Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 8 (9): 1706-1712, 2008
ISSN 1812-5654

Determination of Inter-Stage Idle Times in Scheduling Design of a Three Stage Zero Wait Batch Process

Amir Shafeeq, M.I. Abdul Mutalib, K.A. Amminudin and Ayyaz Muhammad
Department of Chemical Engineering, Universiti Teknologi Petronas,
31750 Tronoh, Perak Darul Ridzuan, Malaysia

Abstract: The present research proposes a novel approach based on matrix to determine the makespan of a batch process adopting zero wait transfer policy. The proposed approach uses simple mathematical formulae, applied on a matrix representing the batch process recipes according to a selected production sequence. Rearrangement of the matrix rows according to the varied production sequences possible for the specified batch process recipes enables the makespan to be determined for each sequence. Designer is then provided with the production sequence options with its corresponding makespan from which a selection could be made according to the process conditions.

Key words: Scheduling, batch process, matrix approach, zero wait

INTRODUCTION

It is usually practiced in chemical industry especially which is dealing with processing of paint, food, pharmaceuticals and specialty chemicals, that batch process is normally preferred over the continuous process due to ease of operation and flexibility to meet frequent changes in product specifications according to current market demand. Significant improvement in production systems has made it possible to; (i) produce wide range of products specifications (multiproduct process) and (ii) produce simultaneously different products (multipurpose process), by making use of the same facility and resources (Orçun *et al*., 2001; Balasubramanian and Grossmann, 2002). In each of the above case, scheduling appears to be one of the most important design considerations that require significant attention. One of the important aspects in scheduling batch process is the selection of inter stage transfer policy to transfer intermediates from one stage to another. Sometimes, the intermediate product produced in a batch process is not stable and must be transferred immediately to the next stage. This type of batch process observes zero wait transfer policy (Birewar and Grossmann, 1989; Jung *et al*., 1994; Ryu and Pistikopoulos, 2007).

Despite comprehensive research to solve the batch scheduling problem, there are still opportunities exist for new work to improve its design method. Determination of batch production sequence that produces minimum makespan based on specified batch process recipes is recognized as one of the most important design parameters as it helps to decide for the best scheduling design. Just by considering makespan as an independent decision parameter would permit designer to select the best sequence from the various possible batch production sequences. The literature revealed mostly the use of mathematical approaches such as MILP and MINLP in the formulations for determining minimum makespan and the corresponding optimal batch production sequence (Voudouris and Grossmann, 1992; Kim *et al*., 1996; Caraffa *et al*., 2001; Burkard *et al*., 2002). Though it is capable of providing the optimal solution but also offers mathematical complexity in formulating the batch scheduling problem. The use of traditional Gantt chart method is simple and could also generate design options that are near optimal based on makespan calculation. However, the procedures are very tedious and difficult to implement on computer programming, hence not attractive.

In this study, a new matrix approach is proposed to quickly calculate the makespan for all possible batch production sequences derived from specified batch process recipes. The batch process recipes are firstly arranged in a matrix according to a selected sequence. A series of simple procedure using simple mathematical formulae wherever applicable, is then applied to the matrix to calculate the makespan. By rearranging the row of the matrix to reflect changes in the production sequence, the makespan for other possible sequences could be calculated. In addition, the approach also determines

Corresponding Author: Amir Shafeeq, Department of Chemical Engineering, Universiti Teknologi Petronas, 31750 Tronoh, Perak Darul Ridzuan, Malaysia Tel: +60124069384 Fax: +6053654090

possible idle time existence in the production sequence which could offer improvement opportunity. Overall, the approach produces the result on makespan calculation for all possible production sequences from which selection could be made by designer according to the process conditions.

MAKESPAN DETERMINATION USING GANTT CHART METHOD

Calculation of makespan requires data particularly on the number of products to be produced and its corresponding batch process recipes containing the number of process stages and the processing time for each stage. Based on a selected production sequence, the data could be used to draw a Gantt chart before the makespan could be determined. The procedure is repeated for other possible production sequences to determine their corresponding makespan. In addition, the presence of idle time in each sequence could also be determined.

