

## Research Article

# An Evolutionary Algorithm: An Enhancement of Binary Tournament Selection for Fish Feed Formulation

Cai-Juan Soong <sup>1</sup>, Rosshairy Abd Rahman <sup>2</sup>, Razamin Ramli <sup>3</sup>,  
Mohammed Suhaimee Abd Manaf,<sup>4</sup> and Chek-Choon Ting<sup>5</sup>

<sup>1</sup>Faculty of Data Science & Information Technology, INTI International University, Nilai, Malaysia

<sup>2</sup>Institute of Strategic Industrial Decision Modelling, Universiti Utara Malaysia, Changlun, Malaysia

<sup>3</sup>School of Quantitative Sciences, Universiti Utara Malaysia, Changlun, Malaysia

<sup>4</sup>Fisheries Research Institute, Department of Fisheries, Kedah, Malaysia

<sup>5</sup>Koperasi Aquaculture, Sungai Udang, Penang, Malaysia

Correspondence should be addressed to Cai-Juan Soong; [soongcj19812014@gmail.com](mailto:soongcj19812014@gmail.com)

Received 23 June 2022; Revised 20 September 2022; Accepted 28 September 2022; Published 22 November 2022

Academic Editor: Mojtaba Ahmadih Khanesar

Copyright © 2022 Cai-Juan Soong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Binary tournament (BT) selection is known as an established selection operator that has been employed in various problems. However, in the development of evolutionary algorithms (EA), this selection operator has a drawback in providing an efficient implementation of the union procedure, which cannot guarantee a parsimonious knowledge base with reduced number of rules. Therefore, this paper introduces binary-standard deviation (SD) tournament selection into EA as an enhancement of BT that can lead to focus on more exploration in terms of searching for the best solutions. The proposed selection operator has been experimented within fish feed formulation in grouper fish farming as a case study on finding the minimum cost and fulfilling constraints. This approach is better than experimental design in terms of costs and time. The motivation for doing so is to search for alternative ingredients for the grouper fish, as the price of trash fish is too luxurious. It is because grouper fish are carnivorous and need many trash fish for better growth. The novelty of the proposed SD tournament selection is compared with BT selection in terms of searching for an efficient but not myopic algorithm. Hence, based on the comparative study, the findings of the enhanced selection operator towards the EA have been convinced and accepted in terms of better cost and fulfilling constraints requirements.

## 1. Introduction

Evolutionary algorithms (EA) are nature-inspired meta-heuristics [1–11]. The operators involved in EA are the initialization operator, selection operator, crossover operator, and mutation operator. The selection operator plays an important role in EA because it defines direction search, while other genetic operators suggest new search points in a nondirected way [12].

The selection operator can explore the search space efficiently and effectively. The role of selection is to distinguish among individuals based on their quality and, in particular, to allow the better individuals to become parents

of the next generation [13]; Eiben and Smith; 2015; [6–8, 10, 14–16]. The selection operator gives preference to better individuals for the next generation and prohibits the worst-fit individuals into the next generation [10, 16–18]. As the generation pass, the members of the population should get fitter and fitter [2, 4, 10, 15, 17, 19].

Since the selection operator has advantages in many aspects, many landscapes, especially in practical management applications, have increasingly drawn the attention of researchers from a variety of backgrounds. Of note, much of the research in this landscape has focused on the EA selection operator phase because of the need to combine vector performance gauges with the scalar method that existing EA

rewards independent capabilities. The earliest pioneering studies on selection operators emerge in the early 1990s [20, 21]. Subsequently, a few different selection operator employments were suggested from the years 1995 to 1996 [22–25]. After that, these different selection operator approaches were fruitfully practical to numerous problems [13, 18, 26–34].

For instance, Roulette wheel selection has been applied to berth scheduling problems [35, 36], shrimp diet formulation [8, 37], and vehicle routing problems [38, 39]. Meanwhile, tournament selection has also been used in berth scheduling problems [35, 40] and fish feed formulation [41, 42]. Whereas binary tournament (BT) selection has been deployed in berth scheduling problems [35], shrimp diet formulation [6, 37], vehicle routing problems [35, 39], and travelling salesman problem [43–45]. Consequently, this study aims to shed light on the performance of the selection operator to solve real-life grouper fish feed formulation problem as a case study for unlocking the frontier in selection operator in EA research on formulating the fish feed. The standard grouper fish feed formulation is an experimental design that involves many trial-and-error efforts, which is time-consuming. The feed formulation problem has been reported as a nondeterministic polynomial, NP-hard problem [46–48] due to its complex nature. It has been revealed that studies related to animal feed formulation modelling are generally still limited (Sahman et al., 2009 [49], and similar studies on aquaculture, including grouper, are even more limited [7, 42]). Therefore, it is essential to develop grouper feed formulations that prioritize necessary ingredients and nutrients. Hence, this study aims to develop a novelty selection operator in EA, applied in formulating the fish feed for grouper. Subsequently, the following section reviews the literature to highlight the advantages of EA and its operators. It is then followed by the methodology that discusses the application of fish feed formulation in EA. Experimental results and conclusions are drawn in the last section.

## 2. Literature Review

In the late 1990s, there was a relatively unknown identification of EA [19, 50–52]. These were due to unavailable platforms of powerful computers at that time [19, 53] and some weaknesses of early methodological [14, 19]. A few decades after emerging EA in the late 1950s [14, 19], this evolutionary computation attracted growing research interests in combinatorial optimization problem-solving [50, 54–58]. Intricate and complex optimization problems, including unavailable derivatives and multiple local optima, which are intractable by traditional mathematical methods [19, 58, 59], have been successfully applied by EA due to its strong search ability (adaptability) and easy implementation [1, 19, 56, 58, 60].

EA evolved from genetic algorithm (GA), evolution strategy (ES), evolutionary programming (EP), and genetic programming (GP) [6–8, 14, 19, 50, 56, 61]. GA was invented by John Holland in 1975, whereas Rechenberg and Schwefel developed ES in 1981. Lawrence J. Fogel was initiated in

1962, and lastly, GP was instigated by John Koza in 1994 [14, 56, 62–64]. Their process of solving problems from nature was so parallel that they eventually synthesized different methods in early 1990s [14, 19, 64], thus introducing EA and using EA in subsequent research [14, 60, 65–67].

