Staff scheduling for a courier distribution centre using evolutionary algorithm

Pua Si Ying, Zeratul Izzah Mohd Yusoh
Computational Intelligence and Technologies Laboratory (CIT), Department of Intelligent Computing and Analytics,
Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Durian Tunggal, Malaysia

Article Info

Article history:

Received Sep 14, 2021
Revised Jun 8, 2022
Accepted Jun 16, 2022

Keywords:

Courier distribution centre
Evolutionary algorithm
Genetic algorithm
Optimization
Staff scheduling

ABSTRACT

Staff scheduling is a combinatorics optimization problem and companies face this complex task on daily basis in constructing a schedule fitting all conditions. In a courier distribution center, staffs are assigned to work in processes of a continuous workflow. Staffs have varying work ability for each process. Instead of generating staff schedule instinctively, it is an advantage to optimize staff's schedule by measuring the performance of each staff. An optimized schedule improves the operation's efficiency and fully utilize staffs' work ability, hence, minimizing the cost. This paper proposed evolutionary algorithm, namely genetic algorithm as the solution to courier center staff scheduling. Based on the result, the produced schedule can reduce up to 30% of the staff in schedule while not affecting operation workflow. The cut down on number of working staffs could amount to a substantial reduction of operation cost every month. The generated schedule is significantly customized and take less time to complete an operation. Although the proposed solution is specific to the use case of a courier distribution center, it is however, potentially a generalize model for the logistics industry, introducing a more effective staff scheduling system to cope with the industry's ever-rising demands.

Corresponding Author:

Zeratul Izzah Mohd Yusoh
Computational Intelligence and Technologies (CIT) Laboratory
Department of Intelligent Computing and Analytics
Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia
Email: zeratul@utem.edu.my

1. INTRODUCTION

The growth of e-commerce industry has increased the demands of courier services, especially during seasonal peaks and sales, further increasing the workload of the already stretch thin workforce of courier companies [1]. In times like this, arranging a feasible staff's schedule for a courier distribution centre is essential to achieve maximum efficiency to fulfil market's demands.This problem has been a challenge to courier companies as they strive to serve the demands. This paper proposed an evolutionary algorithm to produce an optimal staff's schedule based on one of the international courier companies in Malaysia as a case study.

Due to the arising of e-commerce industry, the logistics industry has a rapid increase in demand to accommodate to consumers' needs. The logistics industry in Malaysia has been given special emphasis in the industrial master plan 3 2016-2020 (IMP3), which further pushes the growth of the local logistics industry. In a report by DHL Express and IBM, the logistics industry is more than ready to integrate with artificial intelligence (AI) technologies as it is now accessible and affordable to a company seeking performance boost

[2]. In addition, based on research conducted on courier service quality, one of the main criteria for a good courier service is the implementation of information and communication technology (ICT) technologies in the service itself [3]. ICT technology can be applied in various activities of logistic industry. For instance, in operations planning, insights can be provided to forecast parcel volumes using reliable data [4].

In scheduling automation using AI, Job shop scheduling (JSP) is a combinatorial optimization problem classified as NP-hard [5]-[9]. JSP involves assigning various jobs, otherwise called processes, with varying attributes to a set of machines to be processed or completed. The goal of a standard JSP is to minimize the makespan of the whole process [10]. As for the problem of the courier distribution centre, with reference to standard JSP, machines, *M* can be mapped to staffs, *S*. Whereas Jobs, *J* can be mapped to flow, *F*. Operations within a job, *O* can be mapped to processes, *P*. A flow is complete when all processes had been completed by at least one staff. Make span in the context of JSP, can be mapped to the time taken to complete the whole operation. There are several hard constraints and soft constraints regarding process assignment to staff. Since this problem of staff scheduling for the courier distribution center is an extended version of standard JSP, it is also a NP-hard problem.