Example 1 and 2 demonstrated below is based on a batch process where each product follows the same sequence of operations on all process stages. The batch process consist of three main unit operations (stages) i.e., mixing, reaction and separation stage. The cleanup and transfer times are assumed to be negligible (Biegler *et al*., 1997).

Example 1. (Makespan of two products): Table 1 shows the batch process recipes arranged accordingly for a production sequence producing two products namely A followed by B. From the Gantt chart as shown in Fig. 1, it is observed that there are two locations where idle time are readily available within the process i.e., at the end of stage 1 and stage 3 (shaded area), while there are none for stage 2. Table 2 shows the summary of the idle time results.

Careful examination on the Gantt chart reveals calculation of makespan using three different approaches. One approach is to take the sum AS_1, AS_2, BS_2 and BS_3 which does not require calculation of the idle time. However, the other two approaches i.e., either taking the sum of AS_1, BS_1, BS_2, BS_3 or the sum of AS_1, AS_2, AS_3, BS_3, need to account for the idle time calculation prior to determining the makespan. As expected, the makespan calculated by all three approaches yielded the same answer i.e., 45 h.

For the purpose of verifying the observation, the batch process sequence is switched over as shown in Table 3. The inter stage idle times calculated using the Gantt chart method is shown in Table 4. The Gantt chart for the process producing product B first followed by product A is shown in Fig. 2. It could easily be observed that the process has idle time located at end of stage 1 and stage 3, while there are none for stage 2. This shows the same situation from Fig. 1 although idle times between process stages are different. The calculation of the makespan is performed using the same three approaches and again it is found to produce the same answer i.e., 45.

Example 2. (Makespan of three products): In this example, the makespan calculation is performed for batch process

Table 1: Processing time of two products in three stages for production sequence AB

Products	Processing time (h) S_1	S_2	S_3
A	10	20	5
B	8	12	3

Table 2: Idle time between three stages of two products from Table 1

Products	Idle time (h) S_1	S_2	S_3
A and B	12	0	7

Table 3: Processing time of two products in three stages for production sequence BA

Products	Processing time (h) S_1	S_2	S_3
B	8	12	3
A	10	20	5

Table 4: Idle time between three stages of two products from Table 3

Products	Idle time (h) S_1	S_2	S_3
B and A	2	0	17

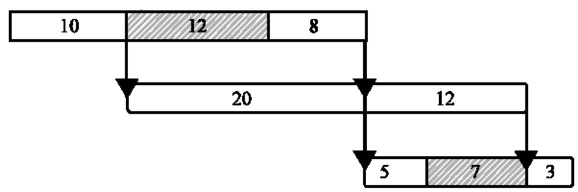

Fig. 1: Gantt chart for two products in three stages for production sequence AB

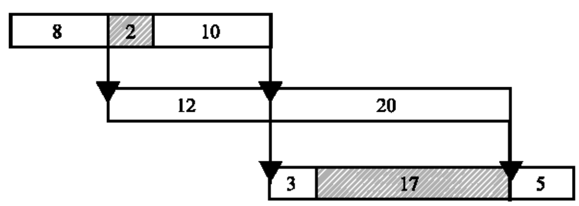

Fig. 2: Gantt chart for two products in three stages for production sequence BA

producing three products namely A, B and C. The batch process recipes based on 3 stage operation i.e., S_1, S_2 and S_3, are shown in Table 5.

From the constructed Gantt chart as shown in Fig. 3, the idle time is found to be located at the end of stage 1 and 3 for both products. The detail result is summarized in Table 6. The value of makespan is calculated from the Gantt chart as earlier done in example 1 above and the result is 50 h.

The procedure is again repeated for different batch sequence as shown in Table 7. The results for the presence of the idle time within the process as observed in the Gantt chart shown in Fig. 4 is summarized in Table 8. The calculated makespan is found to be 48 h.

From the observations made in the above examples, it can be concluded that the makespan and the idle time period and locations, varies with different batch process sequences. Although the procedure for determining both the makespan and the idle time using Gantt chart method appears to be relatively simple, it is expected to become extremely tedious as the problem size grows. This is in view of the number of repetitive construction of the Gantt chart needed to calculate makespan for all possible sequences of all products before coming up with the result of production sequence with minimum makespan.

