International Journal of Engineering & Technology, 7 (4.31) (2018) 336-340

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Parameter Tuning of Groceries Scheduling by Genetic Algorithm

Mohd Nazmi Noor Shamsuazman [1], Shuzlina Abdul Rahman[2], Nurzeatul Hamimah Abdul Hamid[3]*

[1]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA,
40450 Shah Alam, Selangor, Malaysia
[2]Research Initiative Group of Intelligent Systems,
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA,
40450 Shah Alam, Selangor, Malaysia
**Corresponding author E-mail: nurzeatul@tmsk.uitm.edu.my*

Abstract

Genetic Algorithm (GA) is a promising optimization algorithm that can provide effective solutions to help overcome optimization problems. However, applying the GA without parameter tuning may result in a low efficiency. Thus, this study investigates the importance of parameter tuning and its applications in solving an optimization problem. The parameter tuning was experimented on groceries scheduling problems to find the shortest time required in delivering the groceries. The experiments were performed on different values of mutation operator, crossover operator and number of iterations on groceries scheduling. The knowledge representation and the architecture of the problem are presented. The improvements of the search ability are verified by a series of experiments. Our results show that the best mutation operator is the Inversion Mutation, while the best crossover operator is Order 1 Crossover.

Keywords: *Genetic Algorithm; Optimization; Parameter Tuning; Scheduling.*

1. Introduction

The increase in e-commerce transactions over the past years requires its enablers such as delivery services to be more efficient. The last mile delivery becomes very critical as consumers expect the products to reach them as fast as possible. This paper focuses on the delivery schedule generation to guarantee the groceries are delivered to the consumers. One of the important things to manage the delivery of groceries is the delivery schedule. Scheduling is the process of managing, controlling and optimizing the work and the workload in the process. Scheduling allocates resources such as raw materials, human resources planning, planning of production processes and material purchases. In groceries last mile delivery, orders must arrive at the specific address in the time window chosen by the consumers. In this context, the scheduling should be carried out to determine the order of the delivery location and solve many delivery locations at every time slot [1].

In addition, the delivery usually requires one vehicle to make delivery in one area. In the case where the delivery needs to deliver to more than one area, it requires more than one vehicle to ensure the delivery arrives on time [2]. Therefore, to complete this task, it requires a structured schedule to ensure consumers receive their groceries in the specified time window. Therefore, an alternative approach is needed to identify and solve scheduling problems for facilitating the delivery process.

Genetic Algorithm (GA) appears as one of the effective solutions to optimization algorithms that can provide good results for such a problem. However, applying the GA without parameter tuning may result in a low efficiency. Thus, this study attempts to investigate and tune the parameters that affect the performance of GA. The experiments were based on groceries scheduling problems to find the shortest time in delivering the groceries. This paper is organized as follows. The next section presents the related works while section 3 describes the methodology. In section 4, the experiments results are discussed and finally we conclude the paper in Section 5.

2. Related Works

Artificial Intelligence (AI) approach refers to a technique used to create a system that is as smart and intelligent as humans. McCarthy [3], AI is "The science and engineering of making intelligent machines, especially intelligent computer program". AI attempts to make computers capable of doing tasks humans can do. AI can be achieved by studying how the human brain thinks, how people learn to understand a thing, the way people make decisions, how people solve the problems, and then use the findings as a basis for developing intelligent systems. It attempts to incorporate reasoning and other high-level capabilities in activities such as planning and scheduling to achieve designated goals [4].

Various AI techniques such as expert systems, fuzzy logic, neural networks, and genetic algorithms are designed to solve the complex problem [5-7]. AI techniques have evolved from as simple as rule-based representation in expert systems to algorithms that allows self-organization and self-learning to achieve its designated goals. Problem solving in AI context is the process of searching practical solutions based on the environmental representation [6]. Some complex problems include scheduling, optimization, forecasting such as weather forecasts, traffic control, and disease diagnosis. In these complex problems, solutions search is implemented using more sophisticated algorithms.