In solving a NP-hard problem, metaheuristics techniques are more widely used against other techniques. This is due to its capability to find the optimal solution within reasonable time [11]. Some of the metaheuristic techniques used in scheduling problems are evolutionary algorithms including genetic algorithm, differential evolution, artificial bee colony, ant colony optimization, memetic algorithm and many more [8], [12]-[14]. In JSP, genetic algorithm is the most used method [15]. Existing literature on courier centre distribution are focusing on routing and delivery problems, and several works that solving the problem using metaheuristic techniques are reviewed [16]-[20]. Based on the reviews done, the solution proposed to the scheduling problem is highly dependent on the characteristic of the problem itself especially on how the problem and constraints are formulated. However, it is proven that meta heuristic technique is a good solution for such problem, with a common technique being used is genetic algorithm [21].

The remainder of the paper is organised as follows. Section 2 presents the proposed algorithm for the problem. Section 3 present the result the algorithm, and finally, Section 5 summarises and concludes the paper.

2. METHOD

2.1. The problem's background

A courier distribution center processes incoming parcels and redistribute the parcels towards their destination. The courier distribution center used as a case study handles two operations, inbound, and outbound. Inbound operation receives incoming parcels from a flight and distribute the processed incoming parcels locally through trucks. Outbound operation receives parcels from courier trucks locally, processes them in the reverse order of inbound operation for international distribution. Figure 1 and Figure 2 depicts the general inbound and outbound operation respectively. A unit load device (ULD) is a shipment container containing all the incoming parcels unloaded from a flight. Each operation has different flow; each handles a type of parcel. There are three types of parcels: conveyable (COY), flyer (FLY), and non-conveyable (NCY). The term Flyer also refers to two types of parcel DOX and WPX.

The clearance rate by arrival of parcel at the centre is almost always at 80% of total incoming parcel volume. A term for cleared parcels upon arrival at the centre is cleared on arrival (COA). Parcels that is not cleared upon arrival are sent to bond cage (TBC) to await instructions.

In general, there are several flows of process handled at the courier distribution centre. Each of the flow corresponds to one type of parcel. If the parcels are further divided to COA or TBC, another flow is required. For flyers, this process is entirely managed by conveyor belt, hence requiring no extra flow. To summarize, a courier distribution centre generally handles five flows, based on parcel's type and clearance.

The goal of the staff's scheduling is to minimize the time to complete an operation. Based on the courier distribution centre requirement, an operation must be completed within one hour. Hence, the measurement of staff'scheduling is parcel per hour (PPH), referring to the number of parcels able to be processed by a staff within an hour. A process also has a target PPH, referring to the number of parcels need to be processed within an hour by staffs at a particular process point. PPH of staff is observed and recorded based on the staff's performance on a process. PPH of processes varies according to total incoming parcel volume and parcel type. Table 1 shows breakdown of parcel volume by parcel type.

There are constraints to be satisfied when assigning staffs to processes. First, all processes must be assigned to at least one staff, leaving no process unassigned. Second, a staff cannot be assigned to concurrent processes of different flows. Lastly, the PPH value of each process must be satisfied by the combined PPH value of all staffs working on a particular process. This is to ensure that one whole operation can be completed within the time frame of 1 hour as required by the courier company.