THE PROPOSED MATRIX APPROACH

While the Gantt chart method looks relatively simple and capable of generating near optimal design options, it is actually tedious to conduct particularly for large number of products. On the other hand, most previous research conducted on batch scheduling design focused on the use of complex mathematical equations namely the mixed integer linear or non-linear programming. Although the approach is highly efficient in its execution, formulation of the batch scheduling design problem could be a problem to most designers as it requires considerable understanding in the use of high level mathematics.

The proposed matrix approach is a significant addition to our earlier study (Shafeeq *et al*., 2007) on the determination of makespan for zero wait batch processes. This work is more on developing some new algorithms for determining inter-stage idle times between every two stages of a zero wait batch process.

For the ease of understanding the matrix approach, details of the approach are described below using an example of a 3-product (A, B, C) batch production process with each, requiring three processing stages (S_1, S_2, S_3). The duration of each stage is normally different for different product, subject to the batch process recipes. The transfer policy assumed is zero wait transfer i.e., intermediates can neither wait in the same stage nor there is any storage tank available to store intermediates.

Table 5: Processing time of three products in three stages for production sequence ABC

	Processing time (h)		
Products	S_1	S_2	S_3
A	10	20	5
B	8	12	3
C	5	6	2

Table 6: Idle time between three stages of three products from Table 5

	Idle time (h)		
Products	S_1	S_2	S_3
A and B	12	0	7
B and C	7	0	3

Table 7: Processing time of three products in three stages for production sequence BAC

	Processing time (h)		
Products	S_1	S_2	S_3
B	8	12	3
A	10	20	5
C	5	6	2

Table 8: Idle time between three stages of three products from Table 7

	Idle time (h)		
Products	S_1	S_2	S_3
B and A	2	0	17
A and C	15	0	1

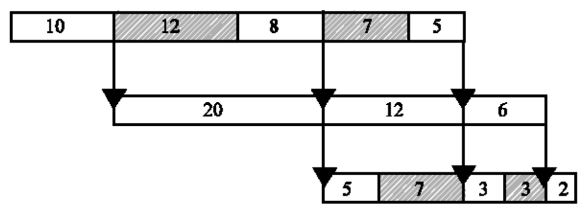

Fig. 3: Gantt chart for three products in three stages for production sequence ABC

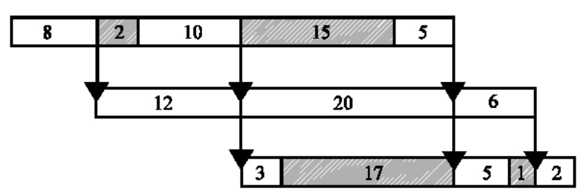

Fig. 4: Gantt chart for three products in three stages for production sequence BAC

The execution of the matrix approach follows the sequence described below.

Step 1: Arrange the batch process recipes in a matrix that represent the production sequence according to the arrangement below. In this case the scheduling is based on a production sequence where product A is produced first, followed by product B and lastly product C.

$$\begin{array}{cccc} & 0 & 1 & 2 \\ 0 & AS_1 & AS_2 & AS_3 \\ 1 & BS_1 & BS_2 & BS_3 \\ 2 & CS_1 & CS_2 & CS_3 \end{array}$$

Step 2: Introduce slack variables in between the rows of the matrix. In the matrix shown, there are six slack variables introduced i.e., V_{AB1} located in between AS_1 and BS_1, V_{AB2} located in between AS_2 and BS_2, V_{AB3} located in between AS_3 and BS_3, V_{BC1} located in between BS_1 and CS_1, V_{BC2} located in between BS_2 and CS_2 and V_{BC3} located in between BS_3 and CS_3.

$$\begin{array}{cccc} & 0 & 1 & 2 \\ 0 & AS_1 & AS_2 & AS_3 \\ & V_{AB1} & V_{AB2} & V_{AB3} \\ 1 & BS_1 & BS_2 & BS_3 \\ & V_{BC1} & V_{BC2} & V_{BC3} \\ 2 & CS_1 & CS_2 & CS_3 \end{array}$$