The design of heuristics or meta-heuristics algorithms aims to produce elegant solutions to achieve the goals in the most optimum way [6]. However, exploring the suitable parameter tuning prior to applying the computational model is vital for stochastic problem. Many researchers have stated the importance of parameters tuning to yield a better result [9-11]. In [9] for example, the parameter tuning techniques were performed in analyzing the behavior of a computational model of psychosocial influences on physical activity behavior. Another study by Cheung et al. [10] produced an automated parameter tuning to overcome the limitation of manual tuning in producing a high-quality point clouds for stereo vision system. A different approach by means of statistical test was performed by [11] to know which parameters have the greatest influence on the behaviour and performance of the algorithm.

2.1 Optimization Algorithms

Optimization can be defined as population-based methods for finding solutions for some mathematical problems [6]. Optimization is a task to find the alternative ways for finding the most cost-effective or to achieve the highest performance at the given constraints, by maximizing or minimizing the desired factors. For comparison purposes, maximizing means trying to achieve results or results of the highest or maximum regardless of costs or expenses. The practice of optimization is limited by the lack of full information and lack of time to assess what information is available.

GA has been developed in the year 1970's in the USA. The early names in GA are J. Holland, K. DeJong, and D. Goldberg, they have developed the algorithms based on the process of natural selection from Darwin's Evolution Theory [13]. Darwin's Theory is a theory related to biological evolution created by Charles Darwin (1809-1882). He states that all species of organisms arise and thrive through natural selection. These species often increase the individual's ability to compete, survive, and reproduce. Besides, GA belongs to the larger class of evolutionary algorithms (EA). GA process is to engender solutions to optimization problems using techniques inspired by natural evolution such as inheritance, mutation, selection, and crossover [14]. In GA, the population of the candidate is solutions to an optimization problem by evolving towards better solutions [15]. GA is the simplest and most widely available method that has been used successfully in solving various problems such as image processing, delivery problems, economics, scheduling, control problems and more [6] [16] [17].

GA has been used in many problems as been mentioned in many past studies [18-21]. Alaïa et. al [18] proposes a combination of genetic algorithms with clustering algorithms. They [18] aim to solve the various vehicles, various depots, pickup and delivery problems. The clustering algorithm is used to group nearby nodes by calculating Euclidean distance between each pickup node and delivery node. Then, GA is deployed to find the shortest path to go to all nodes. With a combination of these two methods, the objective of minimizing total vehicle usage and overall travel distance can be achieved.

Wang [19] solves the optimization problem using GA technique to solve the same problem such as the Traveling Salesman Problem (TSP). The TSP is a traditional combinatorial optimization problem, which is simple to state but very difficult to solve. The main challenge of the problem is that the traveling salesman wants to minimize the total length of the trip. Based on an experiment that has been created to solve TSP, the experiment results show a better performance. Additionally, it can determine an optimal solution quickly. It is suitable and among the best algorithm to solve the vehicle routing problem.

Similar study conducted by Jalaluddin et. al [20] in which a Courier Delivery Services Visualisor (CDSV) was developed to visualize the best route to be taken by the courier service providers so that they courier service can deliver on time. The CDSV employed the GA and A-star Algorithm (A*) that integrates with Geographical Information System (GIS) data. A Graphical User Interface (GIU) in the form of simulation maps that suggests the delivery best route and the optimal distance are displayed for easier courier service distribution route references. In another study, the Monte-Carlo technique was integrated with GA to improve the strength of the tropical timber joint strength [21] in which a satisfactory result was obtained. Other than GA, other optimization algorithms such as the Particle Swarm Optimization (PSO) and the Ant Colony Optimization (ACO) were also proposed to solve the optimization problem. The combination of PSO and ACO was proposed in [22] for the mobile Radio Frequency Identification System (RFID) reader to get the shortest path for object localization, and satisfactory results were obtained. Those studies however, appear to lack in discussing the parameter tuning and its effects towards solving the optimization goal.

3. Methodology

This section explains the methodology of the study i.e. the knowledge representation and the system's architecture based on the groceries scheduling problem. Our data such as the distance of every location in Shah Alam is extracted from Google Maps. The next two subsections present the details of the processes.