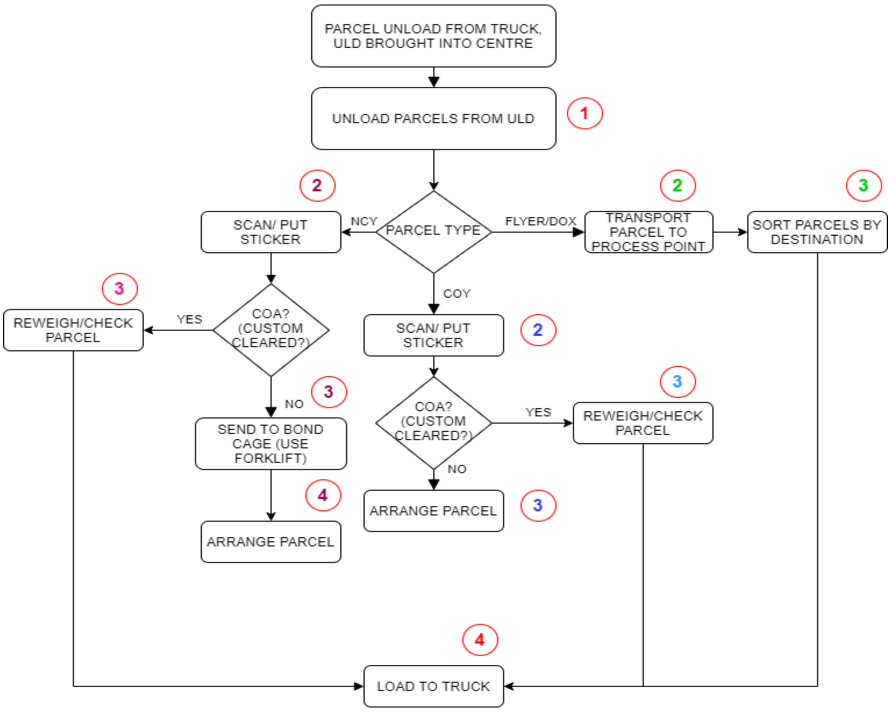

Figure 1. Inbound operation

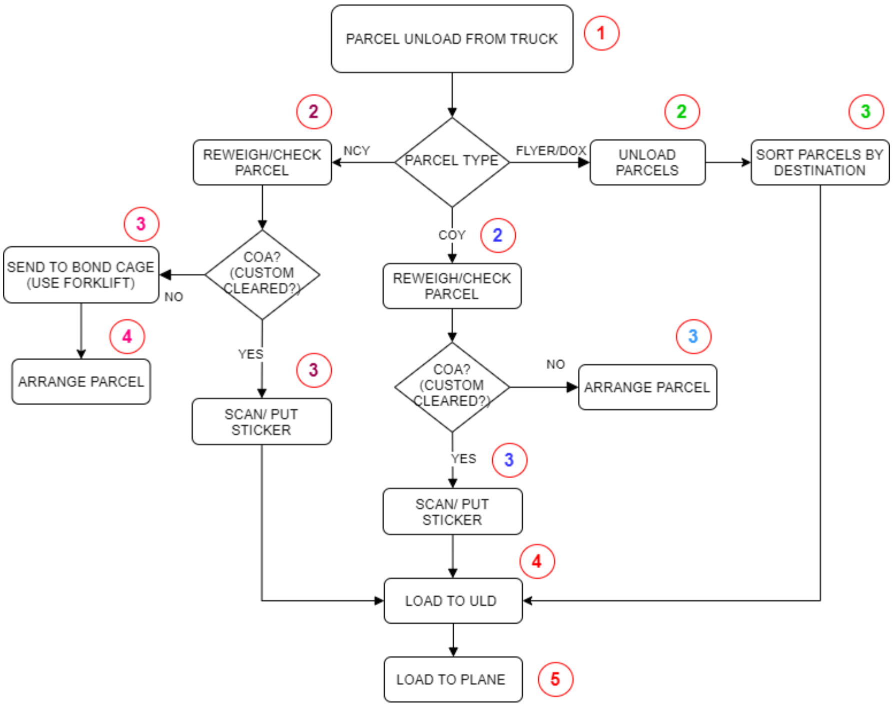

Figure 2. Outbound operation

Table 1. Parcel volume by parcel type

Parcel Type	Percentage of total volume	Breakdown
COY	59%	-
FLY	40%	-
- DOX		25% of 40%
- WPX		75% of 40%
NCY	1%	-

2.2. The evolutionary algorithm

The proposed evolutionary algorithm is a genetic algorithm. The aim is to produce optimal staff schedule by minimizing the total processing time and at the same time minimize the number of staff used. The schedule produced also have to comply with several constraints outlined in the problem. Algorithm 1 illustrate the overall flow of a genetic algorithm and the following sections will elaborate in detail each of the algorithm's component.

```
Algorithm 1. Proposed genetic algorithm
bestFitness = 0
bestSchedule = 0
create an initial population Population of PopSize individuals, based
on the initialisation procedure in Section 2.1
while termination condition is not true do
 for X ∈ Population do
  Calculate X fitness value, F(X), penalised if X has the invalid or
  illogical schedule
  if F(X) > bestFitness then
     Replace bestFitness
     bestPlan = X
  end
 end
 Select individuals from the Population based on binary tournament
 selection
 Probabilistically apply the crossover operator to generate new
 individual
 Probabilistically select individuals for mutation
 Use the new individuals to replace the old individuals in the
 Population
end
Output bestFitness
Output bestSchedule
```