Technically, EA claimed as metaheuristics since heuristics was incorporated in higher-level frameworks [1–8, 68]; Eiben and Smith; 2015; [9, 11, 14, 61]. There are slightly different between heuristics and metaheuristics. The characteristic of the heuristics optimization algorithm is to generate satisfactorily good solutions but does not guarantee optimal solutions [6–8, 61, 69–71], while metaheuristics can generate and guarantee near-optimal solutions to applicative optimization problems [61, 72–74]. Heuristic algorithms are designed to solve specific problems and must be well explained mathematically [61, 75, 76], while metaheuristic algorithms are designed to solve demanding NP-hard problems [10, 61, 77–80]. Normally, these NP-hard problems with complex real-world optimization are suggested to apply using metaheuristic algorithms [45, 61, 74, 81, 82].

The advantages of EA can be recognized in several aspects, likely population-based optimal solutions, robustness, machine learning, the feasibility of providing fast solutions, multiobjective optimisation and constraint handling [58, 60, 66, 83]. Furthermore, the most significant advantages of using EA are flexibility and adaptability [1, 19, 56, 58, 60]. Hence, EA is widely utilized due to EA provides practical advantages in solving complex combinatorial optimization problems [50, 54–58] and benefits of EA as discussed below.

First and foremost, the argument is that this approach is simple, robust to change circumstances, and flexible [1, 19, 56, 58, 60]. EAs are easy to apply and often provide satisfactory solutions compared with other global optimization techniques. According to Fogel [1]; Burke and Kendall [69]; Grosan and Abraham [60]; Rahman [6]; Abd Rahman et al. [7]; Rahman et al. [8]; Hien and Gillis [70]; Bashab et al. [71]; and Ojstersek et al. [61]; EAs can be applied to problems where heuristics solutions lead to unsatisfactory results due to EA has global search characteristics. Therefore, EAs have been widely used for practical problem-solving, even in feed formulation.

Second, Banzhaf et al. [2] are in similar judgment to Fogel [1] and Grosan and Abraham [60]; Simon [5] and Zhou et al. [58] that the aim of choosing EAs is EAs easy in solving combinatorial optimization problems or learning tasks in computer science compare with other conventional methods. It is because EA has two powerful elements of exploration and exploitation in the search process of EA. Thus, it can be concluded that EAs can solve combinatorial optimization problems as claimed by Fogel [1]; Schwefel [84]; Banzhaf et al. [2]; Grosan and Abraham [60]; Kumar and Singh [85]; Şahman et al. [86]; Simon [5]; Eiben and Smith (2015); and Zhou et al. [58].

Third, Schwefel [84]; Kumar and Singh [85]; Simon [5]; and Eiben and Smith (2015) criticized that EA can tackle complex problems such as discontinuities, multimodality, disjoint feasible spaces, and noisy function evaluations and reinforces the

potential effectiveness of EAs in search and optimization, which may be a problem where evolutionary computation differentiates itself from its competitors. Thus, it can be concluded that EAs can solve complex problems, as claimed by Schwefel [84]; Kumar and Singh [85]; Simon [5]; Rahman [6]; Rahman et al. [7]; Rahman et al. [8]; and Eiben and Smith (2015).

In conclusion, EA benefit over other methods in terms of searching lies in its flexibility and ability to adapt to existing tasks [1, 19, 56, 58, 60], global search characteristics [14, 19, 50], and robust performance [19, 56, 58, 60, 66, 83]. EA can be declared as a general adaptable concept to solve complex optimization problems rather than ready-to-use algorithms.

Due to the advantages of EA, EA is applied mainly in many landscapes for solving problems related to management, optimization, scheduling [87], and so forth [57]. Such problems are wherever applicable, like production and so on. The advantages of EA are due to the strength of the selection operator. In this selection stage, EA does not generate new chromosomes. The selection operator prohibits the worst-fit individuals into the next generation [10, 16–18]. The selection work is similar to the search path in the search space. In other words, deliberate parent selection is made by considering selection pressures. Logically, a higher selection intensity can be raised, which might be an advantage for a large population [14, 88, 89]. While in low diversity, high selection pressures can result in rapid convergence (rapid fall to local optimum).

In addition, considering their mating strategies, the question of how dissimilar the chosen parents are has been ignored. In other words, different pairs of parents have the potential to produce additional offspring than similar-looking parents, or vice versa. This is crucial to the exploration of the selection operator because it undermines the flexible reproduction operator, as numerous studies classify the preferred operator as an exploitation strategy [10, 14, 26, 32, 65, 90, 91]. Nevertheless, some studies ignore its importance; they copy a few individuals into a mating pool without a selection strategy [14, 32, 92, 93]. Indeed, the selection operator reacts to give space for permutation potential in the reproduction operator. Controlling or managing the search for diversity provides space to comprehend and improve selection operators.

Several researchers applied new different selection operators. Goldberg [20] applied the invention of the Boltzmann tournament selection in Pascal applications. His results have proven ready for practical to parallel hardware using this Boltzmann tournament selection approach. The Boltzmann tournament selection process was obtained and executed to provide a stable distribution crossways space and time in the population structures that proved close to Boltzmann. The Boltzmann tournament selection is applied using the concept of proportionate selection schemes. Distribution mechanism by imposing a group of individuals sharing restricted resources. The Boltzmann tournament selection executes by imposing individuals to contest their potential. The recovery mechanism of this final mechanism is unclear until this assessment of different individuals leads to more competition among undesirable individuals when selected randomly and uniformly from existing distributions.

Goldberg and Deb [21] conducted a comparative study on the performance of proportionate reproduction, ranking selection, BT selection, and genitor selection. Proportionate reproduction chooses a group of individuals for birth based on objective function values. While the perception of ranking selection is straightforward, where the population is sorted accordingly based on best to worst, performance the proportionate selection based on the assignment function given. The third idea of BT selection is simple; the best individual from a group of random individuals. Whereas, the last concept of genitor selection is difficult due to work one by one compared to concept generational, and the only one worst individual for replacement. BT selection is preferred for computation time due to the better timing complexity. In terms of growth ratio, genitor selection revealed a higher growth ratio than other selections over generations.

In years 1995 to 1996, Blicke and Thiele [22] as well as Blicke and Thiele [23] also conducted a similar comparative study with Goldberg and Deb [21] regards performance on fitness distributions for ranking selection operator, BT selection operator, truncation selection operator and lastly, exponential ranking selection operator. The idea of the first two selection operators is similar to the concept of Goldberg and Deb [21]. The latter concepts of a truncation selection operator and lastly an exponential ranking selection operator that are merely the fraction of best individuals are chosen while exponential ranking selection ranked weighted individuals exponentially. They claimed that the concept of this study is not new, but the finding of the ranking selection operator and BT selection operator is verified in the expected fitness distribution. Hence, the result enables understanding the single methodology aspects independently and isolated.

