APEM journal

Advances in Production Engineering & Management
Volume 14 | Number 2 | June 2019 | pp 166–176
https://doi.org/10.14743/apem2019.2.319

ISSN 1854-6250
Journal home: apem-journal.org
Original scientific paper

Simulation framework for determining the order and size of the product batches in the flow shop: A case study

Ištoković, D.[a,*], **Perinić, M.**[a], **Doboviček, S.**[a], **Bazina, T.**[a]

[a]University of Rijeka, Faculty of Engineering, Rijeka, Croatia

A B S T R A C T

The problems of determining the order and size of the product batches in the flow shop with multiple processors (FSMP) and sequence-dependent setup times are among the most difficult manufacturing planning tasks. In today's environment, where necessity for survival in the market is to deliver the goods in time, it is crucial to optimize production plans. Inspired by real sector manufacturing system, this paper demonstrates the discrete event simulation (DES) supported by the genetic algorithm (GA) optimization tool. The main aim is to develop the simulation framework as a support for the daily planning of manufacturing with emphasis on determining the size and entry order of the product batches within specific requirements. Procedures are developed within the genetic algorithm, which are implemented in Tecnomatix Plant Simulation software package. A genetic algorithm was used to optimize mean flow time (MFT) and total setup time (TST) performance measures. Primary constraint for on-time delivery was imposed on the model. The research results show that solutions are industrially applicable and provide accurate information on the batch size of the defined products, as well as a detailed schedule and timing of entry into the observed system. Display of the solution, in a simple and concise manner, serves as a tool for manufacturing operations planning.

A R T I C L E I N F O

Keywords:
Process planning;
Flow shop;
Sequence-dependent setup time;
Batch scheduling;
Discrete event simulation;
Genetic algorithm (GA)

**Corresponding author:*
distokovic@riteh.hr
(Ištoković, D.)

Article history:
Received 25 February 2019
Revised 18 April 2019
Accepted 7 May 2019

1. Introduction

One of the key problems in operational planning of manufacturing is to determine the order and size of a product batches. In today's environment, the solution to this problem is a necessity for survival in the market. Manufacturing companies must deliver the goods in time to avoid losses and ensure competitiveness where activities are planned to effectively use the available resources [1].

This paper explores the problem of determining the order and size of a product batches with the aim of developing the simulation framework as support for the daily planning of manufacturing. The study deals with the problem of the real sector.

FSMP planning includes, among other things, the order of jobs in unidirectional flow system where at any workstation more than one machine of the same type can be located. The machine can process at most one job at any given time. All jobs are subject to priorities that limit them to the same processing order throughout all processing stages [2]. Each customer order must be processed in all or some of the workstations. Special attention is paid to the presence of the sequence-dependent setup times, the size of the product batches and the availability of production resources [3]. Setup time is the time required for the staff to prepare and provide a job for uninterrupted work when changing the operation. It does not create any additional value, and by increasing the number of setups, available machine processing time is consequently reduced. Reducing the available processing time increases the possibility of delays with delivery, which every company wants to avoid.

This paper presents a discrete model of the manufacturing system, made using the simulation package Tecnomatix Plant Simulation. Using the genetic algorithm implemented within the simulation package itself, the problem of determining the order and size of the product batches is solved. This simulation shows the exact values of the required parameters that are applicable to any system of that type.

The rest of the article is organized as follows. A review of relevant literature is presented in the section 2. Section 3 introduces the formulation of the observed problem, as well as an overview of the development of the simulation model. Section 4 shows the details of the simulation experiments. Section 5 provides an analysis of experimental results. Finally, section 6 gives final notes.

2. Literature review

Discrete event simulations (DES) are powerful and effective tools for solving many real-world problems. They are also one of the most commonly used techniques for analysing and understanding the dynamics of manufacturing systems. The proof of this is a large number of published scientific papers on this particular subject. Negahban and Smith provided a comprehensive review of discrete event simulation publications with a focus on applications in manufacturing, including manufacturing operations planning and scheduling problems [4]. Another comprehensive survey on scheduling problems, which provides an extensive review of about 500 papers that have appeared since the mid of 2006 to the end of 2014, was presented by Allahverdi [5]. He introduced very significant classification and notation of scheduling problems, based on shop ambient, process features, setup conditions and performance measures. I. Ribas *et al.* provided overarching review of recently published articles about the problems of scheduling hybrid flow shop (HFS). The works are divided into two categories based on the characteristics of the HFS and production constraints and in view of the proposed approach to problem solving [6]. The most important surveys of using DES for solving flow-shop specific manufacturing operations planning and scheduling problems are reviewed in this section.