2.2.1. Chromosome representation

Each gene in a chromosome represents a staff selected for scheduling. For instance, gene 1 represents staff 1, gene 2 represents staff 2 and so on. The gene's value is a set of processes assigned to the staff in a range of 0 to the highest number of processes in all flows selected. For instance, two flows are selected for scheduling, flow 1, and flow 2. Flow 1 has four processes, flow 2 has three processes. Flow 1 has the highest number of processes in all flow. Hence, a staff is only possible to have been assigned a maximum of 4 processes, since a staff cannot be assigned to concurrent processes. When a staff has been assigned less than 4 processes, the gene is filled with 0 to ensure all chromosomes have equal length. The genes are separated by the indicator '|'. The order in which the processes are assigned matters. They represent the order of work for the staff. Figure 3 depicts the chromosome representation. Processes assigned are represented as alphabets.

The gene values are assigned randomly on the condition there is no clashes among processes for a staff. After the random initialization, the set of processes will be sorted according to its precedence value. Sorting the processes according to their precedence eliminates errors of reversed work flow in the chromosome.

A,B,C,0 | F,0,0,0 | B,D,0,0 | E,G,0,0 | D,L,0,0 | F,K,L,0 | E,H,L,M |

Figure 3. Chromosome representation

2.2.2. Fitness evaluation

The fitness function is implemented using the normalized weighted additive utility function (NWAUF) [22]. There are two objectives that will be minimized: i) the time taken to complete the operations, and ii) the number of staff used. These two objectives are calculated separately, normalized, multiplied by a weight, and added together in one function. With the weight multiplied with the objectives, the importance of each objective can be adjusted easily according to preferences or algorithm performance.

The first objective, *TIME* refers to the time taken to complete the operations. It is calculated by sing the combined PPH value of all staffs assigned to a process divided by the PPH of a process. This will roduce the time needed to complete the process, in a range of (0, 1), 1 indicating one hour is needed, since number of parcel is divided by the parcel per hour of staffs. The resulting value will be in the unit of one hour as well.

The second objective, $STAFF$ refers to the number of staffs involved in operations in this generated schedule or chromosome. It is calculated by counting the number of staffs involved, divided with the total number of staffs. With this, the staff objective is measured as the lower the value, the better the solution is. Since this is in contrary to the first objective, it is first multiplied by the staff weight, SW, then minus by the staff weight. TW refers to time objective's weight.

$$TIME = (1/\text{total time} * \text{TW}) \tag{1}$$

$$STAFF = \left(SW - (\textit{staffs involved/total staffs} * SW)\right) \tag{2}$$

$$fitness = STAFF + WEIGHT \tag{3}$$

Penalties are applied to chromosomes that produce invalid or illogical schedule. Two types of penalties are introduced, namely, the clash penalty and invalid penalty. Both of the penalties are calculated and then added to the fitness value defined in (3). The clash penalty calculates clashes among the number of processes assigned to each staff by their precedence. In other words, if process A with precedence value 1 and process B with precedence value 1 is both assigned to a staff, one clash is calculated. This calculation is accumulated for every gene in a chromosome to find out the total number of clashes. A generated schedule with clashes among processes is illogical because a staff cannot be working on two processes with the same precedence as these happen concurrently. The number of clashes calculated is divided by the maximum number of clashes possible, then, multiplied by a weightage. The result is then minus by the weightage itself. This is presented in (4).

$$illogical\ penalty = Weight_{clash\ penalty} - \left(\left(\frac{clashes}{maximum}\right) * Weight_{clash\ penalty}\right) \tag{4}$$

Where clashes is the number of clashes among assigned processes, whereas maximum is the total possible clashes.

