Research Journal of Applied Sciences, Engineering and Technology 6(8): 1446-1449, 2013
DOI:10.19026/rjaset.6.3968
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: October 22, 2012 Accepted: January 03, 2013 Published: July 10, 2013

Research Article

Dynamic-Model Assembly Line Scheduling

[1]Aaima Najam, [2]Faizan Ahmad and [3]Zeeshan Ahmed
[1]Department of Computer Science, COMSATS Institute of Information Technology, Lahore, Pakistan
[2]Department of Computer Science and Engineering, Beihang University, Beijing, China
[3]Department of Information Technology, Education University, Lahore, Pakistan

Abstract: The assembly line scheduling solution is restricted to two assembly lines that fulfill the requirement of small manufacturing industry by identifying the least cost path. Problem arises when large manufacturing industry comes under discussion where more than two assembly lines say three to fulfill the job, In this case two types of assembly line cost are involve: switching from one assembly line to another; switching from one station to the next. This study considers a solution for above mentioned scenario by least cost path identification, path cost calculation through back tracking, and a derived solution formula in order to reduce the computational complexity of scheduling at latter stages for n station. That provides the understanding for m number of assembly lines at the same time.

Keywords: Assembly line, configuration of assembly lines, dynamic programming, optimization-based scheduling, scheduling algorithm

INTRODUCTION

Assembly line scheduling is manufacturing problem that provides a fastest way through a factory (Hui, 2005). There are two assembly lines and each with n stations; j^{th} station on line i is known as $S_{i,j}$ and the assembly time at that station is $a_{i,j}$ An automobile chassis enters the factory and goes onto line i where i = 1 or 2, taking e_i time. After reaching the j^{th} station on a line, the chassis goes onto the $(j+1)^{st}$ station on either line (Shin and Zheng, 1991; Zhang *et al*., 1997). There is no transfer cost if it stay at the same line, but its takes time $t_{i,j}$ to transfer to the other line after station $S_{i,j}$. After exiting n^{th} station on a line, it takes x_i time for the completed auto to exit the factory (Altuger and Chassapis, 2010). The problem is to determine which station to choose from line 1 and which to choose from line 2 in order to minimize the total time through the factory for one auto (Cormen, 2001).

This study addresses one step ahead problem which arises when there is more than two assembly lines say three and there is another transfer cost $T_{i,j}$ from one station to the next while product is on the same assembly line.

To determine which station to choose from line 1, which station to choose from line 2 (Minghai and Huanmin, 2010) and which station to choose from line 3 in order to minimize the total time through the factory for an automobile?

SCENARIO

Characterization of structure:

- If j = 1, there is only one way that the chassis could have gone
- If j = 2 then

For line 1:

- Either the chassis arrive from $S_{1,j-1}$ to go through the station $S_{1,j}$
- Or the chassis arrive from $S_{2,j-1}$ to go through the station $S_{1,j}$

For line 2:

- Either the chassis arrive from $S_{2,j-1}$ to go through the station $S_{2,j}$
- Or the chassis arrive from $S_{1,j-1}$ to go through the station $S_{2,j}$
- Or the chassis arrive from $S_{3,j-1}$ to go through the station $S_{2,j}$

For line 3:

- Either the chassis arrive from $S_{3,j-1}$ to go through the station $S_{3,j}$
- Or the chassis arrive from $S_{2,j-1}$ to go through the station $S_{3,j}$

- **If j = 3, 4….n then**

Corresponding Author: Aaima Najam, Department of Computer Science, COMSATS Institute of Information Technology, Lahore, Pakistan

For line 1:

- Either the chassis arrive from $S_{1,j-1}$ to go through the station $S_{1,j}$
- Or the chassis arrive from $S_{2,j-1}$ to go through the station $S_{1,j}$
- Or the chassis arrive from $S_{3,j-2}$ to go through the station $S_{1,j}$

For line 2:

- Either the chassis arrive from $S_{2,j-1}$ to go through the station $S_{2,j}$
- Or the chassis arrive from $S_{1,j-1}$ to go through the station $S_{2,j}$
- Or the chassis arrive from $S_{3,j-1}$ to go through the station $S_{2,j}$

For line 3:

- Either the chassis arrive from $S_{3,j-1}$ to go through the station $S_{3,j}$
- Or the chassis arrive from $S_{2,j-1}$ to go through the station $S_{3,j}$
- Or the chassis arrive from $S_{1,j-2}$ to go through the station $S_{3,j}$

