is only small difference between the proposed representation and previous work, an important contribution is a new description of assembly direction by distinguishing between fixed and moving parts in the assembly task. By using this approach, the assembly direction parameter can be associated with task-based representation. Therefore, the assembly task-based representation for ASP and ALB that takes into account all important optimisation parameters has successfully been proposed.

#### REFERENCES

- Chen, R., Lu, K. and Yu, S. (2002), "A hybrid genetic algorithm approach on multi-objective of assembly planning problem", *Engineering Applications of Artificial Intelligence*, vol. 15, no. 5, pp. 447-457.
- De Fazio, T. L. and Whitney, D. E. (1987), "Simplified generation of all mechanical assembly sequences.", *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 6, pp. 640-658.
- Fouda, P., De Lit, P., Rekiek, B. and Delchambre, A. (2001), "Generation of precedence graphs for a product family using a disassembly approach", *Proceedings of the IEEE International Symposium on Assembly and Task Planning*, 2001, pp. 226-231.
- Tseng, H., Chen, M., Chang, C. and Wang, W. (2008), "Hybrid evolutionary multi-objective algorithms for integrating assembly sequence planning and assembly line balancing", *International Journal of Production Research*, vol. 46, no. 21, pp. 5951-5977.
- Marian, R. M. (2003), "Optimisation of Assembly Sequences using Genetic Algorithm", PhD thesis, University of South Australia, ProQuest.
- Rashid, M., Tiwari, A. and Hutabarat, W. (2011), "A Review on Assembly Sequence Planning and Assembly Line Balancing Optimisation using Soft Computing Approaches", accepted in *International Journal of Advanced Manufacturing Technology*.
- Whitney, D. E. (2004), "Mechanical Assemblies: Their Design, Manufacture and Role in Product Development", Oxford University Press, New York.