The invalid penalty calculates the number of processes that is not assigned to any staff at all. A generated schedule is invalid if there are unassigned processes. Unassigned processes in (5) refers to number of unassigned processes, maximum refers to total number of processes.

$$invalid\ penalty = Weight_{invalid\ penalty} - \left(\left(\frac{unassigned\ processes}{maximum}\right) * Weight_{invalid\ penalty}\right) \tag{5}$$

Both penalty values are added to the fitness value earlier. A good chromosome will have higher fitness value whereas a bad chromosome will have a lower fitness value. To further differentiate among invalid and valid chromosomes, a threshold value is applied to the fitness value. If the chromosome has at least 1 clashes or unassigned process, the evaluated fitness value will be multiplied by sum of the weightage of both of the penalties. With this, the fitness value of chromosomes with clashes or unassigned processes will not be higher than the threshold. For instance, a chromosome with fitness value 0.90 is very good in terms of time taken to complete whole operations, however, if it has at least 1 clashes, the value is then multiplied by the sum of both penalties, say, (0.3+0.3)=0.6. Hence, 0.9*0.6=0.54 as the final fitness value. Hence, in this case the threshold is the sum of both penalties, 0.6. Any chromosomes with fitness value 0.6 is valid. The formula is as (6).

$$fitness = fitness * (invalid\ penalty + illogical\ penalty) \tag{6}$$

2.2.3. Selection processes and genetic operations

The strategies for the selection processes and genetic operations are based from existing literature with similar work [23]-[25], and then the strategies are modified to suit with the problem. The strategies are implemented and tested to choose the final strategy for the evaluation process. For the selection, the Tournament Selection will be used to select parents, and survival selection uses elitism technique. Elitism technique retains the few top performing chromosomes from the old generation to be brought into the next generation. The rests are replaced by the children according to best fitness. For the genetic operations, the crossover operation uses two-point crossover and random mutation technique is used in which three genes will be mutated randomly by replacing the genes with another value. In this problem, the crossover rate is set as 0.9. Mutation rate is set as 0.2. The algorithm stops when it reaches a certain number of generations.

3. RESULT AND DISCUSSION

For the evaluation, the data were provided by the courier center. There are 13 processes, 22 staffs, and 286 PPH values. The processes are from 6 different flows handling different parcel types with different precedence values. Each staff has varying PPH values for each of the processes. The total number of parcels to process is set at 5500 parcels, where it is the standard amount of number of parcels to be processed at a courier distribution centre. Each process has different number of parcels to process, which needs to be covered by the combined PPH values of the staffs working on this process. The breakdown of the number of parcels to process for each process is calculated. The process of handling FLY parcels of clearance COA will have 2090 parcels to process. Analysis of the staff scheduling module's algorithm performance is conducted at the system testing stage. Parameter tunings of the algorithm is done to suit with the problem. The final parameters values are the ratio of clash penalty in the fitness function to 0.6, decreasing ratio of time to 0.2, the population size 70 and maximum generation of 20, the crossover rate is 0.9, and the mutation rate is 0.2. Figure 4 depicts the graphical results of a run using these settings. This setting is able to generate a good result with a processing time in the range of 30 seconds to 2 minutes.