Recursive definition of values: Let $f_{i,j}$ be the fastest possible time to get a chassis from the starting point through station $S_{i,j.}$

Let f* be the fastest time to get the chassis all on line 1 or line 2 or line 3 then to the factory exit:

$f^* = \min(f_{1,n}+x_1, f_{2,n}+x_2, f_{3,n}+x_3)$ From starting,

For j = 1:

$f_{1,1} = e_1+a_{1,1}$
$f_{2,1} = e_2+a_{2,1}$
$f_{3,1} = e_3+a_{3,1}$

Now for j = 2:

$f_{1,j} = f_{1,j-1}+T_{1,j-1}+a_{1,j}$
$f_{1,j} = f_{2,j-1}+t_{2,j-1}+a_{1,j}$
$f_{1,j} = \min(f_{1,j-1}+T_{1,j-1}+a_{1,j}, f_{2,j-1}+t_{2,j-1}+a_{1,j})$
$f_{2,j} = f_{2,j-1}+T_{2,j-1}+a_{2,j}$
$f_{2,j} = f_{1,j-1}+t_{1,j-1}+a_{2,j}$
$f_{2[j]} = f_{3[j-1]}+t_{3,j-1}+a_{2,j}$
$f_{2,j} = \min(f_{2,j-1}+T_{2,j-1}+a_{2,j}, \; f_{1,j-1,}+t_{1,j-1}+a_{2,j} \; , f_{3,j-1}+t_{3,j-1}+a_{2,j})$
$f_{3,j} = f_{3,j-1}+T_{3,j-1}+a_{3,j}$
$f_{3,j} = f_{2,j-1}+t_{2,j-1}+a_{3,j}$
$f_{3,j} = \min(f_{3,j-1}+T_{3,j-1}+a_{3,j} \; , f_{2,j-1}+t_{2,j-1}+a_{3,j})$

Now for j = 3, 4…n:

$f_{1,j} = f_{1,j-1}+T_{1,j-1}+a_{1,j}$
$f_{1,j} = f_{2,j-1}+t_{2,j-1}+a_{1,j}$
$f_{1,j} = f_{3,j-2}+t_{3,j-2}+a_{2,j-1}+t_{2,j-1}+a_{1,j}$
$f_{1,j} = \min(f_{1,j-1}+T_{1,j-1}+a_{1,j} \; , f_{2,j-1}+t_{2,j-1}+a_{1,j} \; , f_{3,j-2}+t_{3,j-2}+a_{2,j-1}+t_{2,j-1}+a_{1,j})$
$f_{2,j} = f_{2,j-1}+T_{2,j-1}+a_{2,j}$
$f_{2,j} = f_{1,j-1,}+t_{1,j-1}+a_{2,j}$
$f_{2,j} = f_{3,j-1}+t_{3,j-1}+a_{2,j}$
$f_{2,j} = \min(f_{2,j-1}+T_{2,j-1}+a_{2,j}, \; f_{1,j-1}+t_{1,j-1}+a_{2,j}, \; f_{3,j-1}+t_{3,j-1}+a_{2,j})$
$f_{3,j}=f_{3,j-1}+T_{3,j-1}+a_{3,j}$
$f_{3,j} = f_{2,j-1}+t_{2,j-1}+a_{3,j}$
$f_{3,j} = f_{1,j-2}+t_{1,j-2}+a_{2,j-1}+t_{2,j-1}+a_{3,j}$
$f_{3,j} = \min(f_{3,j-1}+T_{3,j-1}+a_{3,j} \; , f_{2,j-1}+t_{2,j-1}+a_{3,j} \; , f_{1,j-2}+t_{1,j-2}+a_{2,j-1}+t_{2,j-1}+a_{3,j})$

Derived function:

$f_{1,j} = \{ e_1+a_{11}$ for j = 1
$\min(f_{1,j-1}+T_{1,j-1}+a_{1,j}, f_{2,j-1}+t_{2,j-1}+a_{1,j})$ for j = 2
$\min(f_{1,j-1}+T_{1,j-1}+a_{1,j} \; , f_{2,j-1}+t_{2,j-1}+a_{1,j} \; , f_{3,j-2}+t_{3,j-2}+a_{2,j-1}+t_{2,j-1}+a_{1,j})$ for j> = 3}
$f_{2,j} = \{ e_2+a_{21}$ for j = 1
$\min(f_{2,j-1}+T_{2,j-1}+a_{2,j}, \; f_{1,j-1}+t_{1,j-1}+a_{2,j}, f_{3,j-1}+t_{3,j-1}+a_{2,j})$ for j>= 2}
$f_{3,j} = \{ e_3+a_{31}$ for j = 1
$\min(f_{3,j-1}+T_{3,j-1}+a_{3,j} \; , f_{2,j-1}+t_{2,j-1}+a_{3,j})$ for j = 2
$\min(f_{3[j-1]}+T_{3,j-1}+a_{3,j} \; , \; f_{2,j-1}+t_{2,j-1}+a_{3,j} \; , f_{1,j-2}+t_{1,j-2}+a_{2,j-1}+t_{2,j-1}+a_{3,j})$ for j> = 3}