Gourgand *et al.* [7] researched scheduling problems in two and *m* machine stochastic flow shop with infinite buffers. They implemented hybrid approach, consisting of recursive heuristics or metaheuristics and performance evaluation algorithm. Makespan, used for measuring performance of generated feasible job schedules, was computed using Markov chain, or estimated using DES. Wang *et al.* [8] used genetic algorithms (GA) for stochastic flow-shop scheduling problem. Their objective was to avoid premature convergence of the GA. Yang *et al.* [9] solved a multi-attribute combinatorial dispatching (MACD) decision problem in a flow shop with multiple processors (FSMP) environment. The same problem solved Azadeh *et al.* [10]. I. A. Chaudhry and M. Usman used a genetic algorithm and independent spreadsheets to simultaneously solve scheduling problems and process planning in a job shop environment [11]. R. Meolic and Z. Brezočnik proposed a new approach to solving job shop scheduling problems with an emphasis on identifying feasible solutions. The new approach allows all schedules of relatively large systems to be found using the data structure, the zero-suppressed binary decision diagrams [12].

Hendizadeh *et al.* [13] considered a flow shop scheduling problem of a manufacturing cell that contains families of jobs. Setup times are sequence-dependent of the families. To minimize makespan and total flow time, the authors proposed multi-objective genetic algorithm (MOGA). The same problem has been considered by Lin and Ying [14] using a two-level multi-start simulated annealing (TLMSA). Lee [15] dealt with a problem of estimating order lead time in hybrid flow shops, where orders arrive dynamically. Alfieri [3] proposed the solution for multiple objective flow shop scheduling problem on model with work calendars on resources, multi-machine stages, re-entrant flows, external operations, sequence-dependent setup times, and transfer product batches between stages. Dugardin *et al.* [16] presented L-NSGA multi-objective

GA which uses the Lorenz dominance relationship for a re-entrant hybrid flow shop scheduling problem. They had two objectives: maximum utilization rate of the bottleneck and minimum completion time. Ladhari *et al.* [17] researched two-machine permutation flow shop problem with the sequence-independent setup times. Their objectives were minimizing the sum of completion times. Ying *et al.* [18] examined the no-wait flow shop manufacturing cell scheduling problem (FMCSP) with the sequence-dependent family setup times with makespan criterion as objective. Galzina *et al.* [19] deal with the flow shop scheduling problem using hybrid fuzzy logic and intelligent swarm method. The compiled model was compared with stochastic algorithms for assessing applicability to general problems. Chen and Hao [20] solved the problem of distributing the flow shop by applying a non-dominated sorting genetic algorithm (NSGA). They use it for multi-objective optimization of non-compact flow shops in view of process linking. Yan *et al.* used a Tabu search algorithm and particle swarm optimization in a two-stage semi-continuous flow shop to optimize the production and distribution decisions at the same time [21].

Liu *et al.* [22] compiled an overview of different optimization approaches for solving manufacturing planning and scheduling problems. They listed numerous global and local optimization methods, along with application examples and associated constraints. Frequently used techniques, like response surface methodology, gradient-based methods and evolutionary algorithms, as well as emerging ones, like stochastic approximation, particle swarm optimization and ant colony optimization were encompassed. Supsomboon and Vajasuvimon proposed simulation model using Tecnomatix Plant Simulation for making machine parts in the job shop. The simulation model shows that job expansion, plant allocation, group technology, and capacity expansion ultimately contribute to lower operating costs and increase employee utilization [23].

3. Materials and methods

The objectives of each manufacturing are to achieve the required product quality with the least cost of manufacturing, and delivery on time [24]. Delivering on time is the primary condition which must be satisfied. This generally means that the total required quantity of products q_j must be produced within the observed period, i.e. the completion time of the last workpiece (makespan) C_{max} must be less than the maximum available time of the manufacturing equipment $C_{max,goal}$ for the observed period, Eq. 1.

$$C_{max} < C_{max,goal} \tag{1}$$

Batch size of the *j*-product Lot_j is defined as a natural number in a given interval, Eq. 2. *Dg* represents the minimum batch size, and *gg* the maximum batch size of the *j*-product that cannot be greater than the total quantity of product q_j.

$$Lot_j \in [dg, gg], \quad \forall\, Lot_j \in \mathbb{N} \tag{2}$$

Taking any number from the defined range as the batch size enables a lot more potential optimization solutions than [25] the use of different batch size with the assumption that the total quantity of the product must be multiplied by the batch size (the total quantity of the product must be divided by the batch size). This means that each batch size is the same, which greatly reduces the ability to find a better solution.

If it is assumed that each batch size of the same product is equal to the total quantity of the product, it is likely that more products will be produced than needed. This ultimately does not change the mean flow time, but it extends the total processing duration. Also, this creates a stock that creates additional cost that is undesirable for the company.

Simple example: It is necessary to produce 1,234 workpieces in 2 weeks. A batch size is 400 workpieces. It is evident that if three batches of 400 workpieces are produced, there are still 34 workpieces left to produce. If four batches of 400 workpieces are produced, the stock will be 366 workpieces, which will increase the makespan, and therefore the possibility of not delivering on time. For this reason, it is ensured that the last batch of the same product is equal to the Eq. 3.

$$Lot_{j,last} = q_j - Lot\_Num_j \cdot Lot_j \tag{3}$$

where $Lot_{j,\mathrm{last}}$ is the size of the last batch of the same product, $Lot_{j,\mathrm{last}} \in [1, Lot_j]$; q_j is the total amount of the same product; Lot_Num_j is the amount of produced batches of the same product; Lot_j is the batch size of the same product.

