Performance Investigation of Artificial Bee Colony (ABC) Algorithm for Permutation Flowshop Scheduling Problem (PFSP)

N A Sidek, S A Bareduan and A Nawawi

Faculty of Mechanical and Manufacturing Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Johor, Malaysia

Email: noorazizah@uthm.edu.my

Abstract. The purpose of the Permutation Flowshop Scheduling Problem (PFSP) is to find the best permutation of jobs. PFSP is also considered as an optimization problem and it is often solved using the swarm intelligence approach. In this paper, Artificial Bee Colony (ABC) algorithm is used for solving PFSP. To investigate the effect of trial counter (TC) to the performance of ABC, the TC value was set to six (6), twelve (12) and eighteen (18). Additionally, the percentage of errors was selected as the response. From the experimental results, it can be concluded that ABC algorithm performs best when the TC is set at six (6) because the value of cumulative error percentages is at the lowest. Moreover, the generated data is also located in the smallest spread compared to TC=12 and TC=18. It can also be concluded that the exploration principle works better in a 6 jobs and 3 machines PFSP environment.

1. Introduction

A good scheduling system will contribute directly to the performance of a manufacturing enterprise. In the actual industries, the most common scheduling layout is the flowshop scheduling problem (FSP). Additionally, FSP is applied in a manufacturing environment where a family of products (with some similar features) to be manufactured [1], [2].

Permutation flowshop scheduling problem (PFSP) is one of the most discussed problem in FSP. The purpose of PFSP is to find the best processing (job) sequence in order to satisfy the optimization requirement. In most of the literature, the objective PFSP optimization is the makespan which is considered as an act to minimize the total completion time for the production processes [3]–[5]. The formulas for calculating the makespan are shown by Eq. (1)-(4).

Generally, PFSP is used to find the best jobs’ permutation in order to generate the lowest makespan. One basic rule of PFSP is that a job cannot be stopped once it started. Furthermore, the sequence of machine has to be fixed and as a result, the production planner will only has the chance to “play around” with the sequence of jobs (refer Figure 1).
Figure 1. An illustration of a 6 jobs and 3 machines flowshop

\begin{align*}
C(\pi_1,1) &= p(\pi_1,1) \\
C(\pi_j,1) &= C(\pi_{j-1},1) + p(\pi_j,1) \quad j = 2, \ldots, n \\
C(\pi_k,1) &= C(\pi_{k-1},1) + p(\pi_k,1) \quad k = 2, \ldots, m \\
C(\pi_j,k) &= \max \{C(\pi_{j-1},k), C(\pi_j,k-1) + p(\pi_j,k)\} \quad j = 2, \ldots, n; k = 2, \ldots, m
\end{align*}

Notation

- $C$ : Completion time
- $k$ : Machine identifier
- $\pi$ : Job representative
- $m$ : Number of jobs
- $p$ : Processing time of each job
- $n$ : Number of machines
- $j$ : Job identifier

PFSP is also considered as a highly complex optimization problem (NP-complete). Due to its complexity, PFSP is often solved using heuristic and metaheuristics [5]. There are also nature-inspired algorithms used to solve PFSP such as the Artificial Bee Colony (ABC) algorithm [5]–[7], Genetic Algorithm (GA) [8], [9] and Particle Swarm Optimization (PSO) algorithm [10].

In this paper, an Artificial Bee Colony (ABC) algorithm is used for optimizing PFSP. In addition, the performance of ABC was investigated by varying the value of the trial counter (TC). The performance of ABC was measured in the form of error’s percentage which represents the difference between the optimum solution and the solution generated by ABC.

2. Artificial Bee Colony (ABC) Algorithm
ABC algorithm was invented by Darvis Karaboga in 2005. It is inspired by the foraging behavior of honey bees. ABC was used in various areas such as the function optimization, software testing, exam timetabling, truss structure optimization and production scheduling optimization [5], [7], [11]–[14]. The sequence of ABC algorithm is as follows [15].
2.1. Initialization Phase

ABC starts with the initialization phase where the food sources (solutions) is generated. This phase is performed by the scout bees (SB) and the trial counter (TC) value is also set during this stage [18]. In addition, this phase also yields the solutions in the form of n-dimensional vector,

\[ X_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,n}\} \]

Eq. (5) is used to generate the solutions [5], [16].

\[ x_{i,j} = LB_j + r(UB_j - LB_j) \]  

Based on Eq. (5), \( x_{(i,j)} \) is the ith solution of dimension j. The lower and upper boundaries of dimension j are represented by \([LB]_j\) and \([UB]_j\) respectively. Moreover, r is a uniform random number in the range of 0 to 1.

