Production Scheduling Using Mixed Integer Programming:
Case of Bread Small and Medium Enterprise at Yogyakarta

Ratriani Puspita Hastuti\textsuperscript{a,*}, Henry Yuliando\textsuperscript{b}, Ibnu Wahid Fakhrudin Aziz\textsuperscript{b}

\textsuperscript{a} Undergraduate student of Department of Agro-industrial Technology, Faculty of Agricultural Technology, University of Gadjah Mada, Indonesia
\textsuperscript{b} Lecturer of Department of Agro-industrial Technology, Faculty of Agricultural Technology, University of Gadjah Mada, Indonesia

Abstract

Small and medium enterprise (SME) needs to apply effective scheduling in order to meet the demand schedule. This is done traditionally, though, it is necessary to allocate production resources to produce product variety such as bread. The fact shows that the order fulfillment is often delayed because of incorrect scheduling. This problem can be solved by making production scheduling that have a minimum completion time (makespan) as approached by using mixed integer programming (MIP). This study is proposed to build an effective scheduling for an SME on bread products by utilize MIP. The output is then compared to the serial machine scheduling method with CDS algorithm. Both methods are having an output in minimizing makespan. The results showed that with the proposed scheduling, the SME likely able to fulfill the whole order in time.

Keywords: Flow shop; Mixed Integer Programming; Campbell, Dudek, and Smith (CDS) algorithm; Scheduling

1. Introduction

Scheduling is critical for companies such as in terms of departure time of raw materials and time to finish the product for delivering it to the consumer. Improper scheduling of production can increase cost of production,
increase idle time, and cause delay in production process. Therefore, companies need to make an effective production schedule to prevent those disadvantages.

The SME on bread product has variety range products from various flavors and size. As concerning to the case met in this study, Bangkit Bread, is an SME which produce varies of bread namely mocha, strawberry, coconut, chocolate, and choc-o-nut. Production level of this SME is about 14,000-15,000 pieces/day and increasing at the weekend to 15,000-17,000 pieces/day. However, the actual schedule has not optimal because order delivery is often late. Therefore, production scheduling is needed to shorten completion time and reduce delay.

Mixed Integer Programming (MIP) is employed to obtain a more proper schedule. As an optimization program, it can be applied to several types of problems. In addition, it is found that the bread SME uses serial machines and facilities. In this study another method namely Campbell, Dudek, and Smith (CDS) algorithm is also used to release constraint in the real application of MIP. Both methods are able to give an optimal output.

2. Materials and Methods

This study takes data of processing time, completion time, and departure time in each stage of production of bread. This data is as input for both employed models i.e. MIP and CDS algorithm. Several studies in scheduling have been introduced by distinguished researchers. Wagner in Stafford and Tseng (2003) introduced an all-integer linear programming model for the general job shop. He demonstrated integer models for the general flow shop and for the flow shop with permutation scheduling with M = 3 machines. Pochet and Wolsey (2006) stated that mixed integer linear program (MIP) is an optimization program involving continuous and integer variables, and linear constraints. The model employed is namely F1B. Sawik (2011) stated that it is a model for flow shop scheduling with one engine and no in-process buffers.

Minimize

\[ C_{\text{max}} \]  
subject to:

Part Completion Constraints

\[ c_{1k} \geq p_{ik} \quad k \in K \]  
\[ c_{ik} - c_{i-1k} \geq p_{ik} \quad i \in I, k \in K : i > 1 \]  

Maximum Completion Time Constraint

\[ c_{mk} \leq C_{\text{max}} \quad k \in K \]  

Part Departure Constraints

\[ c_{ik} \leq d_{ik} \quad i \in I, k \in K : i < m \]  
\[ c_{mk} = d_{mk} \quad k \in K \]  

No Buffering Constraint

\[ c_{ik} - p_{ik} = d_{i-1k} \quad i \in I, k \in K : i > 1 \]  

Variable Nonnegativity and Integrality Conditions

\[ C_{\text{max}} \geq 0 \]  
\[ c_{ik} \geq 0 \quad i \in I, k \in K \]  
\[ d_{ik} \geq 0 \quad i \in I, k \in K \]  

where C\text{max} = makespan

\[ i = \text{processing stage}, i \in I = \{1, \ldots, m\} \]  
\[ k = \text{job}, k \in K = \{1, \ldots, n\} \]  
\[ m = \text{number of processing stages} \]  
\[ p_{ik} = \text{processing time for part k in stage i} \]  
\[ c_{ik} = \text{completion time of part k in stage i} \]  
\[ d_{ik} = \text{departure time of part k from stage i} \]

and Campbell, Dudek, and Smith (CDS) Algorithm is written as follows.

Calculate \( k = 1 \), calculate \( t_{*,1} \) and \( t_{*,2} \).

\[ t_{1,1} = \sum_{k=1}^{K} t_{1k} \]  
\[ t_{1,2} = \sum_{k=1}^{K} t_{1, m=k+1} \]  

Use Johnson's algorithm to sort the job, where \( t_{1,1} = t_{*,1} \) and \( t_{1,2} = t_{*,2} \) then calculate the makespan.
If $K = (m-1)$ stops, select a schedule with the smallest makespan. If $K \neq (m-1)$, $K = K + 1$ and return to step 1.

3. Results and Discussions

Scheduling using MIP was performed to optimize the number of lot production to minimize makespan by considering various constraints. Priority rules used in this scheduling was Short Processing Time (SPT) with the aim to minimize makespan. Table 1 shows processing time ($p_k$), completion time ($c_k$), and departure time ($d_k$).