3.1 Knowledge Representation

In this phase, we used the collected data and the data from the customer (e.g. customers address, arrival time) to determine the distance between the starting point to the customer place. A filtering technique was used to assign the runner or deliverer to deliver the groceries. This technique was used to make the next step easier in which we first filtered the large group into the smallest groups. Then, we generated the random population, calculate the fitness function, parent selection, crossover, mutation, and elitism to find the shortest distance between customers' places in a group.

The parameters to be analyzed is the number of iterations, type of crossover (order 1 and cycle) that shows in Figure 2 and Figure 3, and types of mutation which are swap, insertion, inversion, displacement and scramble that are shown in Figure 4. Equation 1 shows the fitness function used in this study where x represents the distance.

$$\min f(x) = \sum_{i=0}^{n-1} x_i \tag{1}$$

Figure 1 indicates a chromosome representation based on the distance between customer's place, the arrival time to the customer's place and several shipments that can be made within one hour. For example, assuming located locations are sections 2, 6, 7, 13 and 14 in which each location has its distance value. Next, we used GA method to find the shortest distance for every section.

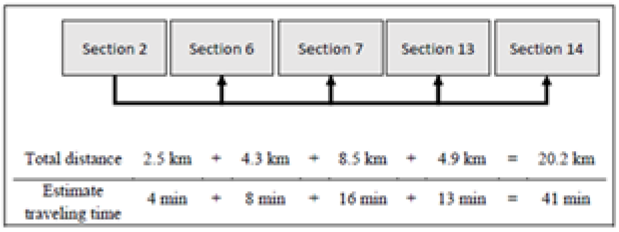

Fig. 1: Chromosome Representation

Figure 2 shows the process of crossover using order 1 crossover involving 6 sections on two parents; parent 1 and parent 2. As can be seen, the shaded boxes of parent 1 and parent 2 will be exchanged (crossover) to form new offsprings, parent 1 and parent 2.

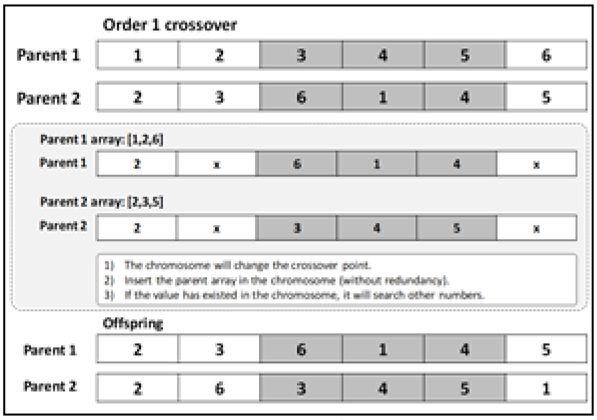

Fig. 2: GA Operator (Order 1 Crossover)

Figure 3 shows the process of crossover using cycle crossover involving 6 sections on two parents; parent 1 and parent 2. As can be seen, parent 2 will copy the value based on the same index as illustrated in the Figure. The detail steps are illustrated in the Figure. Figure 4 shows the process of mutations which are swap, insertion, inversion, displacement and scramble.

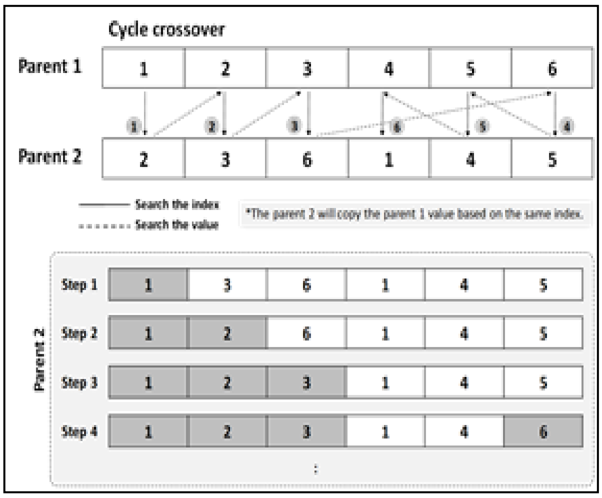

Fig. 3: GA Operator (Cycle Crossover)