Define $l_{i,j}$ to be line no either 1 or 2 or 3 whose station j = 1 is used in the fastest way through station $S_{i,j}$ (i = 1,2,3 j = 2,3,….n) (Table 1, 2).

Define l* to the line whose station n is used in a fastest way through the entire factory.

$l^* = 3 \;\; l_{3,6} = 2, \; l_{2,5} = 2, \; l_{2,4} = 1, l_{1,3} = 1 \; , l_{1,2} = 1$

RESULTS

From Table 1 we have derived Table 3 in it we mention that line 3 is efficient and fast which we will get to know from f*which came from adding x_3 in $f_{3[j]}$ where j = 6. Table 3 shows the result of scenario discussed in Fig. 1.

f* = 23+2 = 25 so line 3 is more efficient and fast

Table 1: Computation of values in bottom up way

J	1	2	3	4	5	6
f1[j]	3	7	12	18	21	24
f2[j]	7	7	14	14	17	27
f3[j]	4	8	11	17	22	23

Table 2: Construction of optimal solution

J	2	3	4	5	6
l_1[j]	1	1	1	2	2
l_2[j]	1	1	1	2	2
l_3[j]	3	2	3	2	2

Table 3: Results with optimal path

J	1	2	3	4	5	6
f1[j]	3	7	12	18	21	24
f2[j]	7	7	14	14	17	27
f3[j]	4	8	11	17	22	23

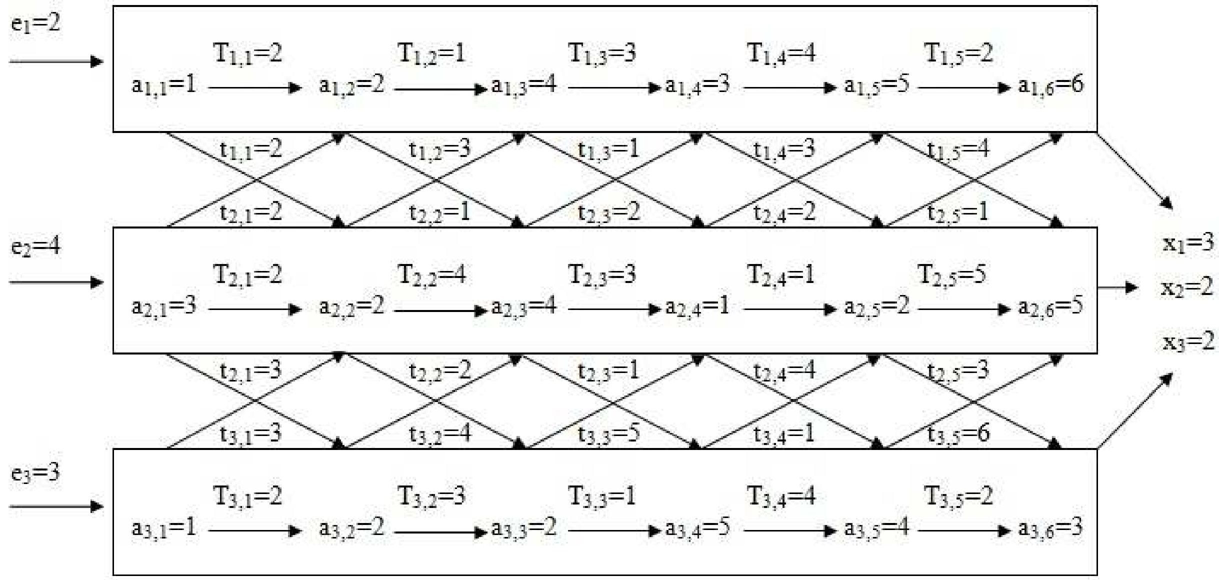

Fig. 1: Assembly lines

Algorithm: (Kaufman, 1974; Hsu, 1984) Fastest-way (a, t, e, x, n,T)