Chakraborty et al. [24] studied four types of selection pressure for population-elitist selection operator, linear ranking selection operator, BT Selection operator, fitness-proportionate selection operator, and genitor selection operator. They investigated the possible values that can be achieved, probably change of utmost importance, probably highest maximum value and lastly, probabilities take overtime distribution. The finding shows that the boundary for linear ranking selection operator, BT selection operator, and fitness-proportionate selection operator match genetic drift. Still, the genitor selection operator is remarkably higher than the three operators above. In conclusion, the genitor selection operator obtained the most arduous push among other selection operators. For the change of ultimate values, the probability of BT Selection operator and linear ranking selection operator is always less than one. In contrast, the item highest value linear ranking selection operator and BT selection operator behave similarly, which is 0.8 selection rate. While item probabilities take over time distribution for the linear ranking selection operator, the linear ranking selection operator, and BT selection operator are almost time-homogeneous.

Miller and Goldberg [25] conducted similar studies to Chakraborty et al. [24] to give practitioners a rational approach to the impact of different noise levels, including

human error, sampling error, and knowledge uncertainty which can affect the accuracy of fitness functions. Results found that proportionate selection never reaches absolute convergence. Tournament selection is the best selection because it can act faster to estimate the convergence time in small, medium, and large noisy environments.

Consequently, many studies explore tournament selection due to larger tournament size that can be used to increase competition between individuals and increased Pareto-approximation quality [32, 94, 95]. Moreover, BT selection is a better size option for selection. [21, 32]. Furthermore, BT selection makes perfect sense when solving unimodal problems [20, 32, 96]. In addition, BT selection with replacement is better in achieving the best solution quality with low computational time [20, 96–99]. Additionally, the correspondence of BT selection in the expected fitness distribution is proven [23, 32]. Likewise, the complexity of the tournament selection is lower than the complexity of other selections, such as the ranking selection [32, 100], and the selective pressure is higher, which allows us to measure whether each crossover can keep the population diversity [21, 32]. Thus, it can be revealed that BT selection is suitable because it is more efficient in time complexity. Due to the BT selection's strength, it is possible to explore this technique further. Thus, for practical management, it has sparked the interest to research on BT selection operator of EA since there is no work to include the standard deviation into BT selection to fill up the gap in fish feed formulation.

### 3. Materials and Methods

The method of EA has been discussed in those above and convinced us that BT selection could be appropriate guidance and help to unlock frontiers for solving fish feed formulation. The SD tournament selection is considered novel since, to the best of our knowledge, there is no work on this kind of computation for standard deviation into BT selection that has been reported for EA, especially in the field of feed formulation. BT selection is the component in the EA, and it is higher-order than heuristic; in other words, it means higher than the neighborhood. BT selection is embedded into this study, which is the key contribution to constructing an improved selection operator based on the BT selection and standard deviation. From the perspective of a selection procedure or operator, a tournament selection has been proven effective by Sahman et al. [5]; which was represented by real-valued alleles in the chromosome of the EA. Furthermore, this triggered utilizing the BT selection operator as Back et al. [19] and Hussain and Muhammad [32]. A possible improvement on the BT Selection operator considers the use of standard deviation in the tournament decision.

Before modeling the EA, information related to fish feed is collected. It included the background of the fish feed formulation and the fish feed formulation problems, which guided the direction and motivation of the study. The information on the composition of the nutrients in the contents of 100 kg feed ingredients used in the grouper diet

formulation applied in this research was taken from sources such as National Research Council [101], Holt [102]; Davis [103]; and Ali [104]. Meanwhile, the nutritional needs for the growth of the grouper fish feed formulation were taken from 30 manufacturers and experts in this field. The ingredients are chosen according to economic status, appropriateness for the digestive system of the grouper fish, and nutritional value needed based on experts' suggestions. Thus, the list of ingredients and its range, as suggested, is depicted in Table 1.

On top of ingredients, nutrients are also essential in feed formulation. The nutrient requirements needed by grouper fish are crude protein, crude fat, and crude fiber, which are precalculated with the tolerance limits carried out with one sample *T*-test hypothesis testing and Pearson correlation. The significance of these nutrients has been tested and analyzed using coefficient of variation (CV), where 16 nutrients elements, including moisture, were considered initially, as exhibited in Table 2. As a result, 15 nutrients are significant (i.e.,  $p < 0.05$ ) with the value of each respective CV is less than 1 (i.e.,  $CV < 1$ ). These nutrients are crude protein, crude fat, crude fiber, crude ash, phosphorus, calcium, arginine, histidine, isoleucine, leucine, lysine, methionine, phenylalanine, threonine, and valine. This research study does not consider moisture because its CV is greater than 1. The rationale is that the smaller CV value indicates that the other 15 nutrients are less dispersed than moisture. Therefore, the 15 nutrients are a good indication to be included in this research. Furthermore, these nutrients positively affect fish's growth in terms of weight (kg) or length (cm), as agreed by the consulted experts.

These listed nutrients are essential and crucial for the growth of the grouper fish. Therefore, they need to be included in this researcher's list of essential nutrients. This is based on the recommendation by Muhammadar et al. [105] and the experts. Subsequently, the final list of nutrients, together with its minimum and maximum percentages required in the formulation of grouper feed, is given in Table 3.

Hence, all of those grouper above feed information is stored in the database of a prototype. Then, the EA model was developed by vb.net, giving a user-friendly, extensible, and stretchable framework for tuning the metaheuristic-related parameters and increasing the quality solution. The purpose is to test the proposed solution's feasibility and the practicality of using enhanced EA in solving a real-world grouper fish feed formulation. Subsequently, the next section is to be discussed how the requirements and constraints are constructed into the modelling of EA.

### 4. Application of EA in Fish Feed Formulation

All requirements and constraints are modeled and applied to the fish feed formulation problem using the sophisticated EA method. The modeling of EA is developed based on the titles discussed in the following subsections.

*4.1. Objective Function.* The objective function or fitness function value is taken as the minimum summation of the

TABLE 1: List of ingredients and its ranges.

Ingredients	Min weight (kg)	Max weight (kg)
Algae meal (spirulina)	0	40
Cottonseed meal	0	40
Rice bran	0	40
Soybean meal	0	40
Wheat flour	0	40
Dried yeast	0	40
Meat and bone meal	0	100
Squid liver oil	0	100
Soybean oil	0	40
Shrimp meal	0	100
Local fishmeal	0	100
Imported fishmeal	0	100
Corn gluten meal	0	40
Poultry meal	0	100

TABLE 2: One sample *T*-test results in analyzing nutrients' significance of nutrients in grouper feed.