<table>
<thead>
<tr>
<th>Job</th>
<th>Time</th>
<th>Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Stirring</td>
<td>Mixing</td>
</tr>
<tr>
<td>Mocha Bread (1)</td>
<td>$p_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td></td>
<td>$c_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td></td>
<td>$d_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td>Strawberry Bread (2)</td>
<td>$p_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td></td>
<td>$c_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td></td>
<td>$d_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td>Coconut Bread (3)</td>
<td>$p_k$ (sec)</td>
<td>5.80</td>
</tr>
<tr>
<td></td>
<td>$c_k$ (sec)</td>
<td>5.80</td>
</tr>
<tr>
<td></td>
<td>$d_k$ (sec)</td>
<td>5.80</td>
</tr>
<tr>
<td>Chocolate Bread (4)</td>
<td>$p_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td></td>
<td>$c_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td></td>
<td>$d_k$ (sec)</td>
<td>5.81</td>
</tr>
<tr>
<td>Choco-nut Bread (5)</td>
<td>$p_k$ (sec)</td>
<td>5.80</td>
</tr>
<tr>
<td></td>
<td>$c_k$ (sec)</td>
<td>5.80</td>
</tr>
<tr>
<td></td>
<td>$d_k$ (sec)</td>
<td>5.80</td>
</tr>
</tbody>
</table>

Data in table 1 was used to perform scheduling using MIP with a solution based on branch and bound method as calculated using Microsoft EXCEL solver. Figure 1 shows the solution of scheduling using MIP with branch and bound method.

![Fig. 1. Optimal Solution MIP](image-url)
It is known that the optimal result to obtain minimum makespan was one lot for each product that is equal to one baking sheet based on the unit used in the measurement of standard time. There was addition condition to arrange job sequence. Job 1 (mocha) and job 2 (strawberry) became priority than the other products, but job 1 unnecessary to be processed first. By using SPT rule to sort the amount of time the smallest to the biggest, the result scheduling using MIP was 2-1-3-4-5 with makespan 27466.71 seconds.

Since the SME uses serial machine and facilities, taking into a heuristic method such CDS algorithm is valuable to reveal an alternative solution. Based on CDS algorithm the optimal result was obtained at k=7. Table 2 shows calculation of processing time using CDS algorithm at k=7.

Table 2. Calculation of Processing Time Using CDS Algorithm

<table>
<thead>
<tr>
<th>Stage</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1 + M2 + M3 + M4 + M5 + M6 + M7</td>
<td>25671.16</td>
<td>25670.42</td>
<td>25794.12</td>
<td>25820.22</td>
<td>25821.15</td>
</tr>
<tr>
<td>M2 + M3 + M4 + M5 + M6 + M7 + M8</td>
<td>25779.03</td>
<td>25778.24</td>
<td>25901.90</td>
<td>25927.96</td>
<td>25928.98</td>
</tr>
</tbody>
</table>

Scheduling was done by Johnson rule:

Job 2 was scheduled: 2 _ _ _ _
Job 1 was scheduled: 2 1 _ _ _
Job 3 was scheduled: 2 1 3 _ _
Job 4 was scheduled: 2 1 3 4 _
Job 5 was scheduled: 2 1 3 4 5

It can be described that scheduling with MIP and CDS algorithm in this case give the same result in term of processing. It differs only in the job sequence. Job 1 and job 2 become priority than the other products, but job 1 unnecessary to carry out first. In addition, job 1 and job 2 have to be in priority due to the different time to do fermentation which is mocha and strawberry breads are different in time needed compare to coconut and choco-nut bread. When fermentation stage starts, coconut, chocolate and choco-nut breads have been filled with its own flavor, but unlike mocha and strawberry. Therefore, it is necessary to anticipate over processing. This process requires the dough expands perfectly, hence it must go to baking process immediately.

Because of the results between MIP and CDS algorithm were just the same, both would be work out to the proposed schedule for the case. The actual schedule requires 27467.45 seconds with job sequence 1-2-3-4-5. The final step is to make a simulation to compare between the actual schedule and proposed schedule with MIP and CDS algorithm. The result is presented on following figures.

Fig. 2. (a) Actual Scheduling Simulation Period 1 Gantt chart; (b) Proposed Scheduling Simulation Period 1 Gantt chart
Gantt charts show that the proposed scheduling on period 1 provides minimal makespan of 79366.9 seconds or 22.0 hours; while the actual scheduling provides longer makespan at 22.7 hours. As well as in the second chart the proposed scheduling shortens makespan to 21.5 hours against the actual scheduling that requires 22.4 hours.

Taking into account the working time at SME starts from 07:00 am to 05:00 am in the next day production process has to finish at 5:00 am. The delivery time is from 5:00 am to 7:00 am. However, if the actual scheduling takes 22.7 hours in period 1 then the delivery will be delayed for 42 minutes and 24 minutes in period 2. Here, the proposed scheduling can fulfill demand on time in period 1 and can finish 30 minutes earlier in period 2. Finally, delivery on time will give an advantage to SME such as better preparation for distribution and administration.

4. Conclusions

Proposed scheduling using MIP and CDS algorithm approaches resulting minimum makespan. If it is applied, the job sequence should be varies product of strawberry, mocha, coconut, chocolate, and choco-nut breads. In further the simulation compared to the actual give an advantage of time in comply with available warming time. This will be a benefit for SME to manage the time more effective in order to give best response to the demand requirement in punctual delivery schedule.

References