Step 3: Calculate the value of each slack variable by performing the following procedures to the elements of the two rows located above and below the slack variables. Firstly, comparisons are made on the value of the matrix elements between the two rows diagonally i.e., BS_1 and AS_2 and BS_2 and AS_3. For each comparison made, note on the element that has the larger value. Suppose the comparison outcome shows BS_1 has a larger value than AS_2 and BS_2 has a larger value than AS_3 as indicated by the direction of the two arrows in the matrix below.

$$\begin{array}{cccc} & 0 & 1 & 2 \\ 0 & AS_1 & AS_2 & AS_3 \\ & V_{AB1} & V_{AB2} & V_{AB3} \\ 1 & BS_1 & BS_2 & BS_3 \\ & V_{BC1} & V_{BC2} & V_{BC3} \\ 2 & CS_1 & CS_2 & CS_3 \end{array}$$

Secondly, another comparison is made between the sum of BS_1 and BS_2 and the sum of AS_2 and AS_3 Now, suppose this comparison outcome shows the earlier sum is greater than the later one. For the two combined comparison outcomes above, a specific formula has been developed for determining the value of the slack variable and this is shown in Table 9.

For the purpose of generalizing the matrix approach, the standard matrix notation M is used to represent the

Table 9: Calculation of slack variable

Cases		Slack Variables	Condition
A. $(M_{i,0}+M_{i,1})>(M_{i-1,1}+M_{i-1,2})$		$V_{i-1,0} = M_{i-1,1} - M_{i,0}$	$M_{i-1,1}>M_{i,0}$
	1.	$V_{i-1,1} = 0$	$M_{i-1,2}<M_{i,1}$
		$V_{i-1,2} = M_{i,1} - M_{i-1,2}$	
		$V_{i-1,0} = 0$	$M_{i-1,1} = M_{i,0}$
	2.	$V_{i-1,1} = 0$	$M_{i-1,2}<M_{i,1}$
		$V_{i-1,2} = M_{i,1} - M_{i-1,2}$	
		$V_{i-1,0} = 0$	$M_{i-1,1}<M_{i,0}$ or
	3.	$V_{i-1,1} = M_{i,0} - M_{i-1,1}$	$M_{i-1,2}<M_{i,1}$
		$V_{i-1,2} = (M_{i,0}+M_{i-1})-$	$M_{i-1,1}<M_{i,0}$ or
		$(M_{i-1,1}+M_{i-1,2})$	$M_{i-1,2} = M_{i,1}$
			$M_{i-1,1}<M_{i,0}$
			$M_{i-1,2}>M_{i,1}$
B. $(M_{i,0}+M_{i,1})<(M_{i-1,1}+M_{i-1,2})$		$V_{i-1,0} = M_{i-1,1} - M_{i,0}$	$M_{i-1,1}>M_{i,0}$
	1.	$V_{i-1,1} = 0$	$M_{i-1,2}<M_{i,1}$
		$V_{i-1,2} = M_{i,1} - M_{i-1,2}$	
		$V_{i-1,0} = M_{i-1,1} - M_{i,0}$	$M_{i-1,1}>M_{i,0}$
	2.	$V_{i-1,1} = 0$	$M_{i-1,2} = M_{i,1}$
		$V_{i-1,2} = 0$	
		$V_{i-1,0} = (M_{i-1,1}+M_{i-1,2})-$	$M_{i-1,1}<M_{i,0}$ or
		$(M_{i,0}+M_{i,1})$	
	3.	$V_{i-1,1} = M_{i-1,2} - M_{i,1}$	$M_{i-1,2}>M_{i,1}$
		$V_{i-1,2} = 0$	$M_{i-1,1} = M_{i,0}$ or
			$M_{i-1,2}>M_{i,1}$
			$M_{i-1,1}>M_{i,0}$
			$M_{i-1,2}>M_{i,1}$
C. $(M_{i,0}+M_{i,1}) = (M_{i-1,1}+M_{i-1,2})$		$V_{i-1,0} = 0$	$M_{i-1,1} = M_{i,0}$
	1.	$V_{i-1,1} = 0$	$M_{i-1,2} = M_{i,1}$
		$V_{i-1,2} = 0$	
		$V_{i-1,0} = M_{i-1,1} - M_{i,0}$	$M_{i-1,1}>M_{i,0}$
	2.	$V_{i-1,1} = 0$	$M_{i-1,2}<M_{i,1}$
		$V_{i-1,2} = M_{i,1} - M_{i-1,2}$	
		$V_{i-1,0} = 0$	$M_{i-1,1}<M_{i,0}$
	3.	$V_{i-1,1} = M_{i,0} - M_{i-1,1}$	$M_{i-1,2}>M_{i,1}$
		$V_{i-1,2} = 0$	