In a flow shop production system the products travel in batches through the system. The batch size directly affects the flow time in a way that increasing the batch size linearly increases the flow time, worth and vice versa. The flow time F_j is defined as the time the j-product batch performs in the system, i.e. the difference between the j-product's output time and the j-product's input time. The flow time of each j-product batch is different because of uneven waiting times on the processing. For simpler further optimization their mean value is calculated by Eq. 4. Thus, the mean flow time MFT_j is actually the average time of all j-product flow times. By introducing the total setup time, the mean flow time does not change, but the makespan does. The bigger the total setup time, the bigger the makespan.

$$MFT_j = \frac{\sum_1^{Lot\_Num_j} F_j}{Lot\_Num_j} \tag{4}$$

In order to determine the entry sequence of product batches, a second variable is introduced -the probability of entering the j-product batches into the system $Perc_j$, defined as a natural number at a given interval, Eq. 5.

$$Perc_j \in [dg, gg], \quad \forall\, Perc_j \in \mathbb{N} \tag{5}$$

The real probability of entering the j-product batches $RealPerc_j$ is defined as ratio of the probability of entering the j-product batches and sum of the probability of entering product batches, according to Eq. 6.

$$RealPerc_j = \frac{Perc_j}{Perc_1 + Perc_2 + \cdots + Perc_j + \cdots + Perc_n} \tag{6}$$

The entry sequence of a product batches has no effect on the mean flow time, but when entering different product batch, the need for setup time appears. When one batch of a particular product is completed on the same production equipment, then a new batch of a particular product comes in. If the new product batch is the same as the previous product batch, then setup time is not required, i.e. the setup time is equal to zero. If the new product batch is different from the previous product batch, then the setup of the workplace is required before the start of processing. The setup time is randomly selected in a uniform distribution, according to [26]. At the arrival of the first batch of any product, the setup of the workplace is also carried out.

The question is why the setup time cannot be clearly displayed (if not automated, i.e. if it performs at least partially by a man)? There are many reasons, including working staff that performs the setup job is different (work in multiple shifts), fatigue and motivation of the staff members, etc.

It is not possible to analytically determine how much is the total setup time of the j-product for the observed period. It is determined by simulation. The total setup time of the j-product TST_j represents the sum of all setup times of the j-product ST_j, according to Eq. 7. The ST_Num_j represents the number of impressions of setup time ST during the production of the j-product.

$$TST_j = \sum_1^{ST\_Num_j} ST_j \tag{7}$$

It is concluded that the minimum total setup time will ideally be when the production is in a unit as large batches. Also, from the standpoint of the minimum mean flow time, it is preferred that the production takes place in the lowest unit of product batches, which leads to contradictions. For this reason, in order to simultaneously minimize both, the mean flow time and total setup time, it is necessary to conduct optimization.

Optimization will be performed using a genetic algorithm that is embedded in the used software package Tecnomatix Plant Simulation. The structure of genetic algorithm is shown in Fig. 1. Whereby *pop* means population, *gen* means generation and *gen_num* means maximum number of generations.

Based on the defined input variables (batch size Lot_j, probability of entering the batch $Perc_j$) and the simulation model of the production system, the fitness method will be minimized. The fitness method is defined as the sum of individual members where each member has a certain importance.

```
gen = 0
Generating an Initial Population pop(0)
if gen < gen_num
        gen = gen + 1
        Fitness proportionate selection pop'(gen) from pop(gen - 1)
        Crossover in two points pop'(gen) and saving in pop''(gen)
        Mutation pop''(gen)
        Probabilistic selection pop(gen) from pop''(gen) and pop(gen - 1)
minimize fitness
```

Fig. 1 Structure of genetic algorithm

Thus, each individual member is multiplied by the weight factor, with the higher weight factor being more important for the overall result. According to the above, each member for the mean flow time of j-product MFT_j would be multiplied with the weight factor a_j, and each member for the total setup time of the j-product TST_j would be multiplied with the weight factor b_j, shown in Eq. 8. Considering that the total sum of all weight factors must be equal to 1, Eq. 9.

$$fitness = min\ (a_j \cdot MFT_j + b_j \cdot TST_j) \tag{8}$$

$$a_1 + a_2 + \cdots + a_j + \cdots + a_n + b_1 + b_2 + \cdots + b_j + \cdots + b_n = 1 \tag{9}$$

4. Presentation of the problem

4.1 General

Studied production system is designed according to the production plant companies from the real sector which are producing families of technologically similar products. Technologically similar products are those that have a high degree of similarity to the order of processing and duration of the operations. It is assumed that the production of three products (D, E, F) is foreseen for delivery every two weeks, or more precisely every second Friday after the second shift at 10:00 p.m. The two-week quantity determined for the three products as well as the order and duration of the operations are assumed and shown in Table 1. The operation times are set in hours.