2.2. EB Phase

In the EB Phase, the employed bees (EB) will search for new solutions (food sources) based on their memories. After that, the EBs will generate the candidate solutions arranged in the form of n-dimensional vector, \( Y_i = \{y_{i,1}, y_{i,2}, \ldots, y_{i,n}\} \). The formula for generating the candidate solutions is presented by Eq.(6) [5], [16].

\[ y_{i,j} = x_{i,j} + \varnothing (x_{i,j} - x_{k,j}) \]

In the equation, \( x_{(i,j)} \) is the solutions generated by Eq. (5). \( x_{(i,j)} \) and \( x_{(k,j)} \) are located in the same dimension, j but from a different positions. Additionally, \( \varnothing \) is a random number between 0 to 1.

After finding the solutions, the greedy selection method will be applied. In this method, a new and better solution will replace the older one [16]. The information regarding any new food sources (location, distance and fitness value) is transferred via the waggle dance. The waggle dance helps the bees to find the food sources without using any map[17].

2.3. OB Phase

In the OB Phase, onlooker bees (OBs) will use the information originated from the waggle dance for choosing the food sources. Furthermore, any solution with a higher fitness value will be selected. Eq. (7) is used for calculating the probability, \( P_i \) for a food source to be selected by the OB [5], [16].

\[ P_i = \frac{\text{fit}_i}{\sum_{n=1}^{SN} \text{fit}_n} \]  

\[ \text{fit}_i \]
According to the equation, \( \text{fit}_i \) is the fitness value of food source \( X_i \) and SN is the number of solution. OBs will generate their version of solutions based on the information received from the EBs. The solutions generated by the OBs will be evaluated with regards to their fitness. Similarly, the greedy selection method will be used for selecting only the best solution.

2.4. SB Phase
A food source will be abandoned if it cannot be improved after a number of times set by the trial counter (TC). This scenario will initiate the SB Phase. In this phase, the EB that owns the abandoned food source will become a scout bee (SB) and will go out to find a new food source [5], [15], [18].

3. Application of ABC in PFSP
The main goal of PFSP is to obtain the best permutation (sequence) of jobs in order to minimize the value of makespan. This process is very crucial for cutting down the production lead time. In the same way, the production efficiency can be enhanced due to the higher rates of machines utilization.

In this paper, a 6 jobs and 3 machines PFSP scenario is considered. As an example, Table 1 shows the lists of processing times for each job on a particular machine. It is also worth to note that the makespan value is equal to the total time needed to complete all jobs. Based on the example in Table 1, the makespan value is 47 minutes (refer Figure 2).

| Number of Job | Machine 1 | Machine 2 | Machine 3 |
|---------------|-----------|-----------|-----------|
| Job 1         | 5         | 6         | 8         |
| Job 2         | 4         | 5         | 7         |
| Job 3         | 7         | 5         | 6         |
| Job 4         | 6         | 4         | 5         |
| Job 5         | 5         | 3         | 6         |
| Job 6         | 8         | 3         | 4         |

Table 1. An example of the processing time of 6 jobs in 3 machines [4]
Figure 2. Gantt chart for the flowshop. The green, orange and blue coloured bars represent Machine 1, Machine 2 and Machine 3 respectively (The number in each bar is the processing time in minutes)

According to the figure, the gap between jobs is known as the waiting time. In PFSP, it is very important to eliminate the waiting time as much as possible. From here, an effective arrangement of jobs (permutation) is considered as the way to reduce the amount of waiting times.

In ABC, when there is no improvement to the result after several iterations, the food source will be abandoned. The number of repetitions allowed before abandoning the food source is set by the trial counter (TC). The TC is very useful for initiating the searching process in a new area. It is also considered as the medium to set the balance between the exploration and exploitation activities.

A large value of TC will make the particles (bees) to search in the same area and this will enhance the exploitation capability. Once the food source is abandoned, scout bees are released to find a new area and this activity is also known as the exploration activity. A lower TC value will encourage the exploration process because the scout bees will travel more frequently to find new areas.