batch process recipes according to a selected production sequence. $M_{i,j}$ is used to represent the duration of the process in stage j for product i and $V_{i,j}$ is used to represent the corresponding slack variables introduced in between the rows of matrix M. The subscripts i and j stand for rows and columns respectively with initial value starting from zero.

For the matrix presented earlier, the developed formulae for calculating the slack variables V_{AB1}, V_{AB2} and V_{AB3} are;

$$V_{AB1} = \text{zero}$$
$$V_{AB2} = BS_1\text{-}AS_2$$
$$V_{AB3} = (BS_1+BS_2)\text{-}(AS_2+ AS_3)$$

where, $V_{AB1} = V_{0,0}$, $V_{AB2} = V_{0,1}$, $V_{AB3} = V_{0,2}$, $AS_2 = M_{0,1}$, $AS_3 = M_{0,2}$, $BS_1 = M_{1,0}$, $BS_2 = M_{1,1}$ for i = 1 in Table 9.

The same procedure is then repeated for the second and third row of the matrix element in order to calculate the next set of slack variables, V_{BC1}, V_{BC2} and V_{BC3}. Suppose BS_2 has a larger value than CS_1 and CS_2 has a larger value than BS_3 as indicated by the direction of the two arrows in the matrix below.

$$
\begin{array}{cccc}
 & 0 & 1 & 2 \\
0 & AS_1 & AS_2 & AS_3 \\
 & V_{AB1} & V_{AB2} & V_{AB3} \\
1 & BS_1 & BS_2 & BS_3 \\
 & V_{BC1} & V_{BC2} & V_{BC3} \\
2 & CS_1 & CS_2 & CS_3
\end{array}
$$

Now, suppose the next comparison outcome between the sum of CS_1 and CS_2 and the sum of BS_2 and BS_3 shows that the earlier sum is greater than the later, the formulae used now for calculating V_{BC1}, V_{BC2} and V_{BC3} as shown in Table 9 are:

$$V_{BC1} = BS_2 - CS_1$$
$$V_{BC2} = \text{Zero}$$
$$V_{BC3} = CS_2 - BS_3$$

where, $V_{BC1} = V_{1,0}$, $V_{BC2} = V_{1,1}$, $V_{BC3} = V_{1,2}$, $BS_2 = M_{1,1}$, $BS_3 = M_{1,2}$, $CS_1 = M_{2,0}$, $CS_2 = M_{2,1}$ for i = 2 in Table 9

Step 4: From the calculated values for the slack variables above, the makespan for the multi product batch process can be calculated using one of the following formulae;

- Makespan = $AS_1+AS_2+AS_3+V_{AB3}+BS_3+V_{BC3}+CS_3$
- Makespan = $AS_1+V_{AB1}+BS_1+V_{BC1}+CS_1+CS_2+CS_3$
- Makespan = $AS_1+AS_2+V_{AB2}+BS_2+V_{BC2}+CS_2+CS_3$

Regardless of the formula selected, the calculated makespan produced the same answer.