Table 1 Example data

j q_j	D 4616	E 3232	F 2616	M_{iC}
i				
1	10-0.028	10-0.035	10-0.036	3
2	20-0.031	20-0.035	20-0.036	3
3	30-0.012	30-0.01	30-0.01	1
4	40-0.02	40-0.019	40-0.017	2
5	50-0.08	50-0.012	50-0.14	1

The working week lasts for five days and takes place in two shifts. Operating hours per shift are eight. From this, according to the Eq. 6, the maximum availability of production equipment can be calculated C_{max} is 160 hours. The machines cannot operate continuously, without interruption, so the utilization time is 0.85, according to [27]. The required number of i-th production equipment M_{iC} has been obtained by [28], assuming that the reliability of production equipment equals 1. Production takes place at five workstations, where all three products pass unidirectional through the system and are processed at each workstation. Each workstation consists of the M_{iC} number of the same production equipment. All of the above defines the observed production system as a flow shop with multiple processors (FSMP). Using the Tecnomatix Plant Simulation software package, a discrete FSMP model was developed.

The setup time is defined as relatively large due to the fact that in a serial production, except change of tools, jig, etc., it is extremely important to check the first workpiece of the batch. This is sometimes a request by clients in some industries (e.g., automotive industry). Therefore, the setup time ST_j for workstations 1 and 2 is defined by a uniform distribution between 10 and 18 minutes, while for workstations 3, 4 and 5 is defined by a uniform distribution between 12 and 20 minutes. Bearing in mind that these are technologically similar products, the setup times for any combination of the previous and next batch of j-products are approximately equal ($D \rightarrow E$, $E \rightarrow F$, etc.).

The boundary conditions for batch size Lot_j and probability of entering the j-product $Perc_j$ were given by Eq. 11 and Eq. 12.

$$Lot_j \in [10{,}1000] \quad \forall\, Lot_j \in \mathbb{N} \tag{11}$$

$$Perc_j \in [100{,}1000] \quad \forall\, Perc_j \in \mathbb{N} \tag{12}$$

4.2 Genetic algorithms parameters

The values of genetic operators used were as follows: *probability of crossover* and *probability of mutation* were 0.8 and 0.15, respectively. When triggering optimization, the following limitations for the genetic algorithm were determined: number of population *pop_num* = 50, number of generations *gen_num* = 250, and number of observation *obs_num* = 10.

4.3 Coding of organisms

The genetic algorithm at the beginning of the optimization randomly generates an initial population of 50 individuals. Each individual (chromosome) consists of six genes that represent a specific property: first gene is batch size of the product D, second gene is batch size of the product E, third gene is batch size of the product F, fourth gene is probability of entering into system batch of the product D, fifth gene is probability of entering into system batch of the product E, sixth gene is probability of entering into system batch of the product F.

4.4 Definition of fitness function

The observed optimization task is the assignment task. Therefore, to solve this problem, a given gene assigns a random value:

- according to Eq. 11, for gene of batch size,
- according to Eq. 12, for the probability of entering a certain batch of the product.

When initial individuals with corresponding values (genes) are defined, the fitness method for all individuals within the population is calculated using Eq. 13:

$$\begin{aligned} fitness = C_{max,fit} + a_D \cdot MFT_D + a_E \cdot MFT_E + a_F \cdot MFT_F + b_D \cdot TST_D \\ + b_E \cdot TST_E + b_F \cdot TST_F \end{aligned} \tag{13}$$

Earlier defined objectives are that all the mean flow times are as small as possible, as well as all total setup times. Thus, the task of the genetic algorithm is to find the least value (minimum) of the fitness method. Furthermore, as all the optimization parameters are equally important, the assumption is that all weight factors are equal, i.e.:

$$a_D = a_E = a_F = b_D = b_E = b_F = \frac{1}{6}$$

Delivering on time is the primary condition which must be satisfied, as such, it should be a part of the objective function. However, it is not necessary for the makespan to be as small as possible, only to be satisfied. Therefore, the makespan will not be part of the objective function. In order for the genetic algorithm to "move away" from poor (unsatisfactory) solutions and "approach" better solutions, a penalty $C_{max,penalty}$ should be introduced according to Fig. 2. Value of $C_{max,fit}$ is added to the objective function. If $C_{max,fit} = 0$, the delivery condition is satisfied and will not have any effect on the objective function, but if $C_{max,fit} > 0$ then this means that the condition is not satisfied and that the value of the goal function will increase, which will ultimately result in moving genetic algorithm from bad solutions. In this way, it is achieved that products are made on time and that delivery is not delayed.

```
var Cmax,penalty : real : = [ (Cmax - Cmax,goal) / Cmax,goal ]
var Cmax,fit : time
if  Cmax,penalty  >  0
    Cmax,fit : = ( 1 + Cmax,penalty ) · ( Cmax - Cmax,goal )
else
    Cmax,fit : = 0
end
```

Fig. 2 Penalty condition

The objective function was calculated in the example of two selected individuals.