	One-sample <i>T</i> -Test						Coefficient of variation (CV)
	Test value = 0						
	T	Df	Sig. (2-Tailed)	Mean difference	95% confidence interval of the difference		
				Lower	Upper		
Crude protein	9.471	29	0.000	33.81000	26.5087	41.1113	0.578
Crude ash	10.540	12	0.000	13.56923	10.7643	16.3742	0.342
Crude fat	5.860	29	0.000	6.19033	4.0298	8.3509	0.935
Calcium	4.715	2	0.042	2.83333	.2478	5.4189	0.367
Crude fiber	7.661	29	0.000	2.96667	2.1747	3.7586	0.715
Moisture	4.529	28	0.000	29.02759	15.8987	42.1564	1.189
Phosphorus	5.017	10	0.001	0.90000	0.5003	1.2997	0.661
Histidine	13.980	9	0.000	0.95000	0.7963	1.1037	0.225
Isoleucine	16.812	9	0.000	2.03400	1.7603	2.3077	0.226
Leucine	16.292	9	0.000	3.07800	2.6506	3.5054	0.188
Lysine	14.801	9	0.000	2.75400	2.3331	3.1749	0.194
Methionine	15.395	9	0.000	1.26600	1.0800	1.4520	0.214
Phenylalanine	10.549	9	0.000	1.54800	1.2160	1.8800	0.205
Threonine	14.787	9	0.000	1.76800	1.4975	2.0385	0.300
Valine	17.715	9	0.000	2.14200	1.8685	2.4155	0.214
Arginine	14.074	9	0.000	2.89000	2.4255	3.3545	0.179

weight for each ingredient multiplied by its cost per kg, penalty value for the extra weight of ingredients and penalty value for nutrients that are out of its acceptable range, as formulated,

$$f(s_i) = \min \left( \sum_{j=1}^{j=J} C_j w_j \right) D_j + \left( \sum_{j=1}^{j=J} p_j \right) + \left( \sum_{k=1}^{k=K} z_k \right), \quad (1)$$

where  $s_i$  is the cumulative cost, nutrient penalty, penalty value for weight of  $j$ th ingredient for each chromosome,  $i$  or possible feed solution.  $Z_k$  is the penalty value for  $k$ th nutrient,  $D_j$  is the 100,000 constant number for cumulative cost,  $p_j$  is the penalty value for weight of  $j$ th ingredient not in the specified range.

**4.2. Constraints.** The main aim of this study is to prepare the fish feed formulation to meet nutrient needs. These nutrients needed the contribution to the constraints. There are mainly 15 nutrients needed for fish formulation. These 15 main nutrients constraints where the range of required percentage for  $k$ th nutrient in all suggested 14 ingredients is based on maximum and minimum values as presented in Table 3, where  $E_k \leq N_k \leq F_k$ ,  $E_k$  is the lower boundary of required percentage for  $k$ th nutrient in all ingredients,  $F_k$  is the upper boundary of required percentage for  $k$ th nutrient in all ingredients, and  $N_k$  is the range of required percentage for  $k$ th nutrient in all ingredients. The total weight of 100 kg must be fulfilled, and the range between individual animal-based ingredients and total plant-based ingredients should be 0–100 and 0–40, respectively. The range between total

TABLE 3: List of nutrients with its minimum and maximum required percentages.

Nutrients	Minimum (%)	Maximum (%)
Crude protein	40	45
Crude fat	1	10
Crude fiber	0.5	8
Crude ash	0.4	18
Phosphorus	0.1	1.8
Calcium	2	4
Arginine	2.06	4.21
Histidine	0.66	1.26
Isoleucine	1.37	2.57
Leucine	2.23	4.23
Lysine	1.96	4.04
Methionine	0.89	1.81
Phenylalanine	1.2	2.46
Threonine	1.29	2.59
Valine	1.46	2.86

animal-based ingredients and total plant-based ingredients should be 40–100 and 0–60, respectively.

**4.3. Penalty Values.** Penalty values are calculated for each ingredient wherever each of the total nutrients is not within its acceptable range (refer to Table 3). Penalty values for each nutrient are calculated based on Z-scores using the formula,  $Z_i = x_i - \mu_i / \sigma_i$ , where  $x_i$  is the observed value,  $\mu_i$  is the mean of the sample, and  $\sigma_i$  is the standard deviation of the sample.

The total weight of all ingredients is equal to 100 kg. Thus, another penalty which is ingredient weight penalty values also added to the total penalty. Apart from the constraints, the ingredient weight penalty is calculated based on the difference between the extra weight in kilogram (kg); the maximum required weight of the total weight of all ingredients. The ingredient weight penalty using formula (2)

$$\sum_{j=1}^{j=J} p_j, \quad (2)$$

where  $p_j$  is the penalty value for weight of  $j$  th ingredient not in the specified range.  $J=1, 2, \dots, J$  where  $J$  is the total number of ingredients considered.

**4.4. Costs.** The lowest cost of all ingredients was chosen in this study, where penalty values are zero or the lowest values. Costs are calculated by multiplying the weight for each ingredient multiply with its price value per kilogram of that feed. The cost of each ingredient is calculated using the following formula:

$$\sum_{j=1}^{j=J} C_j w_j, \quad (3)$$

where  $C_j$  is the cost of each ingredient  $j$  per kilogram.  $w_j$  is the weight of the  $j$  th ingredient in kilogram,  $J=1, 2, \dots, J$  where  $J$  is the total number of ingredients considered.

**4.5. Structure of Proposed EA.** Once the requirements and constraints are constructed into mathematical formulations, the modeling of EA is started by generating a population. The whole EA model is shown in Figure 1.

Creating the initial solution for this proposed SR-SD-EA is based on semirandom initialization (SR), whereby the total weight of 100 kg must be fulfilled, and the range between individual animal-based ingredients and total plant-based ingredients should be satisfied. The SD tournament selection used is similar to the BT selection operator, where the concept of the standard deviation of a sample is adapted in the BT selection. The pseudocode for the BT selection and SD tournament selection is displayed in Figures 2 and 3. Both figures clearly show that SD tournament selection incorporated the concept of standard deviation into BT selection, where the comparison between both pairs of chromosomes is made.

The process of EA continues with implementing the one-point crossover, boundary-based mutation and steady-state reproduction operators. The stopping criterion used in this SR-SD-EA is based on the number of generations. The final best-so-far chromosome or solution for the grouper fish feed formulation problem is achieved after a certain number of generations, the 200<sup>th</sup> generation. This new SD tournament selection operator shall be evaluated by comparing its performance with an existing similar selection operator, which is the BT selection operator. Consequently, the results of the proposed model are discussed in the following section.