- $f_{1[1]} \leftarrow e_1+a_{1,1}$
- $f_{2[1]} \leftarrow e_2+a_{2,1}$
- $f_{3[1]} \leftarrow e_3+a_{3,1}$
- if $f_{1,1}+T_{1,1}+a_{1,1} \le f_{2,1}+t_{2,1}+a_{1,2}$
- $f_{1[2]} \leftarrow f_{1,1}+T_{1,1}+a_{1,2}$ $\quad l_{1[2]} \leftarrow 1$
- else
- $f1_{[2]} \leftarrow f_{2,1}+t_{2,1}+a_{1,2}$ $\quad l_{1[2]} \leftarrow 2$

- if $((f_{2,1}+T_{2,1}+a_{2,2} \le f_{1,1}+t_{1,1}+a_{2,2})$ and $(f_{2,1}+T_{2,1}+a_{2,2} \le f_{3,1}+t_{3,1}+a_{2,2}))$
- $f_2[2] \leftarrow f_{2,1}+T_{2,1}+a_{2,2}$ $\quad l_{2[2]} \leftarrow 2$
- else if $((f_{1,1}+t_{1,1}+a_{2,2} < f_{2,1}+T_{2,1}+a_{2,2})$ and $(f_{1,1}+t_{1,1}+a_{2,2} < f_{3,1}+t_{3,1}+a_{2,2}))$
- $f_{2[2]} \leftarrow f_{1,1}+t_{1,1}+a_{2,2}$ $\quad l_{2[2]} \leftarrow 1$
- else
- $f_{2[2]} \leftarrow f_{3,1}+t_{3,1}+a_{2,2}$ $\quad l_{2[2]} \leftarrow 3$

- if $f_{3,1}+T_{3,1}+a_{3,2} \le f_{2,1}+t_{2,1}+a_{3,2}$
- $f_{3[2]} \leftarrow f_{3,1}+T_{3,1}+a_{3,2}$ $\quad l_{3[2]} \leftarrow 3$
- else
- $f_{3[2]} \leftarrow f_{2,1}+t_{2,1}+a_{3,2}$ $\quad l_{3[2]} \leftarrow 2$

- for $j \leftarrow 3$ to n
- if $((f_{1,j-1}+T_{1,j-1}+a_{1,j} \le f_{2,j-1}+t_{2,j-1}+a_{1,j})$ and $(f_{1,j-1}+T_{1,j-1}+a_{1,j} \le f_{3,j-2}+t_{3,j-2}+a_{2,j-1}+t_{2,j-1}+a_{1,j}))$
- $f_{1[j]} \leftarrow f_{1,j-1}+T_{1,j-1}+a_{1,j}$ $\quad l_{1[j]} \leftarrow 1$
- else if $((f_{2,j-1}+t_{2,j-1}+a_{1,j} < f_{1,j-1}+T_{1,j-1}+a_{1,j})$ and $(f_{2,j-1}+t_{2,j-1}+a_{1,j} < f_{3,j-2}+t_{3,j-2}+a_{2,j-1}+t_{2,j-1}+a_{1,j}))$
- $f_{1[j]} \leftarrow f_{2,j-1}+t_{2,j-1}+a_{1,j}$ $\quad l_{1[j]} \leftarrow 2$
- else
- $f_{1[j]} \leftarrow f_{3,j-2}+t_{3,j-2}+a_{2,j-1}+t_{2,j-1}+a_{1,j}$ $\quad l_{1[j]} \leftarrow 3$
- if $((f_{2,j-1}+T_{2,j-1}+a_{2,j} \le f_{1,j-1}+t_{1,j-1}+a_{2,j})$ and $(f_{2,j-1}+T_{2,j-1}+a_{2,j} \le f_{3,j-1}+t_{3,j-1}+a_{2,j}))$
- $f_{2[j]} \leftarrow f_{2,j-1}+T_{2,j-1}+a_{2,j}$ $\quad l_{2[j]} \leftarrow 2$
- else if $((f_{1,j-1}+t_{1,j-1}+a_{2,j} < f_{2,j-1}+T_{2,j-1}+a_{2,j})$ and $(f_{1,j-1}+t_{1,j-1}+a_{2,j} < f_{3,j-1}+t_{3,j-1}+a_{2,j}))$
- $f_{2[j]} \leftarrow f_{1,j-1}+t_{1,j-1}+a_{2,j}$ $\quad l_{2[j]} \leftarrow 1$
- else
- $f_{2[j]} \leftarrow f_{3,j-1}+t_{3,j-1}+a_{2,j}$ $\quad l_{2[j]} \leftarrow 3$
- if$((f_{3,j-1}+T_{3,j-1}+a_{3,j} < f_{2,j-1}+t_{2,j-1}+a_{3,j})$ and $(f_{3,j-1}+T_{3,j-1}+a_{3,j} < f_{1,j-2}+t_{1,j-2}+a_{2,j-1}+t_{2,j-1}+a_{3,j}))$
- $f_{3[j]} \leftarrow f_{3,j-1}+T_{3,j-1}+a_{3,j}$ $\quad l_{3[j]} \leftarrow 3$
- else if $((f_{2,j-1}+t_{2,j-1}+a_{3,j} < f_{3,j-1}+T_{3,j-1}+a_{3,j})$ and $(f_{2,j-1}+t_{2,j-1}+a_{3,j} < f_{1,j-2}+t_{1,j-2}+a_{2,j-1}+t_{2,j-1}+a_{3,j}))$
- $f_{3[j]} \leftarrow f_{2,j-1}+t_{2,j-1}+a_{3,j}$ $\quad l_{3[j]} \leftarrow 2$
- else
- $f_{3[j]} \leftarrow f_{1,j-2}+t_{1,j-2}+a_{2,j-1}+t_{2,j-1}+a_{3,j}$ $\quad l_{3[j]} \leftarrow 1$