	Lot_D	Lot_E	Lot_F	$Perc_D$	$Perc_E$	$Perc_F$
Individual 1	255	703	444	236	199	704
Individual 2	275	325	335	770	400	350

By simulation, the mean flow time and the total setup time are obtained. The time format used in the following text is days:hours:minutes:seconds (d:h:m:s).

	MFT_D	MFT_E	MFT_F	TST_D	TST_E	TST_F	C_{max}
Individual 1	2:11:38:44	2:01:57:41	1:15:18:29	2:04:22:00	1:19:08:15	3:23:06:42	23:02:46:56
Individual 2	0:23:49:18	1:00:46:18	1:01:28:57	1:19:53:49	1:02:33:06	0:22:43:36	5:22:33:50

Using penalty condition (Fig. 2) the penalties for each individual are obtained:

Individual 1
$C_{max,penalty}$ = [(23:02:46:56 - 6:16:00:00) / 6:16:00:00] = 2.467 > 0
$C_{max,fit}$ = (1 + 2.467) · (23:02:46:56 - 6:16:00:00) = 40:14:04:53

Individual 2
$C_{max,penalty}$ = [(5:22:33:50 - 6:16:00:00) / 6:16:00:00] = -0.122 ≤ 0
$C_{max,fit}$ = 0

Furthermore, in Eq. 13 the objective function is calculated for the two individuals mentioned.

Individual 1

$$fitness = 40:14:04:53 + \frac{1}{6}\cdot 2:11:38:44 + \frac{1}{6}\cdot 2:01:57:41 + \frac{1}{6}\cdot 1:15:18:29 + \frac{1}{6}\cdot 2:04:22:00 + \frac{1}{6}\cdot 1:19:08:15 + \frac{1}{6}\cdot 3:23:06:42 = \mathbf{42:22:40:11}$$

Individual 2

$$fitness = 0 + \frac{1}{6}\cdot 23:49:18 + \frac{1}{6}\cdot 1:00:46:18 + \frac{1}{6}\cdot 1:01:28:57 + \frac{1}{6}\cdot 1:19:53:49 + \frac{1}{6}\cdot 1:02:33:06 + \frac{1}{6}\cdot 22:43:36 = \mathbf{1:03:52:31}$$

The mentioned individuals (parents) can be selected for cloning by roulette wheel selection, whereby Individual 2 being much more likely to be selected than Individual 1. Every individual can be selected more than once, but it is also possible not to be selected even once. Genetic operators (2-point crossover, mutation) are applied on cloned individuals (offsprings) [29]. Then the probabilistic method selects individuals for the next generation between parents and offsprings. The process is repeated until it reaches the 250 generation. The genetic algorithm then shows the best solutions.

5. Results and discussion

The previously defined optimization by genetic algorithm was performed on an 8-core processor of 2.66 GHz, with duration of 2:04:40:50 (d:h:m:s). The best generated solution for a defined optimization task is 1:00:55:36 (d:h:m:s), which is the minimum value of the objective function. The optimization parameters obtained are given in Table 2.

From an evolutionary diagram, Figure 3, are visible solutions obtained during a defined number of generations. Also, it can be observed that the genetic algorithm has relatively quickly found a fairly good solution, but an increased number of generations were given an even better solution.

Through optimization, the values for batch size of *j*-product are obtained. By initiating the simulation of the production system model for previously obtained optimization results, the sequence of entering of the *j*-products is determined. Fig. 4, besides the sequence of the *j*-product entries, also presents other values such as: batch size of *j*-product, finished quantity of *j*-product, completion time (makespan), mean flow time of *j*-product batch, total number of setup times, total setup time of *j*-product and others.

Since the resulting makespan is smaller than the delivery deadline, the start of production can be shifted from Monday 6:00 a.m. to Tuesday 10:49:30 a.m. Also, by means of the Gantt chart, the correct timing of the *j*-product batch is visible. Due to a large number of batches and workdays, the Gantt chart is large and unobtrusive on a small display. For this reason, the Gantt chart for one working day is shown in Fig. 5. In addition to the start date of the product batches on particular production equipment, the end date of the processing of the last workpiece from the batch is shown, as well as the total processing time (the time the product batch spent on particular production equipment).