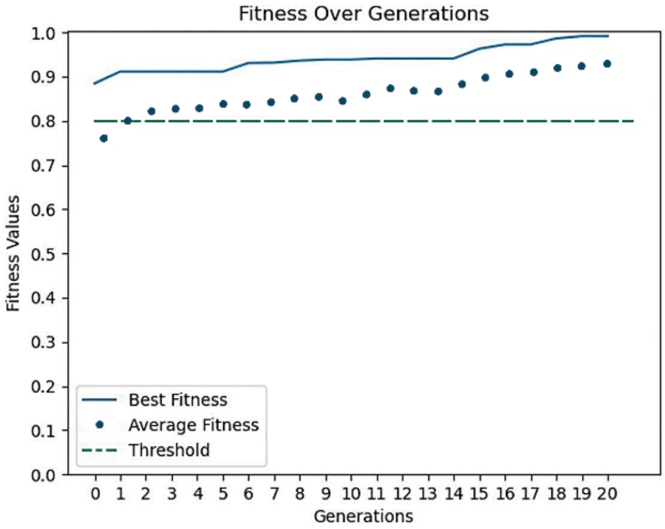

Figure 4. 70 generations with population size 70

The algorithm performance is also evaluated by the average and standard deviation of several indicators: fitness, time taken, number of staffs involved, and clashes. The fitness value ranges from 0-1, the higher indicating a better result. Time taken refers to the total time needed to complete the operations using the generated schedule, hence, it is the lower the better, with favorable values around 1. Number of staffs involved indicates the number of staffs that has been assigned to work in this schedule, which is the lower the better. Clashes refers to the number of processes assigned to staffs that clashed with one another in terms of processes' precedence, this indicator is the lower the better. Table 2 shows the results of 10 runs.

Table 2. Average and standard deviation of algorithm performance

Runs	Fitness	Time taken (hour)	Number of staffs	Clashes
1	0.9195	1.3663	22	0
2	0.9617	1.2361	22	0
3	0.9818	1.1	22	0
4	0.9348	1.4834	22	0
5	0.9552	1.0384	22	0
6	1	0.9614	22	0
7	0.9650	1.211	22	0
8	0.9681	1.18918	22	0
9	1	0.9614	22	0
10	1	0.9796	22	0
Average	0.9686	1.1527	22	0
Std Dev	0.0277	0.1781	0	0

From the results, the average fitness value of 10 runs is 0.96, indicating the algorithm is able to produce high quality results in almost every run. A standard deviation of 0.02 for fitness value indicates that the quality of schedules generated is consistent. The time taken indicator has an average value of 1.15, indicating the average schedules generated will be slightly more than one hour, however still in the satisfactory range. A standard deviation of 0.17 is higher than the standard deviation of fitness value, hence, the time taken indicator variates more, either less than one hour or more than one hour. The number of staffs maintains the same value throughout all 10 runs in this setting, with an average of 22 and standard deviation 0, proving 22 staffs is enough to generate a good schedule consistently. The last indicator, clashes, has a value of 0 throughout all 10 runs, proving the algorithm is able to generate viable and quality schedule with no clashes among assigned processes for each staff.

4. CONCLUSION

Based on observation at the courier-distribution centre, there was generally around 33 staffs working onsite during an operation. Based on the result, the produced schedule can bring this number down to 22 staffs, reducing 10 staffs while not affecting operation workflow. The cut down on number of working staffs could amount to a substantial reduction of operation cost every month. Besides cost efficiency, the general time efficiency of the operations is improved. Schedules generated assign processes to staffs based on parcel amount to handle, which enables the flexibility of assigning more or less staff to a process based on demand. The processes are also assigned in a manner that maximize a staffs' capabilities, assigning a staff to another process, once a process has completed. Such schedules will take up a big amount of an administrative staff's time to construct. To sum up, the proposed evolutionary algorithm is able to generate an optimal staff schedule for courier distribution center based on the time taken and total number of staff used. The proposed solution can improve the cost, time, and human resource efficiency of courier companies. The algorithm could transform staff scheduling in the logistic industry and improve the efficiency of local courier companies towards serving their customers better thus increasing their advantages in this increasing competitive, and on-demand industry. For future work, the implementation of concurrent execution is to be explored since the algorithm execution time took around 60 to 120 seconds to complete due to the large number of data.

ACKNOWLEDGEMENTS

We acknowledge financial assistance received and the support from Universiti Teknikal Malaysia Melaka and the data provided by the courier company through Dream Catcher Innovate Malaysia.