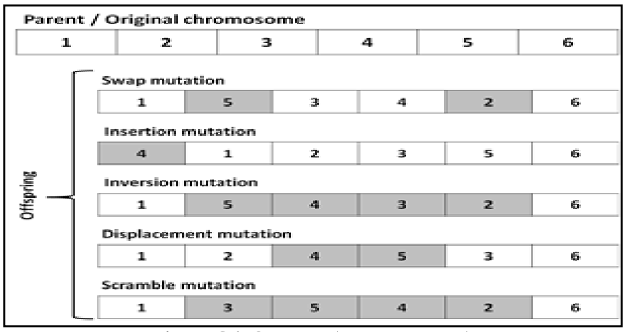

Fig. 4: GA Operator (Mutation Type)

3.2 System Architecture

The system architecture is initiated by receiving the inputs from the user, which are shopping and delivery details. Then, the information will be stored in the database before the next process. Secondly, the system will retrieve the related information, which is delivery details (delivery date, time and location) and list of the deliverers. There are two processes involved, which are filtering and genetic algorithm. In the filtering process, the delivery location will be filtered based on the number of delivery, whereby each delivery consists of not more than six delivery locations. This is because the system is created for delivering the groceries since the groceries cannot last long. The purpose of this filtering process is to generate the delivery schedule.

Thirdly, the GA process is to find the best sequence of the path. In the GA engine, the process is started by initializing the parameters which are the number of population, number of iterations, crossovers and mutation operators. The next process is generating the random population and calculate the fitness for each candidate in the population. Next, the system will check with the stopping condition, which is the number of iterations. If the iteration equals with the number of iterations, the system will select two best results from the population to undertake the crossover and mutation processes. Then, the system will evaluate the new candidates and carry out the elitism. In the elitism process, the current candidate will replace the best candidates from the crossover and mutation results with the worst candidates in the population. The process will be repeated until the number of iterations have been achieved. Lastly, the output will show the delivery schedule and list of the deliverers. Figure 5 shows the System Architecture.