In this paper, the TC value is the factor to be varied. This is important for investigating its effect on the performance of ABC algorithm. The TC was set at six (6), twelve (12) and eighteen (18). The experiment was performed on 100 sets of data. The response (output) was measured in the form of errors percentage. Error percentage is the difference between the optimum solution for each jobs’ sequence and the solution generated using ABC. To calculate the error percentage, Eq. (8) was utilized.

\[
Error \ Percentage = \left( \frac{\text{Generated Makespan} - \text{Optimum Makespan}}{\text{Optimum Makespan}} \right) \times 100
\]  

(8)

Figure 3 depicts the structure of bees for each TC variation. Based on the figure, the first set of flow chart depicts a scenario where TC is set at six (6). There are six bees in the group (3 EBs and 3 OBs). In this scenario, the optimum results are compared between the six bees. Similarly, the results will be compared among twelve (12) or eighteen (18) bees if the TC=12 or TC=18 respectively. As mentioned previously, a higher TC value will result in the better exploitation activities. As a trade-off, the completion time for TC=18 is longer than TC=12.
4. Results and Discussion

Based on the experimental results, the highest number of errors was accumulated when TC=18 (refer Figure 4). The figure shows a time series plot for all TC variants. This plot is useful for presenting the frequency of errors for each TC variant. In the figure, it can be seen that TC=18 shows the highest error frequency. This result shows the level of quality for the particular TC value. Furthermore, the least amount of errors were recorded when TC=6. Additionally, the least amount of errors depicts a higher reliability value.

Additionally, the time series plot is also useful for determining the behavior of ABC towards the 6 jobs and 3 machines PFSP. From the time series plot, it can be seen that the majority of data yielded by TC=6 has no error (error percentage equal to zero).
Figure 4. The time series plot shows the value of errors for each run.

Figure 5. The amount of errors for each TC.

The justification made previously is supported by the histogram in Figure 5. Based on the histogram, TC=6 generated the most error-free data while the TC=18 yields the least amount of data with zero error percentage. This comparison shows that TC=6 is more efficient compared to other TCs.
As the value of error percentage goes bigger (refer the x-scale of histogram), the TC=18 yields the highest amount of data. This means that TC=18 tends to generate data with high value of error. This is another characteristic that is not favorable by the decision makers. Based on the histogram, it can be seen that TC=6 generated the highest amount of data without error. Moreover, TC=12 and TC=18 showed almost equal amount of non-error data.

However, in TC=18, the data with high amount of error is available. Due to that, TC=12 can be considered as the better setting compared to TC=18. Table 2 shows the values of mean and standard deviation for each TC. In order to get a better visualization of the data spread, Figure 6 and 7 can be referred.

Table 2. The mean and standard deviation values for all TC

| Trial counter (TC) | Mean  | StDev |
|-------------------|-------|-------|
| 6                 | 1.185 | 2.338 |
| 12                | 1.322 | 2.476 |
| 18                | 1.965 | 3.848 |

From the table, the best setting is when TC is set to six (6). This is due to the fact that the mean for the generated data is lower. A lower mean shows that the accumulated value of error is at minimum. Additionally, when TC=6, the value of standard deviation (StDev) is also the lowest among others. This is a sign that the data for TC=6 is distributed closer to the mean value compared to other TCs.

The justifications made from Table 2 are supported by Figure 6. In the figure, it can be seen that TC=6 generated the lowest mean followed by TC=12 and finally TC=18. A higher value of mean indicates that the generated data contains higher error percentages.

Figure 6. The interval plot shows the spread of data for each trial counter value
In the boxplot (refer Figure 7), the data generated by TC=6 are located in a smaller spread compared to other TCs. This shows that TC=6 is more precise compared to TC=12 and TC=18. Moreover, the outliers points for TC=6 also located in the smallest range compared to other TCs.

![Boxplot of TC=6, TC=12, TC=18](image)

**Figure 7.** The boxplot depicts the mean comparison and data variability

The p-value for the experiment is 0.141 which can be considered as a large value. As a result, the null hypothesis should be accepted. In the null hypothesis, the mean values for all TCs are almost equal. However, the mean of TC=18 is quite far from the other two. As a consequence, the setting of TC=18 is not suitable for the usage of ABC in PFSP.

Additionally, the mean of TC=6 and TC=12 are almost equal. It can also be seen that the data for TC=12 is located in a wider range. Due to that, TC=12 can be categorized as less precise compared to TC=6. From here, it can be said that TC=6 is the best setting because it generates lower amount of errors and the data spread is in the lowest range.