## 5. Experimental Results

In this section, the experimentation results of the proposed SR-SD-EA models are discussed to ascertain the model's performance. The model was tested using semirandom (SR) initialization operator, one-point crossover, BM and steady-state operator in the regeneration procedure. In running the EAs, a specified number of generations,  $g=200$ , is used as the stopping criterion. This number is suggested since the best-so-far solutions for several preceding generations before  $g=200$  have shown a plateau trend. The best-so-far solutions or chromosomes obtained at each generation are taken as the best solution. Each operator was tested using several values to find the best parameter values to boost the EA performance. Eventually, each EA experimentation was carried out for 30 independent runs. Finally, the best parameter values obtained from various experiments are tabulated in Table 4, which is suggested for further analysis.

Based on the experimentation results, the population size of 90, crossover rate of 0.8 and mutation rate of 0.5 are used as the parameters in the model comparison of the EA discussed in the following subsections.

**5.1. Results of EA with a Proposed Selection Operator.** In this research, the proposed method SR-SD-EA in the fish formulation was tested on grouper fish selected. The performance of SR-SD-EA is presented in Figure 4.

In this Figure 4, at the 10<sup>th</sup> generation, the best-so-far fitness value for the SR-SD-EA with a population size of 90 is 1533.966. Subsequently, it dropped to 1176.128 in the 20<sup>th</sup>

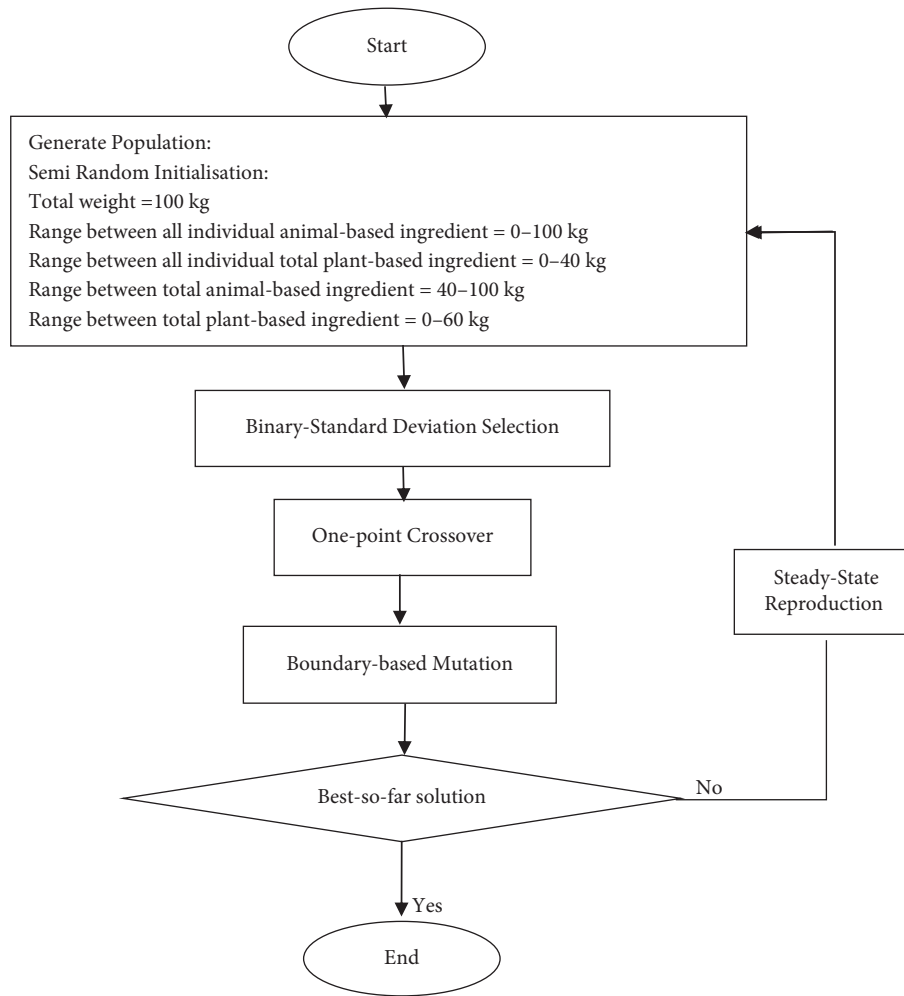


FIGURE 1: Structure of the proposed SD tournament selection.

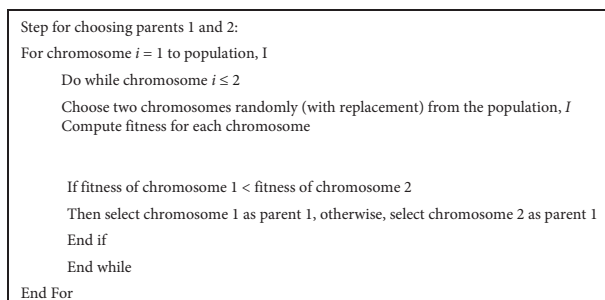


FIGURE 2: Pseudocode for binary tournament selection.

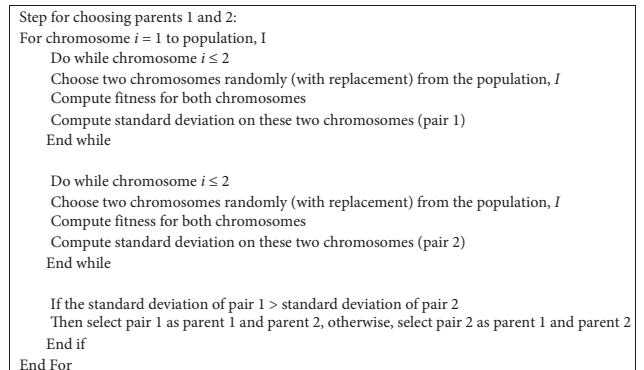


FIGURE 3: Pseudocode for binary-standard deviation tournament selection.

generation. In the 30<sup>th</sup> generation, the best-so-far fitness value continues decreasing to 764.663. There is a fluctuation of fitness values between the 40<sup>th</sup> generation and the 90<sup>th</sup> generation, which eventually reached the lowest value of 495.289 at the 100<sup>th</sup> generation and has shown a plateau trend with the same best-so-far fitness value until the 200<sup>th</sup> generation.