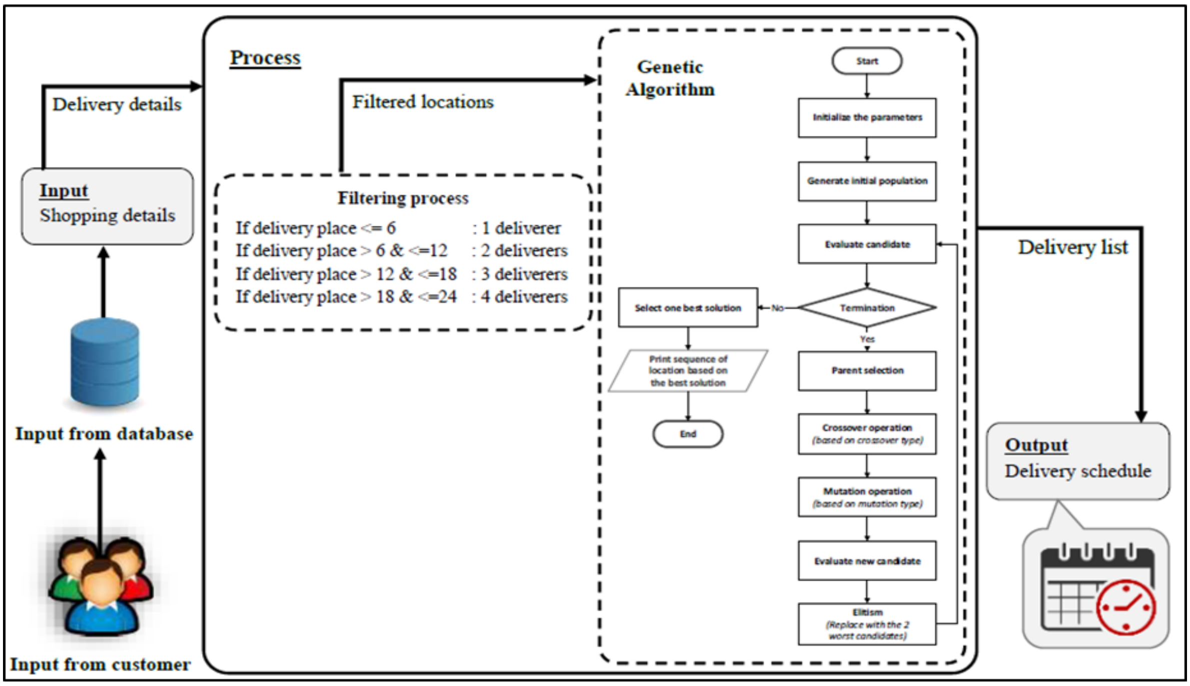

Fig. 5: System Architecture

4. Experiment Result

4.1 First Experiment: Different Mutation Type

In the first experiment, three cases were conducted. This experiment uses the same number of population, which is 100 candidates and a different number of iterations, which are 50, 100 and 150 iterations. From this experiment as shown in Table 1, we found that the best mutation operator is the inversion mutation.

Table 1: Summary of Mutation Type

Case number	Number of iteration	Best mutation
Case 1	50	Insertion
	100	**Inversion**
	150	Insertion
Case 2	50	**Inversion**
	100	Insertion
	150	**Inversion**
Case 3	50	**Inversion**
	100	**Inversion**
	150	**Inversion**

4.2 Second Experiment: Different Crossover Type

The purpose of the second experiment is to find which crossover operator that can give the optimal results. This experiment uses the same number of population, which is 100 candidates and a different number of iterations, which are 50, 100 and 150 iterations. Table 2 shows the best crossover operator generated from the second experiment. It shows that the best crossover operator that is based on minimum distance is order 1 crossover.

Table 2: Summary of Crossover Type

Case number	Number of iteration	Best crossover
Case 1	50	**Order 1**
	100	Cycle
	150	**Order 1**
Case 2	50	**Order 1**
	100	**Order 1**
	150	**Order 1**
Case 3	50	Cycle
	100	Cycle
	150	Cycle

4.3 Third Experiment: Different Number of Iteration

This third experiment aims to find what number of iterations is best to provide optimal results. This experiment uses the same number of population, which is 100 candidates and a different number of iterations, which are 50, 100 and 150 iterations. From this third experiment as shown in Table 3, the best results generated is from 150 iterations. Therefore, we can conclude that the greater the number of iterations, the better the results.

Table 3 : Summary of Number of Iteration

Case number	Number of iteration	Crossover	Mutation	Result (km)
Case 1	50	Order 1	Insertion	28.60
	100	Cycle	Inversion	28.13
	150	Order 1	Insertion	**27.90**
Case 2	50	Order 1	Inversion	42.50
	100	Order 1	Insertion	41.30
	150	Order 1	Inversion	**38.87**
Case 3	50	Cycle	Inversion	60.90
	100	Cycle	Inversion	60.00
	150	Cycle	Inversion	**58.83**

4.4 Case 1

In case 1, the system was tested with 8 sections, which are sections 2, 5, 13, 15, 17, 19, 21 and 23 with 40320 possible sequence paths. The results have been produced using 100 number of population, 150 number of iteration, order 1 crossover, insertion mutation and the optimal sequence path with a total distance of 27.90 km. Figure 6 shows the optimal graph for case 1, and as we can see, the optimal results have been achieved starting from iteration number 78.

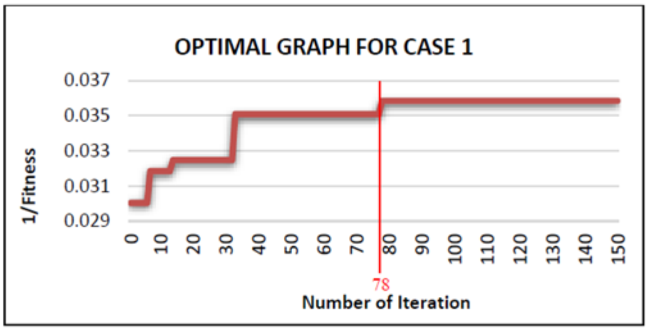

Fig. 6: Optimal Graph for Case 1

4.5 Case 2

In case 2, the system was tested with 17 sections, which are sections 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 and 24 with 355687428096000 possible sequence paths. The results have been produced using 100 number of population, 150 number of iteration, order 1 crossover, inversion mutation and the optimal sequence path with a total distance 38.87 km. Figure 7 shows the optimal graph for case 2, and as we can see, the optimal results have been achieved starting from iteration number 146.