- if $((f_{1[n]}+x_1 \le f_{2[n]}+x_2)$ and $(f_{1[n]}+x_1 \le f_{3[n]}+x_3))$
- $f^* \leftarrow f_{1[n]}+x_1$ $\quad l^* \leftarrow 1$
- else if $((f_{2[n]}+x_2 \le f_{1[n]}+x_1)$ and $(f_{2[n]}+x_2 \le f_{3[n]}+x_3))$
- $f^* \leftarrow f_{2[n]}+x_2$ $\quad l^* \leftarrow 2$
- else
- $f^* \leftarrow f_{3[n]}+x_3$ $\quad l^* \leftarrow 3$

PREVIOUS WORK

In previous work, the assembly line scheduling solution is defined that is restricted to two assembly lines that fulfill the requirement of small manufacturing industry by identifying the least cost path. Problem arises when large manufacturing industry comes under discussion where more than two assembly lines say three to fulfill the job. So our work fulfills that requirement.

CONCLUSION

To fulfill the need of customer now days is a huge challenge for the manufacturer. This paper provides the dynamic solution for the fastest way through the entire factory assembly line overloading problem which deals with more assembly lines and station-to-station transfer cost that will make work flow more efficient and fluent.

FUTURE WORK

This research study is initial step to enhance the capabilities of assembly line scheduling by proposing an idea to upgrading this scheduling algorithm for three assembly lines which faces the same complexity issues as any number of assembly lines (more than three) will face. This idea could be upgrade for unlimited number of assembly lines to deal with the dynamic need of manufacturing industry as discussed above by designing a more dynamic solution.

REFERENCES

Altuger, G. and C. Chassapis, 2010. Manual assembly line operator scheduling using hierarchical preference aggregation. Proceeding of the Winter Simulation Conference (WSC), pp: 1613-1623.

Cormen, T.H., 2001. Introduction to Algorithms. 2nd Edn., MIT Press, Cambridge, Mass, pp: 277-284.

Hsu, W.L., 1984. Approximation algorithms for the assembly line crew scheduling problem. Math. Oper. Res., 9: 376-383.

Hui, S., 2005. Performance evaluation of hybrid genetic algorithm for assembly line scheduling. Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05), pp: 224-231.

Kaufman, M.T., 1974. An almost-optimal algorithm for the assembly line scheduling problem. IEEE T. Comput., 23(11): 1169-1174.

Minghai, Y. and X. Huanmin, 2010. Research on reconfigurable assembly line multi-objective scheduling decision-making. Proceedings of the IEEE International Conference on Measuring Technology and Mechatronics Automation, 3: 151-154.

Shin, K.G. and Q. Zheng, 1991. Scheduling job operations in an automatic assembly line. IEEE T. Robotic. Autom., 7(3): 333-341.

Zhang, Y., P.B. Luh, K. Yoneda, T. Kano and Y. Kyoya, 1997. Mixed-model assembly line scheduling using the lagrangian relaxation technique. Proceedings of the IEEE International Conference on Control Applications. Hartford, CT, USA, pp: 429-434.