Table 2 Batch size of products D, E, F for the observed case

Lot_D	Lot_E	Lot_F	$Perc_D$	$Perc_E$	$Perc_F$
274	324	330	776	425	351

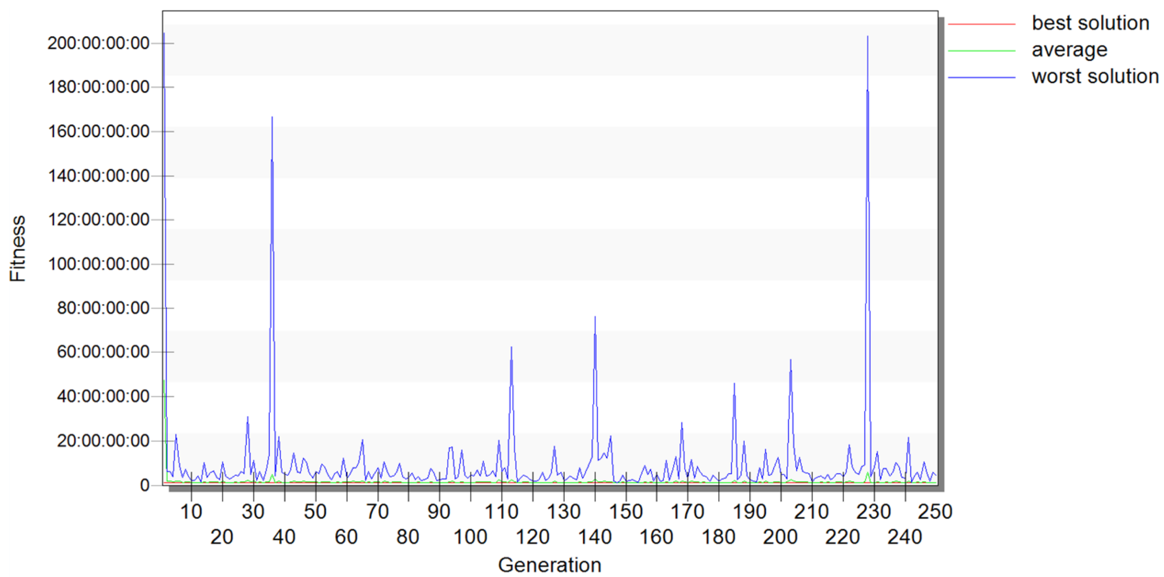

Fig. 3 Evolution diagram shows a quick finding of a enough good solution

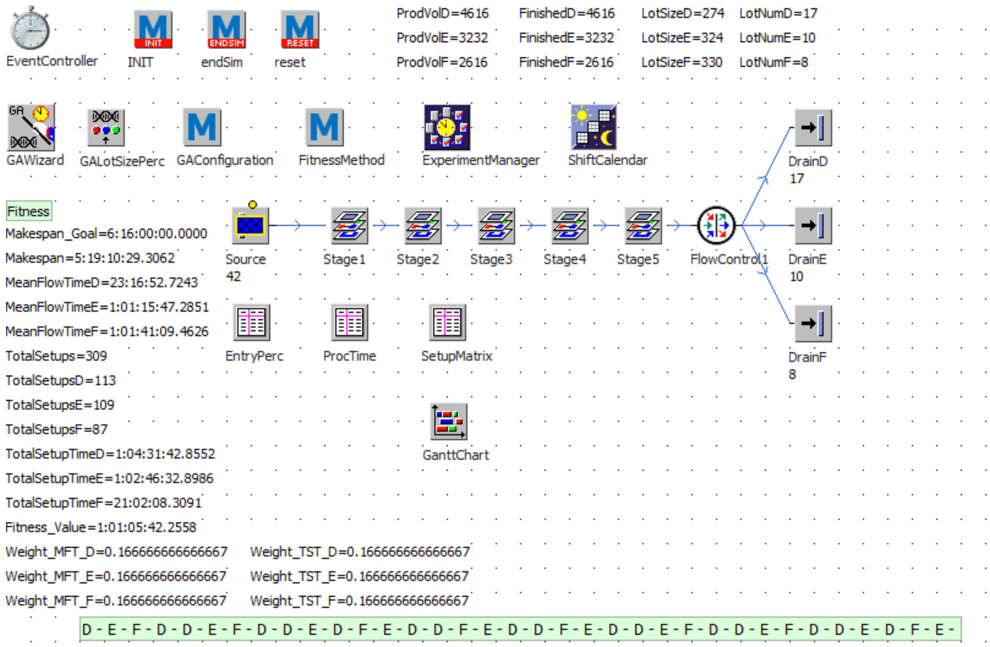

Fig. 4 FSMP model with displayed results

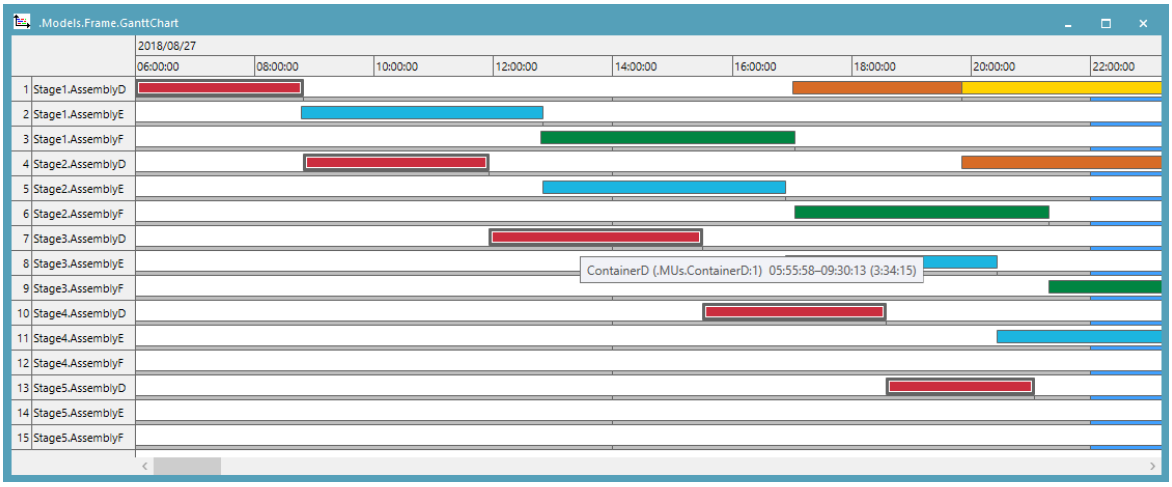

Fig. 5 The Gantt chart of products D, E, F for one working day

6. Conclusion

This paper deals with the problem of determining the batch size and the sequence of entering the product batches into the system, focusing on sequence-dependent setup times. A GA simulation approach was presented as a combination of stochastic modelling of discrete event simulation capabilities and intelligent GA search algorithm. A discrete model for flow shop with multiple processors (FSMP) was developed using the Tecnomatix Plant Simulation software package, specialized in manufacturing engineering as a tool to support the manufacturing planning of technologically similar products.

Based on the developed model and developed procedures within the genetic algorithm, optimal values for the mean flow time and total setup time are obtained, along with the primary condition of delivering the product on time. The GA simulation approach has shown that for the defined performance measures, the mean flowtime and the total setup time, the discrete model provides good solutions. The solution is applicable and shows the exact batch entering into the process and gives a detailed order and timing of entering a particular product batch into the default system. The main contribution of this paper is the simplicity and concision of the display solution that serves as a tool for manufacturing operations planning.

Acknowledgement

This work was financially supported by University of Rijeka, Croatia, contract number uniri-tehnic-18-223-6165 and by University of Rijeka, Croatia, contract number uniri-tehnic-18-100-5893. The authors would like to thank the company iTCR d.o.o. for their extensive help.

References

[1] Pinedo, M.L. (2008). *Scheduling: Theory, algorithms and systems*, 3rd edition, Springer, New York, USA, doi: 10.1007/978-0-387-78935-4.