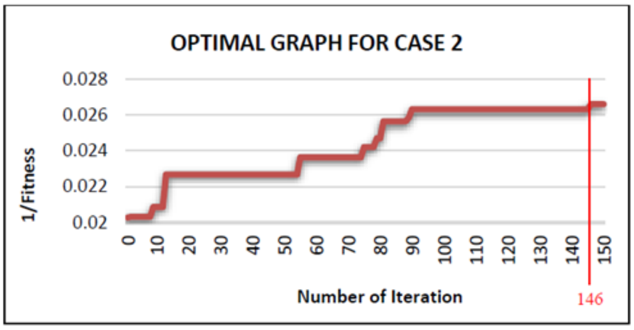

Fig. 7: Optimal Graph for Case 2

4.6 Case 3

In case 3, the system was tested with 23 sections, which are sections 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23 and 24 with 2.585201673888498e+22 possible sequence paths. The results have been produced using 100 number of population, 150 number of iteration, cycle crossover, inversion mutation and the optimal sequence path with a total distance of 58.83 km. Figure 8 shows the optimal graph for case 3, and as we can see, the optimal results have been achieved starting from iteration number 123.

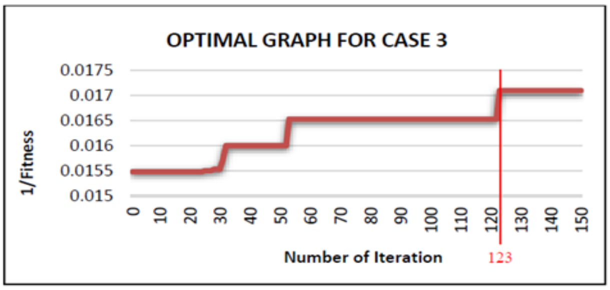

Fig. 8: Optimal Graph for Case 3

The system was tested with three conditions, which are Case 1 with 8 sections, Case 2 with 17 sections and Case 3 with 23 sections. The purpose of the experiment is to find the best number of iterations, best crossover operator, and best mutation operator. Based on the findings, the increasing value of iteration can produce better results. Furthermore, the best combination of the genetic operators is order 1 crossover and inversion mutation.

5. Conclusion

The research experimented on the application of parameter tuning in solving groceries scheduling problem. This was done by using GA to assist the delivery services companies deliver their products within the minimum time by optimizing the sequence location. The parameters such as mutation and crossover were experimented on three different cases. It appears that the best mutation operator on minimum distance was the inversion mutation, while the best crossover operator on minimum distance was order 1 crossover. However, the limitations of this system are that the system does not use the Google API to get the actual address and does not visualize the route for the user. A solution to overcome the problem can be derived by employing multi-objective optimization to visualize the best route, and to get the real-time data of traffic conditions. In addition, more variables such as the traffic conditions, road speed limits and real-time speed could be incorporated in the said system.

Acknowledgement

The authors would like to thank Faculty of Computer and Mathematical Sciences, and Research Management Centre of Universiti Teknologi MARA for supporting this research.