GENERALIZED MATHEMATICAL EXPRESSION FOR N NUMBER OF PRODUCTS IN MATRIX APPROACH

On the basis of the observations made from the Gantt chart method, a set of generalized mathematical expressions were developed to determine the values of the slack variables and the makespan of the batch process with n number of products involving 3 process stages. The calculation procedures begin by making the required comparisons as stated in step 3 for the elements in the first and second row of the matrix before selecting the right formulae for determining the corresponding slack variables. This is then followed by adopting the same procedures for the elements in the second and third row of the matrix. The procedures are repeated until the elements in the last two rows have been addressed. Generally, the comparison outcomes as stipulated in step 3 for the matrix elements in each set of two rows addressed i.e., $(M_{i,0} + M_{i,1})$ and $(M_{i-1,1} + M_{i-1,2})$ where (i = 1,2,……n-1), could be categorized under three different conditions termed as Case A, Case B and Case C. For each case, three different set of formulae are available from which one is to be selected for the slack variables calculation. The guide for selecting the right formula is based on the condition as illustrated in Table 9.

Using the calculated value obtained for all the slack variables, the makespan is calculated using one of the following generalized mathematical expressions developed for n number of products with three batch process stages.

$$\text{Makespan} = \sum_{i=0}^{n-1} M_{i,0} + \sum_{j=1}^{2} M_{n-1,j} + \sum_{i=0}^{n-2} V_{i,0}$$

or

$$\text{Makespan} = \sum_{j=0}^{2} M_{0,j} + \sum_{i=1}^{n-1} M_{i,2} + \sum_{i=0}^{n-2} V_{i,2}$$

or

$$\text{Makespan} = M_{0,0} + \sum_{i=0}^{n-1} M_{i,1} + \sum_{i=0}^{n-2} V_{i,1} + M_{n-1,2}$$

The application of the matrix approach on example 2 for which earlier Gantt chart method applied, is shown below to determine the makespan.

$$
\begin{array}{cccc}
 & 0 & 1 & 2 \\
0 & 10 & 20 & 5 \\
 & V_{0,0} & V_{0,1} & V_{0,2} \\
1 & 8 & 12 & 3 \\
 & V_{1,0} & V_{1,1} & V_{1,2} \\
2 & 5 & 6 & 2
\end{array}
$$

Comparisons are made on the matrix elements located in the first two rows and this resulted in the following conditions;

$$\left(M_{1,0} + M_{1,1}\right) < \left(M_{0,1} + M_{0,2}\right) \text{ and } \begin{array}{l} M_{0,1} > M_{1,0} \\ M_{0,2} < M_{1,1} \end{array}$$

i.e., (8+12) is less than (20+5) and 20 is greater than 8 while 5 is lesser than 12.

This satisfies the condition under Case B1 (Table 9) and therefore, the slack variables values are calculated using the formulae;

$$V_{0,0} = M_{0,1} - M_{1,0} = (20-8) = 12$$
$$V_{0,1} = 0$$
$$V_{0,2} = M_{1,1} - M_{0,2} = (12-5) = 7$$

The same procedures are then repeated for the elements in the second and third rows of the matrix. The comparisons resulted in the following conditions.

$$\left(M_{2,0} + M_{2,1}\right) < \left(M_{1,1} + M_{1,2}\right) \text{ and } \begin{array}{l} M_{1,1} > M_{2,0} \\ M_{1,2} < M_{2,1} \end{array}$$

i.e., (5+6) is lesser than (12+3) and 12 is greater than 5 while 3 is lesser than 6.

The comparison outcome is referred to Table 9 and it is found that it satisfies Case B1. Therefore, the slack variables are calculated using the formulae;

$$V_{1,0} = M_{1,1} - M_{2,0} = (12 - 5) = 7$$
$$V_{1,1} = 0$$
$$V_{1,2} = M_{2,1} - M_{1,2} = (6 - 3) = 3$$

Using the values of the slack variables and the required elements from the batch process recipes matrix, the makespan for the specified production sequence is calculated using one of the following formulae;

$$\text{Makespan} = \sum_{i=0}^{2} M_{i,0} + \sum_{j=1}^{2} M_{2,j} + \sum_{i=0}^{1} V_{i,0}$$
$$= (10 + 8 + 5) + (6 + 2) + (12 + 7) = 50$$

or

$$\text{Makespan} = \sum_{j=0}^{2} M_{0,j} + \sum_{i=1}^{2} M_{i,2} + \sum_{i=0}^{1} V_{i,2}$$
$$= (10 + 20 + 5) + (3 + 2) + (7 + 3) = 50$$

or

$$\text{Makespan} = M_{0,0} + \sum_{i=0}^{2} M_{i,1} + \sum_{i=0}^{1} V_{i,1} + M_{2,2}$$
$$= 10 + (20 + 12 + 6) + (0 + 0) + 2 = 50$$

It is worth noting that the makespan value obtained above is for the production sequence where A is produced first, followed by B and lastly C.