A sample solution with less than fourteen ingredients was selected with a total weight of 100 kg, as shown in Table 5. The nutrient value of each solution attained from the SR-SD-EA model is shown in Table 6.

It can be seen in Table 5 that ingredients for SR-SD-EA are algae meal (spirulina) = 0 kg, cottonseed meal = 1 kg, rice bran, = 16 kg, soybean meal = 37 kg, wheat flour = 0 kg, dried yeast = 4 kg, meat and bone meal = 0 kg, squid liver oil = 1 kg, soybean oil = 0 kg, shrimp meal = 11 kg, local fishmeal = 25 kg, imported fishmeal = 0 kg, corn gluten meal = 1 kg, and poultry meal = 4 kg.

TABLE 4: Parameter values for experimentation of EA models.

Operators	Value
Number of generation	200
Number of runs	30
Crossover rate	0.8
Mutation rate	0.5
Population size	90

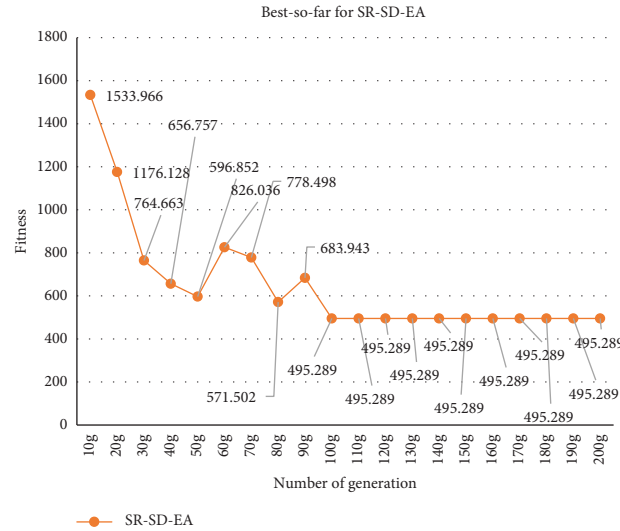


FIGURE 4: Results of EA with proposed SD tournament selection.

TABLE 5: A sample solution of the SR-SD-EA model.

Ingredients	Minimum weight (kg)	Maximum weight (kg)	SR-SD-EA (kg)
Algae meal (spirulina)	0	40	0
Cottonseed meal	0	40	1
Rice bran	0	40	16
Soybean meal	0	40	37
Wheat flour	0	40	0
Dried yeast	0	40	4
Meat and bone meal	0	100	0
Squid liver oil	0	100	1
Soybean oil	0	100	0
Shrimp meal	0	100	11
Local fishmeal	0	100	25
Imported fishmeal	0	100	0
Corn gluten meal	0	100	1
Poultry meal	0	100	4

TABLE 6: Nutrient value for SR-SD-EA.

Nutrients	Minimum (%)	Maximum (%)	SR-SD-EA (%)
Crude protein	40	45	42.763
Crude fat	1	10	5.732
Crude fiber	0.5	8	8.179
Crude ash	0.4	18	14.694
Phosphorus	0.1	1.8	1.099
Calcium	2	4	2.018
Arginine	2.06	4.21	2.762
Histidine	0.66	1.26	1.013
Isoleucine	1.37	2.57	1.846
Leucine	2.23	4.23	3.113
Lysine	1.96	4.04	2.772
Methionine	0.89	1.81	0.862
Phenylalanine	1.2	2.46	1.827
Threonine	1.29	2.59	1.897
Valine	1.46	2.86	2.171

lysine = 2.772%, methionine = 0.862%, phenylalanine = 1.827%, threonine = 1.897%, and valine = 2.171%.

It can be seen in Table 6 that nutrients for the SR-SD-EA model are crude protein = 42.763%, crude fat = 5.732%, crude fiber = 8.179%, crude ash = 14.694%, phosphorus = 1.099%, calcium = 2.018%, arginine = 2.762%, histidine = 1.013%, isoleucine = 1.846%, leucine = 3.113%,

5.2. Comparison of the Selection Operator. In this experiment, a modified selection operator based on the concepts of BT selection and the standard deviation is proposed and is known as the binary-standard deviation (SD) tournament selection operator. The description of this proposed SD



tournament selection operator has been discussed in section Materials and Methods. This unique SD tournament selection operator is incorporated in the whole process of an EA to evaluate its performance and compare it with the existing similar selection operator, that is, the BT selection operator, which is also incorporated in an EA. The BT selection operator has also been discussed in Materials and Methods. The resulting EAs in this experiment are the SR-SD-EA and SR-BT-EA.

For consistency purposes, the standard parameters to be used in the experimentation are population size of 90, crossover rate of 0.8, and mutation rate of 0.5, as decided in Table 4. These parameters are applied in both SR-SD-EA and SR-BT-EA for comparison purposes. Furthermore, SR-BT-EA and SR-SD-EA applied common operators, such as the SR initialization operator, one-point crossover, BM operator, and elitism strategy in the steady-state regeneration operator. A specified number of generations,  $g = 200$ , is used as the stopping criterion. This number is suggested since the best-so-far solutions for several preceding generations before  $g = 200$  have shown a plateau trend. The best-so-far solutions or chromosomes obtained at each generation are taken as the best solution. Eventually, each SR-BT-EA and SR-SD-EA was carried out for 30 independent runs for further experimentation and analysis.

In conclusion, for a fair selection operator comparison, two selection procedures were experimented with and thus suggesting the best selection operator to be used in the proposed model. The first procedure employed was the BT Selection operator, while the second was the SD tournament selection operator. The tournament selection operation was recommended by Sahman et al. [5] to find a potential solution in the selection stage. The proposed SD tournament selection operator is adopted based on the concept of statistical measurement. Standard deviation is a relevant and most-used measure of dispersion [106, 107]. For experimentation purposes, the EA model was implemented with the same operator as one-point crossover, boundary mutation, steady-state replacement strategy, and fixed parameters (e.g., 200 generations, 90 population size and 30 runs). In the model evaluation, comparison on the existing selection operator with the newly proposed operator has been made in terms of the best-so-far fitness, average best-so-far fitness, standard deviation, and average run time of the sample (milliseconds). Subsequently, Figure 5 presents the performance graph of the SR-BT-EA model and SR-SD-EA model.