[2] Kuo, Y., Yang, T., Cho, C., Tseng, Y.-C. (2008). Using simulation and multi-criteria methods to provide robust solutions to dispatching problems in a flow shop with multiple processors, *Mathematics and Computers in Simulation*, Vol. 78, No. 1, 40-56, doi: 10.1016/j.matcom.2007.06.002.

[3] Alfieri, A. (2009). Workload simulation and optimisation in multi-criteria hybrid flowshop scheduling: A case study, *International Journal of Production Research*, Vol. 47, No. 18, 5129-5145, doi: 10.1080/00207540802010823.

[4] Negahban, A., Smith, J.S. (2014). Simulation for manufacturing system design and operation: Literature review and analysis, *Journal of Manufacturing Systems*, Vol. 33, No. 2, 241-261, doi: 10.1016/j.jmsy.2013.12.007.

[5] Allahverdi, A. (2015). The third comprehensive survey on scheduling problems with setup times/costs, *European Journal of Operational Research*, Vol. 246, No. 2, 345-378, doi: 10.1016/j.ejor.2015.04.004.

[6] Ribas, I., Leisten, R., Framiñan, J.M. (2010). Review and classification of hybrid flow shop scheduling problems from a production system and a solutions procedure perspective, *Computers & Operations Research*, Vol. 37, No. 8, 1439-1454, doi: 10.1016/j.cor.2009.11.001.

[7] Gourgand, M., Grangeon, N., Norre, S. (2003). A contribution to the stochastic flow shop scheduling problem, *European Journal of Operational Research*, Vol. 151, No. 2, 415-433, doi: 10.1016/S0377-2217(02)00835-4.

[8] Wang, L., Zhang, L., Zheng, D.-Z. (2005). A class of hypothesis-test-based genetic algorithms for flow shop scheduling with stochastic processing time, *The International Journal of Advanced Manufacturing Technology*, Vol. 25, No. 11-12, 1157-1163, doi: 10.1007/s00170-003-1961-y.

[9] Yang, T., Kuo, Y., Cho, C. (2007). A genetic algorithms simulation approach for the multi-attribute combinatorial dispatching decision problem, *European Journal of Operational Research*, Vol. 176, No. 3, 1859-1873, doi: 10.1016/j.ejor.2005.10.048.

[10] Azadeh, A., Moghaddam, M., Geranmayeh, P., Naghavi, A. (2010). A flexible artificial neural network – Fuzzy simulation algorithm for scheduling a flow shop with multiple processors, *The International Journal of Advanced Manufacturing Technology*, Vol. 50, No. 5-8, 699-715, doi: 10.1007/s00170-010-2533-6.