Determination of optimum batch production sequence using matrix approach: The optimal batch process schedule is often based on designer's choice for production sequence offering minimum makespan. A repetition on the whole procedures in the matrix approach above to address different production sequence will enable it to calculate the makespan for the corresponding sequence. Repeating it for all possible production sequences will lead to the makespan for each of the possible sequence to be determined. The possible number of production sequences could be determined using the simple permutation rule shown below:

$$P(n) = n!$$

Where:

P(n) = No. of possible batch production sequences
n = No. of products

For example, the number of possible production sequences for three products namely A, B and C is P(3) = 3! = 6. Based on the permutation result, the production sequences can then be derived, e.g., ABC, BAC, BCA etc., as shown in Table 10. For all the possible production sequences for the given batch process recipes as shown in example 2 (Table 5), the calculated makespan and inter-stage idle times using the matrix approach are shown in Table 10. The results obtained show the production sequence BAC produces the minimum makespan of 48 h. Apart from that, it is noticed that the inter-stage idle time varies with different production sequences.

Table 10: Makespan and Interstage Idle times for all possible production sequences of three products A, B and C in three stages

Production sequence	Makespan (h)	Inter-stage Idle times (h) V_{AB1}	V_{AB2}	V_{AB3}	V_{BC1}	V_{BC2}	V_{BC3}
ABC	50	12	0	7	7	0	3
BAC	48	2	0	17	15	0	1
BCA	55	7	0	3	0	4	22
CBA	50	0	2	12	2	0	17
CAB	50	0	4	22	12	0	7
ACB	53	15	0	1	0	2	12

Table 11: Processing time of four products in three stages

Products	Processing time (h) S_1	S_2	S_3
A	11	19	5
B	14	8	10
C	21	7	8
D	4	6	5

To show the efficiency and robustness of the matrix approach, a much larger problem with four products is solved as illustrated in example 3 below for calculation of minimum makespan and interstage idle times. It is obvious that the number of permutations will become prohibitively large as shown in Table 12. The processing times data for four products A, B, C and D in three stages is given in Table 11.

Example 3:

$$\begin{array}{cccc}
 & 0 & 1 & 2 \\
0 & 11 & 19 & 5 \\
 & V_{AB1} & V_{AB2} & V_{AB3} \\
1 & 14 & 8 & 10 \\
 & V_{BC1} & V_{BC2} & V_{BC3} \\
2 & 21 & 7 & 8 \\
 & V_{CD1} & V_{CD2} & V_{CD3} \\
3 & 4 & 6 & 5
\end{array}$$

Table 12 shows the makespan calculation for every possible sequence of four products with respective idle times between process stages using matrix approach. The matrix approach is applied using a computer program developed for this purpose in Microsoft Visual C++

Table 12: Makespan and Idle times for all possible sequences of four products A, B, C and D in three stages

Production sequence	Makespan (h)	Inter-stage Idle times (h)								
		V_{AB1}	V_{AB2}	V_{AB3}	V_{BC1}	V_{BC2}	V_{BC3}	V_{CD1}	V_{CD2}	V_{CD3}
ABCD	71	5	0	3	0	13	10	5	2	0
ABDC	78	5	0	3	8	4	0	0	15	17
ADBC	80	15	0	1	0	8	11	0	13	10
ADCB	83	15	0	1	0	15	17	0	7	7
ACDB	73	0	2	4	5	2	0	0	8	11
ACBD	69	0	2	4	0	7	7	8	4	0
BCAD	76	0	13	10	0	4	15	15	0	1
BCDA	79	0	13	10	5	2	0	0	5	19
BDCA	82	8	4	0	0	15	17	0	4	15
BDAC	73	8	4	0	0	5	19	0	2	4
BADC	80	0	3	12	15	0	1	0	15	17
BACD	66	0	3	12	0	2	4	5	2	0
CABD	74	0	4	15	5	0	3	8	4	0
CADB	83	0	4	15	15	0	1	0	8	11
CDAB	78	5	2	0	0	5	19	5	0	3
CDBA	79	5	2	0	0	8	11	0	3	12
CBDA	82	0	7	7	8	4	0	0	5	19
CBAD	76	0	7	7	0	3	12	15	0	1
DBAC	65	0	8	11	0	3	12	0	2	4
DBCA	74	0	8	11	0	13	10	0	4	15
DCBA	74	0	15	17	0	7	7	0	3	12
DCAB	73	0	15	17	0	4	15	5	0	3
DACB	68	0	5	19	0	2	4	0	7	7
DABC	70	0	5	19	5	0	3	0	13	10