5. Conclusion
In this paper, the value of TC was varied in order to investigate its effect on the performance of ABC algorithm. The value of TC is crucial for achieving a good balance between the exploration and exploitation activities. Based on the experimental results, it can be concluded that TC=6 is the best setting because it generates data with the lowest mean and standard deviation. These characteristics shows that TC=6 is more reliable, efficient and precise compared to TC=12 and TC=18. Additionally, TC=6 also yields the lowest error percentages and the data is distributed closely to the mean. A lower error percentages will result in a more efficient algorithm. Moreover, TC=6 can also be considered as the most reliable setting because the generated data has the lowest variation. Further research efforts
should focus on other ABC parameters such as number of bees, number of iterations and different PFSP environments.

Acknowledgement
This research is funded by UTHM under the Short Term Grant (STG Vot U368). Additionally, the authors would like to thank the Faculty of Mechanical and Manufacturing Engineering of UTHM for providing the equipment and software utilized in this research.

References
[1] Naderi B and Ruiz R 2014 A scatter search algorithm for the distributed permutation flowshop scheduling problem Eur. J. Oper. Res. 239(2) 323–334
[2] Moccellin J V 1995 A new heuristic method for the permutation flow shop scheduling problem J. Oper. Res. Soc. 46(7) 883–886
[3] Naderi B and Ruiz R 2010 The distributed permutation flowshop scheduling problem, ” Comput. Oper. Res. 37(4) 754–768
[4] Sidek N A and Bareduan S A 2014 Effect of Trial Counter Variation on the Performance of Artificial Bee Colony (ABC) Algorithm for Permutation Flowshop Scheduling Problem (PFSP) International Conference on Mathematics, Engineering and Industrial Applications (ICoMEIA 2014)
[5] Liu Y-F and Liu S-Y 2013 A hybrid discrete artificial bee colony algorithm for permutation flowshop scheduling problem Appl. Soft Comput. 13(3) 1459–1463
[6] Han Y-Y, Duan J-H and Zhang M 2011 Apply the discrete artificial bee colony algorithm to the blocking flow shop problem with makespan criterion 2011 Chinese Control and Decision Conference (CCDC) 2131–2135
[7] Tasgetiren M F, Pan Q-K, Suganthan P N and Chen A H L 2010 A discrete artificial bee colony algorithm for the permutation flow shop scheduling problem with total flowtime criterion IEEE Congress on Evolutionary Computation 1–8
[8] Gao J, Chen R and Liu Y 2012 A Knowledge-based Genetic Algorithm for Permutation Flowshop Scheduling Problems with Multiple Factories 4 121–129
[9] Chen S-H, Chang P-C, Cheng T C E and Zhang Q 2012 A self-guided genetic algorithm for permutation flowshop scheduling problems Comput. Oper. Res. 39(7) 1450–1457
[10] Liu H, Gao L and Pan Q 2011 A hybrid particle swarm optimization with estimation of distribution algorithm for solving permutation flowshop scheduling problem Expert Syst. Appl. 38(4) 4348–4360
[11] Karaboga D and Basturk B 2007 A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm J. Glob. Optim. 39(3) 459–471
[12] Alzaqebah M and Abdullah S 2011 Artificial bee colony search algorithm for examination timetabling problems Int. J. Phys. Sci. 6(17) 4264–4272
[13] Sonmez M 2011 Artificial Bee Colony algorithm for optimization of truss structures Appl. Soft Comput. 11(2) 2406–2418
[14] Hadidi A, Azad S K and Azad S K 2010 Structural optimization using artificial bee colony algorithm 2nd international conference on engineering optimization
[15] Akay B and Karaboga D 2012 A modified Artificial Bee Colony algorithm for real-parameter optimization Inf. Sci. (Ny). 192 120–142
[16] Banharnsakun A, Sirinovakul B and Achalakul T 2012 Job Shop Scheduling with the Best-so-far ABC Eng. Appl. Artif. Intell. 25(3) 583–593
[17] Karaboga D 2005 An idea based on honey bee swarm for numerical optimization Techn. Rep. TR06, Erciyes Univ. Press. Erciyes
[18] Karaboga D, Gorkemli B, Ozturk C and Karaboga N 2012 A comprehensive survey: artificial bee colony (ABC) algorithm and applications Artif. Intell. Rev. 42(1) 21–57