In this Figure 5, the initial best-so-far fitness values for SR-SD-EA and SR-BT-EA are 1533.966 and 1626.068, respectively. Both SR-SD-EA and SR-BT-EA started to show a plateau trend at 100<sup>th</sup> generation. However, the SR-SD-EA has the lowest fitness value at 495.289, while the SR-BT-EA has the lowest fitness value at 507.359. A sample solution with fourteen ingredients from the SR-BT-EA model and SR-SD-EA were selected with the total weight of 100 kg shown in Table 7. Nutrient values of each solution yielded from the SR-BT-EA model and SR-SD-EA model are revealed in Table 8.

In the SR-SD-EA, only nine ingredients have been selected in the final feed mixed, where the ingredients that are not recommended are algae meal, wheat flour, meat and

bone meal, soybean oil, and imported fishmeal. While in the SR-BT-EA model, twelve ingredients have been selected in the final feed mixed, where the ingredients that are not recommended are algae meal and meat and bone meal. Both SR-SD-EA and SR-BT-EA satisfying constraints involved total weight of 100 kg, individual animal-based ingredients and total plant-based ingredients in the range of 0–100 and 0–40, respectively, as well as total animal-based ingredients and total plant-based ingredients in the range of 40–100 and 0–60, respectively.

The best-so-far feed formulation solution obtained from SR-SD-EA has successfully fulfilled all nutrients requirements. This is evidenced in Table 8 with all individual percentages of nutrients are within the minimum and maximum ranges.

Table 9 shows that SR-SD-EA was able to gain a fairly better result than model SR-BT-EA in terms of best so far to obtain cost minimization. Even though the standard deviation value of the best-so-far fitness for SR-SD-EA is fairly larger than that of the SR-BT-EA but, looking from the positive perspective it can be considered good since there is potential exploration to obtain much lower best-so-far fitness value. This is in line with Delmas and Liu [108] who emphasized that the larger standard deviation reflects that there would be a high chance of exploration that can take place, which leads to lower fitness. Furthermore, the average system run time of the SR-SD-EA needed approximately 479742.667 milliseconds which equivalent to 8.00 minutes of computation time, while the average system run time of the SR-BT-EA is 524842.600 milliseconds which equivalent to 8.75 minutes. Thus, based on this premise of better performance, the binary-standard deviation (SD) tournament Selection operator is a suitable and recommended to be applied in this feed formulation.

The results of *T*-test analysis for both models, SR-SD-EA and SR-BT-EA, are shown in Table 10 and Figure 6. The mean for SR-SD-EA is lower than SR-BT-EA, with the value of 1125.164 and 1142.131, respectively. However, the results of the *T*-test for the difference between two independent means shown in Figure 6 indicated that there is no significant difference between both means. In other words, both methods are equally effective. Based on the value of standard deviations in Table 10, the SR-SD-EA model is considered to have a fairly larger standard deviation than that of the SR-BT-EA. Looking from a positive perspective, it can be considered good since there is a potential exploration to obtain much lower best-so-far fitness value. This is in line with the results of the study by Delmas and Liu [108] that emphasized that the larger standard deviation reflects that there would be a high chance of exploration that can take place, which leads to lower fitness. This shows that SR-SD-EA provides more reliable methods that represent a real problem. On top of that, the standard error term includes all information, such as sample size and allele frequencies, thus providing optimal performance.

In conclusion, 95% confident that the true mean is between and 1125.164 and 1142.131. Although the true mean may or may not be in this interval, 95% of intervals formed in this manner will contain the true mean. Thus, based on

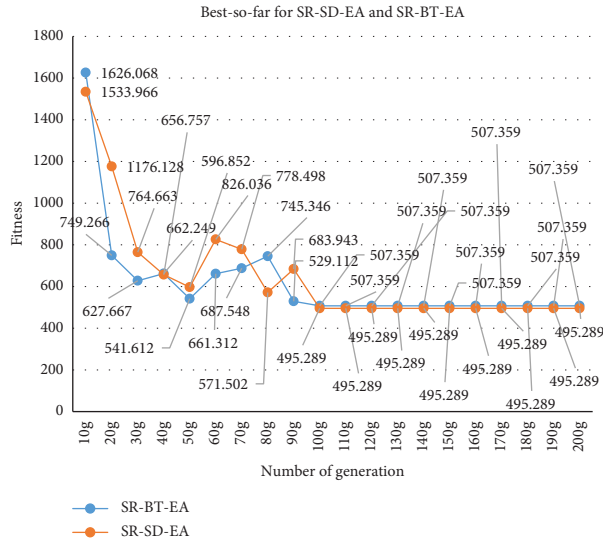


FIGURE 5: The performance graphs of SR-SD-EA and SR-BT-EA for comparison.

TABLE 7: A sample solution of the SR-BT-EA and SR-SD-EA models.

Ingredients	Minimum weight (kg)	Maximum weight (kg)	SR-BT-EA (kg)	SR-SD-EA (kg)
Algae meal (spirulina)	0	40	0	0
Cottonseed meal	0	40	2	1
Rice bran	0	40	17	16
Soybean meal	0	40	35	37
Wheat flour	0	40	1	0
Dried yeast	0	40	1	4
Meat and bone meal	0	100	0	0
Squid liver oil	0	100	1	1
Soybean oil	0	100	1	0
Shrimp meal	0	100	12	11
Local fishmeal	0	100	24	25
Imported fishmeal	0	100	1	0
Corn gluten meal	0	100	1	1
Poultry meal	0	100	4	4

TABLE 8: Nutrients value for the SR-SD-EA model.

