Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 63 (2017) 505 – 510

www.elsevier.com/locate/procedia

The 50th CIRP Conference on Manufacturing Systems

Comparison of Sequential and Integrated Optimisation Approaches for ASP and ALB

M.F.F. Ab Rashid[a,b,]*, A. Tiwari[b], W. Hutabarat[b]

[a]*Faculty of Mechanical Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia*
[b]*Manufacturing Department, Cranfield University, Bedford, MK43 0AL United Kingdom*

* Corresponding author. Tel.: +609-424 6321; fax: +609-424 6222. *E-mail address:* ffaisae@ump.edu.my

Abstract

Combining Assembly Sequence Planning (ASP) and Assembly Line Balancing (ALB) is now of increasing interest. The customary approach is the sequential approach, where ASP is optimised before ALB. Recently, interest in the integrated approach has begun to pick up. In an integrated approach, both ASP and ALB are optimised at the same time. Various claims have been made regarding the benefits of integrated optimisation compared with sequential optimisation, such as access to a larger search space that leads to better solution quality, reduced error rate in planning and expedited product time-to-market. These benefits are often cited but no existing work has substantiated the claimed benefits by publishing a quantitative comparison between sequential and integrated approaches. This paper therefore compares the sequential and integrated optimisation approaches for ASP and ALB using 51 test problems. This is done so that the behaviour of each approach in optimising ASP and ALB problems at different difficulty levels can be properly understood. An algorithm named Multi-Objective Discrete Particle Swarm Optimisation (MODPSO) is applied in both approaches. For ASP, the optimisation results indicate that the integrated approach is suitable to be used in small and medium-sized problems, according to the number of non-dominated solution and error ratio indicators. Meanwhile, the sequential approach converges more quickly in large-sized problems. For pure ALB, the integrated approach is preferable in all cases. When both ASP and ALB are considered, the integrated approach is superior to the sequential approach.

Peer-review under responsibility of the scientific committee of The 50th CIRP Conference on Manufacturing Systems

Keywords: Assembly sequence planning; assembly line balancing; integrated optimisation; sequential optimisation

1. Introduction

Assembly optimisation involves bringing and joining parts and/or sub-assemblies to make the assembly process as efficient as possible [1]. Assembly Sequence Planning (ASP) and Assembly Line Balancing (ALB) are classified to be among major topics in assembly optimisation because both are directly related to assembly efficiency [2]. Traditionally, the ASP and ALB activities are optimised independently since both activities occur in different stages [3]. This approach is known as sequential optimisation, where the ASP is optimised before ALB. Recently, researchers have discovered the benefits of solving and optimising ASP and ALB problems together [4], [5], leading to an increased research focus on testing new or improved algorithms that operate on these combined problems [1], [6]–[8].

Various claims have been made regarding the benefits of integrated optimisation compared with sequential optimisation for ASP and ALB. In one previous work, it was claimed that the integrated ASP and ALB will enhance the quality of the solutions [4]. This is due to avoidance of reduction of the size of search space for ALB. In sequential optimisation, the search space for the second activity (i.e. ALB) will be tremendously reduced because it is formed from the output of the first activity (i.e. ASP). Besides that, integrated optimisation will reduce the error rate in manufacturing planning [5], [9]. Other than that, the integrated ASP and ALB help designers to explore the search space in one shot.

Peer-review under responsibility of the scientific committee of The 50th CIRP Conference on Manufacturing Systems
doi:10.1016/j.procir.2017.03.083

This is important to reduce optimisation time for both activities [5]. Besides that, the integrated optimisation will reduce the lead time and production cost in manufacturing [8], [10].

Although many benefits of integrated ASP and ALB optimisation were discussed by researchers, no existing work has substantiated the claimed benefits by publishing a quantitative comparison between sequential and integrated approaches. This work therefore will compare the quality of solutions of ASP and ALB optimisations that are achieved by sequential and integrated approaches. This work focuses on numerically substantiating the claim of superior solution quality. The rest of the stated benefits, such as reduced error rate and production cost, cannot be compared numerically as yet because they require actual implementation on actual assembly lines.