(Ver. 6.0)™ for faster execution. It is observed from Table 12 that production sequence DBAC offers minimum makespan of 65 h.

CONCLUSIONS

The study presented a new approach based on matrix for determining the makespan for a specified production sequence of a batch process using zero wait transfer policy. The approach is considerably easier to understand and does not require the knowledge of advanced mathematics. In addition, the procedures could be implemented easily on computer programming to speed up the makespan calculation, which is particularly required when determining the optimum production sequence from a given batch process recipes. Once the approach is programmed on computer, the user is only required to provide the batch process recipes and all the possible sequences with its corresponding makespan and inter stage idle time will be automatically determined. The approach is tested against the established mathematical programming approach and a similar optimum solution is obtained. In addition to determining the optimum solution, the approach could also produce other near optimal solutions from which designer could have flexibility to choose the one that meets optimum conditions. Such opportunity could lead towards the development of a more interactive design for batch process scheduling.

REFERENCES

Balasubramanian, J. and I.E. Grossmann, 2002. A novel branch and bound algorithm for scheduling flowshop plants with uncertain processing times. Comput. Chem. Eng., 26 (1): 41-57.

Biegler, L.T., I.E. Grossmann and A.W. Westberg, 1997. Systematic Methods of Chemical Process Design. Prentice-Hall.

Birewar, D.B. and I.E. Grossmann, 1989. Efficient optimization algorithms for zero-wait scheduling of multiproduct batch plants. Ind. Eng. Chem. Res., 28 (9): 1333-1345.

Burkard, R.E., T.C. Fortuna and A.J. Hurkens, 2002. Makespan minimization for chemical batch processes using non-uniform time grids. Comput. Chem. Eng., 26 (9): 1321-1332.

Caraffa, V., S. Ianes, T.P. Bagchi and C. Sriskandarajah, 2001. Minimizing makespan in a blocking flowshop using genetic algorithms. Int. J. Prod. Econ., 70 (2): 101-115.

Jung, J.H., H.K. Lee, D.R. Yang and I.B. Lee, 1994. Completion times and optimal scheduling for serial multi-product processes with transfer and set-up times in zero-wait policy. Comput. Chem. Eng., 18 (6): 537-543.

Kim, M., J.H. Jung and I.B. Lee, 1996. Optimal scheduling of multiproduct batch processes for various intermediate storage policies. Ind. Eng. Chem. Res., 35 (11): 4058-4066.

Orçun, S., I.K. Altinel and Ö. Hortaçsu, 2001. General continuous time models for production planning and scheduling of batch processing plants: Mixed integer linear program formulations and computational Issues. Comput. Chem. Eng., 25 (2-3): 371-389.

Ryu, J.H. and E.N. Pistikopoulos, 2007. A novel approach to scheduling of zero-wait batch processes under processing time variations. Comput. Chem. Eng., 31 (3): 101-106.

Shafeeq, A., M.I. Abdul Mutalib, K.A. Amminudin and A. Muhammad, 2007. Determination of makespan for zero wait batch processes: A novel method using matrix. In Conference Proceedings of International Multi Conference of Engineers and Computer Scientists, Hongkong.

Voudouris, V.T. and I.E. Grossmann, 1992. Mixed-integer linear programming reformulations for batch process design with discrete equipment sizes. Ind. Eng. Chem. Res., 31 (5): 1315-1325.