References

[1] Bitar, A., Dauzère-Pérès, S., & Yugma, C. (2014, 7-10 Dec. 2014). On the importance of optimizing in scheduling: The photolithography workstation. Paper presented at the Proceedings of the Winter Simulation Conference 2014.
[2] Sakurai, Y., Tsuruta, S., Onoyama, T., & Kubota, S. (2009, 11-14 Oct. 2009). A multi-inner-world Genetic Algorithm using multiple heuristics to optimize delivery schedule. Paper presented at the 2009 IEEE International Conference on Systems, Man and Cybernetics.
[3] McCarthy, J. (2007). What Is Artificial Intelligence? Computer Science Department Stanford University.
[4] Boden, M. A. (2016). AI: Its nature and future. Oxford University Press.
[5] Halim, Shamimi A., Azlin Ahmad, Norzaidah Md Noh, Azliza Mohd Ali, Nurzeatul Hamimah Abdul Hamid, Siti Farah Diana Yusof, Rozianawaty Osman, and Rashidi Ahmad. "A development of snake bite identification system (N'viteR) using Neuro-GA." In Information Technology in Medicine and Education (ITME), 2012 International Symposium on, vol. 1, pp. 490-494. IEEE, 2012.
[6] Ahmad, A., Yusof, R., & Mitsukura, Y. (2015). Pheromone-based Kohonen Self-Organizing Map (PKSOM) in clustering of tropical wood species: Performance and scalability. In Control Conference (ASCC), 2015 10th Asian (pp. 1-5). IEEE.
[7] Shamsuddin M.R.B., Sahar N.N.B.S., Rahmat M.H.B. (2017) Eye Detection for Drowsy Driver Using Artificial Neural Network. In: Mohamed A., Berry M., Yap B. (eds) Soft Computing in Data Science. SCDS 2017. Communications in Computer and Information Science, vol 788. Springer, Singapore
[8] Poole, D. L., & Mackworth, A. K. (2017). Artificial Intelligence: foundations of computational agents 2nd Edition. Cambridge University Press.
[9] Mollee J.S., Araújo E.F.M., Klein M.C.A. (2017) Exploring Parameter Tuning for Analysis and Optimization of a Computational Model. In: Benferhat S., Tabia K., Ali M. (eds) Advances in Artificial Intelligence: From Theory to Practice. IEA/AIE 2017. Lecture Notes in Computer Science, vol 10351. Springer, Cham
[10] Cheung, Ernest CH, Jiali Wong, Joceyln Chan, and Jia Pan. "Optimization-based automatic parameter tuning for stereo vision." In Automation Science and Engineering (CASE), 2015 IEEE International Conference on, pp. 855-861. IEEE, 2015.
[11] Castillo, Pedro A., Maribel García Arenas, Nuria Rico, Antonio Miguel Mora, Pablo García-Sánchez, Juan Luis Jiménez Laredo, and J. J. Merelo. "Determining the significance and relative importance of parameters of a simulated quenching algorithm using statistical tools." Applied Intelligence 37, no. 2 (2012): 239-254.
[12] Fletcher, R. (2013). Practical Methods of Optimization: Wiley.
[13] Davis, L. (1991). Handbook of genetic algorithms: Van Nostrand Reinhold.
[14] Wen-jiang, Z., Hong-bin, Y., & Yue, W. (2010, 23-25 Nov. 2010). Parallel genetic algorithm based on Thread-Level Speculation. Paper presented at the 2010 International Conference on Audio, Language and Image Processing.
[15] Kapil, S., Chawla, M., & Ansari, M. D. (2016, 22-24 Dec. 2016). On K-means data clustering algorithm with genetic algorithm. Paper presented at the 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC).
[16] Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning: Addison-Wesley Publishing Company.
[17] Holland, J. H. (1975). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence: University of Michigan Press.
[18] Alaïa, E. B., Dridi, I. H., Bouchriha, H., & Borne, P. 2015, Insertion of new depot locations for the optimization of multi-vehicles multi-depots pickup and Delivery Problems using Genetic Algorithm. Proceedings of International Conference on Industrial Engineering and Systems Management (IESM)
[19] Wang, Y. (2012, 24-28 June 2012). Distribution route optimization of logistics enterprise based on genetic algorithm. Paper presented at the World Automation Congress 2012.
[20] Jalaluddin, M.F., Kamaru-Zaman, E.A., Abdul-Rahman, S. and Mutalib, S., Courier Delivery Services Visualisor (CDSV) with an Integration of Genetic Algorithm and A* Engine, Proceedings of the 6th International Conference on Computing and Informatics, ICOCI 2017, pp. 601-606.
[21] Yusoff, M., Othman, D. F., Latiman, A. T., & Hassan, R. (2016). Evaluation of Hybrid Monte-Carlo And Genetic Algorithm for Tropical Timber Joint Strength. Journal of Engineering and Applied Sciences, 11(7), 1682-1686.
[22] Zakaria, M. Z., & Jamaluddin, M. Y. (2016). Path Optimization and Object Localization Using Hybrid Particle Swarm and Ant Colony Optimization for Mobile RFID Reader. Journal of Information Sciences and Computing Technologies, 6(1), 568-576.