REFERENCES

[1] J. Del Ser, A. I. Torre-Bastida, I. Lana, M. N. Bilbao, and C. Perfecto, "Nature-inspired heuristics for the multiple-vehicle selective pickup and delivery problem under maximum profit and incentive fairness criteria," in *2017 IEEE Congress on Evolutionary Computation (CEC)*, 2017, pp. 480-487, doi.org/10.1109/cec.2017.7969350.

[2] M. Krstić, S. Tadić, and S. Zečević, "Technological solutions in logistics 4.0," *Ekonomika preduzeca*, vol. 69, no. 6–7, pp. 385–401, 2021, doi: 10.5937/ekopre2106385k.

[3] A. Gulc, "Courier service quality from the clients' perspective," *Engineering. Management in Production and Services*, vol. 9, no. 1, pp. 36–45, 2017. doi: 10.1515/emj-2017-0004.

[4] D. C. Bissoli, W. A. S. Altoe, G. R. Mauri, and A. R. S. Amaral, "A simulated annealing metaheuristic for the bi-objective flexible job shop scheduling problem," in *2018 International Conference on Research in Intelligent and Computing in Engineering (RICE)*, 2018, pp. 1-6, doi.org/10.1109/rice.2018.8627907.

[5] E. Kim and H. Jung, "Short-term prediction model of postal parcel traffic based on self-similarity," *Journal of Society of Korea Industrial and Systems Engineering*, vol. 43, no. 4, pp. 76–83, Dec. 2020, doi: 10.11627/jkise.2020.43.4.076.

[6] P. Deepalakshmi and K. Shankar, "Role and impacts of ant colony optimization in job shop scheduling problems: A detailed analysis," *Evolutionary Computation in Scheduling*, pp 11-35, 2020, doi: 10.1002/9781119574293.ch2.

[7] F. Yang, C. M. Ye and M. H. Shi, "A hybrid grey cuckoo search algorithm for job-shop scheduling problems under fuzzy conditions," *Advances in Production Engineering & Management*, vol. 13, no 3, pp 254-266, 2018, doi: 10.14743/apem2018.3.288.

[8] M. A. A. Alabajee, A. A. Fadhil and R. G. Alsarraj, "Job shop scheduling problem: Literature review," *Tikrit Journal of Pure Science*, vol. 25, no. 4, pp 91-100, 2020, doi: 10.25130/tjps.25.2020.075.

[9] A. S. Abdalkafor, A. A. Jihad and E. T. Allawi, "A cloud computing scheduling and its evolutionary approaches," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 21, no. 1, pp. 489-496, 2021, doi: 10.11591/ijeecs.v21.i1.pp489-496.

[10] M. Ehsaei and D. T. Nguyen, "A new job shop heuristic algorithm for machine scheduling problems," *Journal of Applied Mathematics and Physics*, vol. 5, no. 11, 2017, doi: 10.4236/jamp.2017.511177.

[11] S. N. Mhasawade and L. A. Bewoor, "A survey of hybrid metaheuristics to minimize makespan of job shop scheduling problem," in 2017 *International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, 2017 pp. 1957-1960, doi: 10.1109/icecds.2017.8389792.

[12] B. Khan and P. Singh, “Selecting a meta-heuristic technique for smart micro-grid optimization problem: A comprehensive analysis,” in *IEEE Access*, vol. 5, pp. 13951-13977, 2017, doi: 10.1109/ACCESS.2017.2728683.

[13] S. S. Sadiq, A. M. Abdulazeez and H. Haron, “Solving multi-objective master production schedule problem using memetic algorithm,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 18, no. 2, pp. 938-945, 2020. doi.org/10.11591/ijeecs.v18.i2.pp938-945.

[14] A. Faiz, S. Subiyanto and U. M. Arief, “An efficient meta-heuristic algorithm for solving capacitated vehicle routing problem,” *International Journal of Advances in Intelligent Informatics*, vol. 4, no. 3, pp. 212-225, Nov. 2018, doi: 10.26555/ijain.v4i3.244.