[11] Chaudhry, I.A., Usman, M. (2017). Integrated process planning and scheduling using genetic algorithms, *Tehnički Vjesnik – Technical Gazette*, Vol. 24, No. 5, 1401-1409, doi: 10.17559/TV-20151121212910.

[12] Meolic, R., Brezočnik, Z. (2018). Flexible job shop scheduling using zero-suppressed binary decision diagrams, *Advances in Production Engineering & Management*, Vol. 13, No. 4, 373-388, doi: 10.14743/apem2018.4.297.

[13] Hendizadeh, S.H., ElMekkawy, T.Y., Wang, G.G. (2007). Bi-criteria scheduling of a flowshop manufacturing cell with sequence dependent setup times, *European Journal of Industrial Engineering*, Vol. 1, No. 4, 391-413, doi: 10.1504/EJIE.2007.015388.

[14] Lin, S.-W., Ying, K.-C. (2012). Scheduling a bi-criteria flowshop manufacturing cell with sequence-dependent family setup times, *European Journal of Industrial Engineering*, Vol. 6, No. 4, 474-496, doi: 10.1504/EJIE.2012.047666.

[15] Lee, G.-C. (2009). Estimating order lead times in hybrid flowshops with different scheduling rules, *Computers & Industrial Engineering*, Vol. 56, No. 4, 1668-1674, doi: 10.1016/j.cie.2008.10.016.

[16] Dugardin, F., Yalaoui, F., Amodeo, L. (2010). New multi-objective method to solve reentrant hybrid flow shop scheduling problem, *European Journal of Operational Research*, Vol. 203, No. 1, 22-31, doi: 10.1016/j.ejor.2009.06.031.

[17] Ladhari, T., Msakni, M.K., Allahverdi, A. (2012). Minimizing the total completion time in a two-machine flowshop with sequence-independent setup times, *Journal of the Operational Research Society*, Vol. 63, No. 4, 445-459, doi: 10.1057/jors.2011.37.

[18] Ying, K.-C., Lee, Z.-J., Lu, C.-C., Lin, S.-W. (2012). Metaheuristics for scheduling a no-wait flowshop manufacturing cell with sequence-dependent family setups, *The International Journal of Advanced Manufacturing Technology*, Vol. 58, No. 5-8, 671-682, doi: 10.1007/s00170-011-3419-y.

[19] Galzina, V., Lujić, R., Šarić, T. (2012). Adaptive fuzzy particle swarm optimization for flow-shop scheduling problem, *Tehnički Vjesnik – Technical Gazette*, Vol. 19, No. 1, 151-157.

[20] Chen, W., Hao, Y.F. (2018). Genetic algorithm-based design and simulation of manufacturing flow shop scheduling, *International Journal of Simulation Modelling*, Vol. 17, No. 4, 702-711, doi: 10.2507/IJSIMM17(4)CO17.

[21] Yan, R., Li, M.M., Wei, W.C. (2018). Integrated production scheduling and distribution planning with a two-stage semi-continuous flow shop environment, *International Journal of Simulation Modelling*, Vol. 17, No. 3, 553-561, doi: 10.2507/IJSIMM17(3)CO15.

[22] Liu, R., Xie, X., Yu, K., Hu, Q. (2018). A survey on simulation optimization for the manufacturing system operation, *International Journal of Modelling and Simulation*, Vol. 38, No. 2, 116-127, doi: 10.1080/02286203.2017.1401418.

[23] Supsomboon, S., Vajasuvimon, A. (2016). Simulation model for job shop production process improvement in machine parts manufacturing, *International Journal of Simulation Modelling*, Vol. 15, No. 4, 611-622, doi: 10.2507/IJSIMM15(4)3.352.

[24] Schroeder, R.G. (1993). Operations management: Decision making in the operations function, McGraw-Hill Education, New York, USA.

[25] Brajković, T., Perinić, M., Ikonić, M. (2018). Production planning and optimization of work launch orders using genetic algorithm, *Tehnički Vjesnik – Technical Gazette,* Vol. 25 No. 5, 1278-1285, doi: 10.17559/TV-20161207195125.

[26] Sharma, P., Jain, A. (2015). Performance analysis of dispatching rules in a stochastic dynamic job shop manufacturing system with sequence-dependent setup times: Simulation approach, *CIRP Journal of Manufacturing Science and Technology*, Vol. 10, 110-119, doi: 10.1016/j.cirpj.2015.03.003.

[27] Mikac, T. (1994). *Optimization of the manufacturing system concept,* PhD thesis (in Croatian), Faculty of Engineering, Rijeka, Croatia.

[28] Koren, Y., Shpitalni, M. (2010). Design of reconfigurable manufacturing systems, *Journal of Manufacturing Systems*, Vol. 29, No. 4, 130-141, doi: 10.1016/j.jmsy.2011.01.001.

[29] Gen, M., Cheng, R. (1997). *Genetic algorithms and engineering design*, John Wiley & Sons, New York, USA, doi: 10.1002/9780470172254.