Substantiating the claim of superior solution quality is important because of its impact on existing practice in both ASP and ALB. It is proof that most manufacturing assembly line, even those that have been optimised using sequential ASP and ALB, are not operating in the best possible way. More importantly, it provides evidence that the integrated ASP and ALB approach is a practical way to increase the assembly line productivity even further than what has been achieved with the standalone ASP and ALB.

2. ASP and ALB Modelling

According to existing ASP and ALB works, there were a few modelling approaches implemented. The first approach is to model the problem based on the assembly components [11], [12]. Besides that, the researchers also model the problem based on the assembly task [13], [14]. Meanwhile, some researchers also model the problem based on the assembly connectors [8], [15].

In this work, we will implement task-based modelling for a simple version of ASP and ALB. The assembly problem based on assembly task is represented using a precedence diagram as shown in Figure 1. In this figure, the numerical nodes represent the assembly task, while the arcs represent precedence constraints among the assembly tasks. As an example, the outgoing arc from node 1 to nodes 2, 3 and 4 means that the assembly task 1 needs to be completed before tasks 2, 3 and 4 can be started. The assembly data for this example is presented in Table 1.

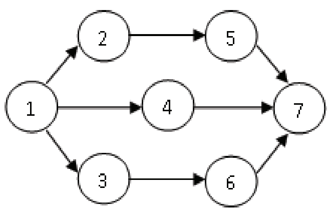

Fig. 1. Example of Precedence Diagram

In Table 1, for each task there are three types of assembly data which are required to calculate the predefined objective functions. To evaluate the ASP objectives (i.e. number of direction change (n_{dc}) and number of tool change (n_{tc}), the assembly direction and tool information for each task are needed. Meanwhile, to evaluate ALB objectives (i.e. cycle time (*ct*), number of workstation (*nws*) and workload variation (*v*)), only the assembly time information is required.

The main constraint in this work is precedence restriction which represents the compulsory sequence that must be followed in assembling a particular product. In handling this constraint, the topological sort approach is applied. Topological sort is an approach to establish feasible sequence by selecting only one available assembly task in each iteration. The topological sort procedure is repeated until all tasks are selected [16].

Table 1. Data table for Fig. 1

Task	Direction	Tool	Time
1	+x	T1	4
2	-x	T2	12
3	+x	T1	7
4	-x	T3	4
5	+x	T1	12
6	+x	T1	5
7	-x	T2	12

2.1. Objective Functions

Various objective functions have been designed and used to optimise ASP and ALB problems. A prior literature survey has collated objective functions that have been used by researchers in both problems [17]. This survey also found that the most frequently used ASP optimisation objectives are to minimise assembly direction change and to minimise the number of tool change. In ALB works, the dominant optimisation objectives are to minimise cycle time, minimise number of workstation and minimise workload variance [17].

Number of assembly direction change (n_{dc}) is counted when the next assembly task requires a different assembly direction compared with the present assembly task. In equations (1) and (2), s refers to the position of a task in a feasible assembly sequence.

$$n_{dc} = \sum_{s=1}^{n-1} d_s \tag{1}$$

$$d_s = \begin{cases} 1 & \text{if direction } s \neq \text{direction } s+1 \\ 0 & \text{if direction } s = \text{direction } s+1 \end{cases}$$

Number of assembly tool change (n_{tc}) is also counted when the next assembly task requires a different assembly tool compared with the present assembly task.

$$n_{tc} = \sum_{s=1}^{n-1} t_s \tag{2}$$

$$t_s = \begin{cases} 1 & \text{if tool } s \neq \text{tool } s+1 \\ 0 & \text{if tool } s = \text{tool } s+1 \end{cases}$$

Cycle time (*ct*) refers to the duration in between completion of one product unit with the following consecutive unit. The cycle time is important to be complied in order to