[15] I. A. Chaudhry and A. A. Khan, “A research survey: review of flexible job shop scheduling,” *International Transactions in Operational Research*, vol. 23, no. 3, pp. 551-591, 2016, doi: 10.1111/itor.12199/

[16] A. F. O. Gaffar, A. Wajiansyah and S. Supriadi, “The map-based shortest route selection by using ant colony optimization algorithm,” *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 1, no. 1, p. 15, Mar. 2018, doi: 10.24014/ijaidm.v1i1.4600.

[17] K. Lee, J. Chae and J. Kim, “A courier service with electric bicycles in an Urban Area: The case in Seoul,” *Sustainability*, vol. 11, no. 5, 2019, doi: 10.3390/su11051255.

[18] M. H. Syauqi and T. Y. M. Zagloel, “Optimization of heterogeneous vehicle routing problem using genetic algorithm in courier service,” in *Proceedings of the 3rd Asia Pacific Conference on Research in Industrial and Systems Engineering*, 2020, pp. 48-52, doi: 10.1145/3400934.3400945.

[19] B. P. Bruck, A. G. dos Santos and J. E. C. Arroyo, “Hybrid metaheuristic for the single vehicle routing problem with deliveries and selective pickups,” in *IEEE Congress on Evolutionary Computation*, pp. 1-8, 2012, doi: 10.1109/cec.2012.6256456.

[20] J. D. Ser, M. N. Bilbao, C. Perfecto, S. Salcedo-Sanz, “A harmony search approach for the selective pick-up and delivery problem with delayed drop-off,” *Harmony Search Algorithm*, pp. 121-131, 2016, doi: 10.1007/978-3-662-47926-1_13.

[21] N. S. Ibrahim, N. M. Yahya, and S. B. Mohamed, “Metaheuristic nature-inspired algorithms for reservoir optimization operation: a systematic literature review,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 26, no. 2, pp. 1050-1059, 2022, doi: 10.11591/ijeecs.v26.i2.pp1050-1059.

[22] M. Dai, Z. Zhang, A. Giret, and M. A. Salido, “An enhanced estimation of distribution algorithm for energy-efficient job-shop scheduling problems with transportation constraints,” *Sustainability*, vol. 11, no. 11, 2019, doi: 10.3390/su11113085.

[23] M. H. Abed and M. N. M. Kahar, “Guided genetic algorithm for solving unrelated parallel machine scheduling problem with additional resources,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 26, no. 2, pp. 1036-1049, 2022, doi: 10.11591/ijeecs.v26.i2.pp1036-1049.

[24] M. Yuan, Y. Li, L. Zhang and F. Pei, “Research on intelligent workshop resource scheduling method based on improved NSGA-II algorithm,” *Robotics and Computer-Integrated Manufacturing*, vol. 71, 2021, doi: 10.1016/j.rcim.2021.102141.

[25] W.-Q. Zou, Q.-K. Pan, and L. Wang, “An effective multi-objective evolutionary algorithm for solving the AGV scheduling problem with pickup and delivery,” *Knowledge Based System*, vol. 218, 2021, doi: 10.1016/j.knosys.2021.106881.

BIOGRAPHIES OF AUTHORS

Pua Si Ying was born in Pahang, Malaysia. She completed her Bachelor degree of Computer Science in Artificial Intelligence in 2021 from the Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM). She is currently working as an AI Engineer at an institution providing intelligent software as a service (SaaS) for the healthcare industry. Her work essentially involves computer vision, deep learning, and machine learning algorithms. She has co-developed several deep learning projects such as Toilet Entrance Monitoring with neural network and proprietary algorithms, Human Profilization using Thermal Images. Currently, her research interests lie in the area of improving neural networks performance and deployment pipeline with MLOps. She can be contacted at psiying97@gmail.com.

Zeratul Izzah Mohd Yusoh is a senior lecturer in the Department of Intelligent Computing and Analytics, Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM) where she has been a faculty member since 2003. Her PhD work is focusing on SaaS Resource Management System in Cloud Computing using Evolutionary Computation. Her research interests lie in the area of developing soft computing solutions to improve the system’s intelligence. In recent years, her research is focusing on intelligent analytics and evolutionary computation techniques. She can be contacted at zeratul@utem.edu.my.