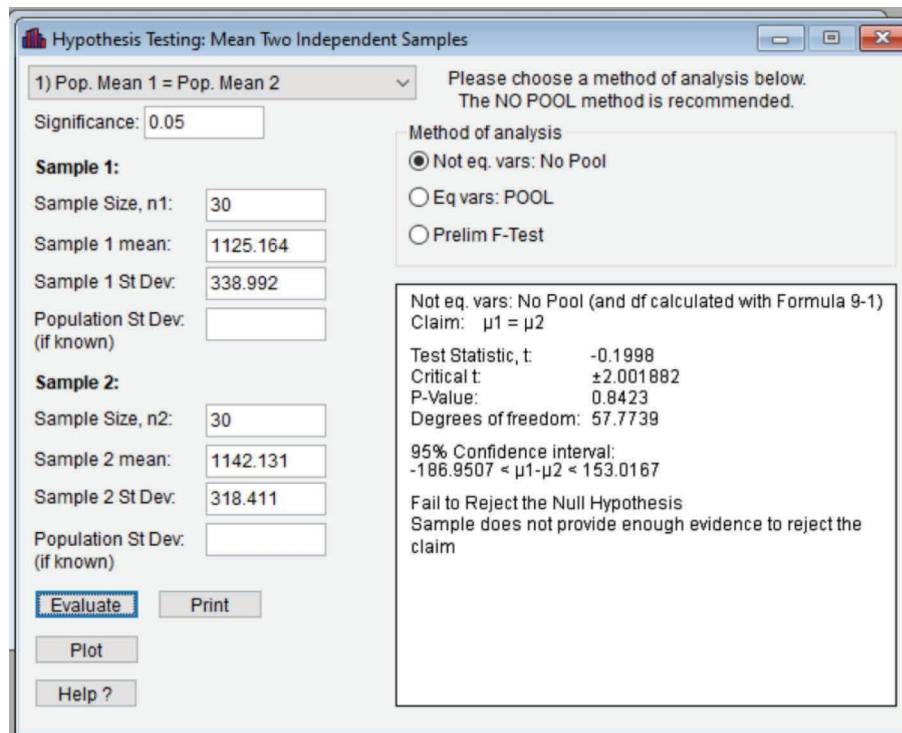
Nutrients	Minimum (%)	Maximum (%)	SR-BT-EA model (%)	SR-SD-EA model (%)
Crude protein	40	45	41.724	42.763
Crude fat	1	10	6.769	5.732
Crude fiber	0.5	8	8.37	8.179
Crude ash	0.4	18	14.922	14.694
Phosphorus	0.1	1.8	1.077	1.099
Calcium	2	4	2.063	2.018
Arginine	2.06	4.21	2.707	2.762
Histidine	0.66	1.26	0.981	1.013
Isoleucine	1.37	2.57	1.777	1.846
Leucine	2.23	4.23	3.014	3.113
Lysine	1.96	4.04	2.666	2.772
Methionine	0.89	1.81	0.846	0.862
Phenylalanine	1.2	2.46	1.774	1.827
Threonine	1.29	2.59	1.855	1.897
Valine	1.46	2.86	2.097	2.171

TABLE 9: Descriptive results for SR-SD-EA and SR-BT-EA.

Model	Best-so-far fitness	Average best-so-far fitness	Standard deviation	Average run time of the sample (milliseconds)
SR-SD-EA	495.289	1125.164	338.992	479742.667 (8.00 minutes)
SR-BT-EA	507.359	1142.131	318.411	524842.600 (8.75 minutes)

TABLE 10: Result of *T*-test analysis (one sample statistics) using SPSS.

Model	<i>N</i>	Mean	Standard deviation	Std. error mean
SR-SD-EA	30	1125.164	338.992	61.891
SR-BT-EA	30	1142.131	318.411	58.134

FIGURE 6: *T*-test for difference between two independent means.

this premise of better performance, the SR-SD-EA selection operator is a suitable and recommended to be applied in this feed formulation.

## 6. Conclusion

In conclusion, both the SR-BT-EA and the SR-SD-EA yield good and reasonable results in solving nonlinear cost optimization of a feed formulation problem. However, the performance of SR-SD-EA is better than the SR-BT-EA model. Furthermore, the SR-SD-EA model produces quicker results with same penalties 135.719 in the solution of nonlinear problems when compared with the SR-BT-EA model. This effort has convinced us that this selection operator has shown the potential of exploring and exploiting the function as the alternative solution, thus enhancing the method for complex grouper fish feed formulating.

Potentially, future works on other operators of the EA can be explored and consequently inaugurate a widespread research space for the growth and impact of the grouper fish feed formulation [22, 23, 32, 109].

## Data Availability

Data were collected via primary data.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

We are appreciative to have funding for the initial work in this study through FRGS/1/2013/STWN03/UUM/03/01 by

the MOHE and regulated by UUM. Much appreciation also to both Dr. Ahmad Daud bin Om from Fisheries Research Institute Tg. Demong, Terengganu and Dr. Zainuddin bin Jamari from Fisheries Research Institute Gelang Patah, Johor. We also thank those who have supported us in this research directly and or indirectly. We also thank the anonymous appraiser for valued viewpoints.

## References

- [1] D. B. Fogel, "The Advantages of Evolutionary Computation," in *Bio-computing and Emergent Computation*, D. Lundth, B. Olsson, and A. Naraganan, Eds., pp. 1–11, World Scientific Press, 1997.
- [2] W. Banzhaf, P. Nordin, R. E. Keller, and F. D. Francone, "Genetic programming: an introduction on the automatic evolution of computer programs and its applications," *dpunkt verlag and morgan kaufmann publishers, inc*, 1998.
- [3] E. G. Talbi, *Metaheuristics: From Design to Implementation*, John Wiley & Sons, New Jersey, 2009.
- [4] X. J. Yu and M. Gan, *Introduction to Evolutionary Algorithms*, Springer, Berlin Heidelberg New York, 2010.
- [5] D. Simon, *Evolutionary Optimization Algorithms: Biologically-Inspired and Population-Based Approaches to Computer Intelligence*, John Wiley & Sons, Inc, New Jersey, 2013.
- [6] A. R. Rahman, *Evolutionary algorithms with average crossover and power heuristics for aquaculture diet formulation. Unpublished doctoral thesis*, Universiti Utara Malaysia, Malaysia, 2014.
- [7] R. Abd Rahman, R. Ramli, Z. Jamari, and K. R. Ku-Mahamud, "Evolutionary algorithm with roulette-tournament selection for solving aquaculture diet formulation," *Mathematical Problems in Engineering*, vol. 2016, pp. 1–10, 2016.
- [8] A. R. Rahman, G. Kendall, R. Ramli, Z. Jamari, and K. R. Ku-Mahamud, "Shrimp feed formulation via evolutionary algorithm with power heuristics for handling constraints. Complexity," in *Introduction to Evolutionary Computing*, A. E. Eiben and J. E. Smith, Eds., Springer, Verlag Berlin Heidelberg, New York, 2017.
- [9] V. Nissen, "Applications of evolutionary algorithms to management problems," in *Innovative Research Methodologies in Management*, pp. 211–235, Palgrave Macmillan, Cham, 2018b.
- [10] T. Nakane, N. Bold, H. Sun, X. Lu, T. Akashi, and C. Zhang, "Application of evolutionary and swarm optimization in computer vision: a literature survey," *IPSI Transactions on Computer Vision and Applications*, vol. 12, no. 1, pp. 3–34, 2020.
- [11] K. Kumar and J. P. Davim, *Optimization Using Evolutionary Algorithms and Metaheuristics: Applications in Engineering*, CRC Press, Boca Raton, Florida, 2020.
- [12] B. T. Zhang and J. J. Kim, "Comparison of selection methods for evolutionary optimization," *Evolutionary Optimization*, vol. 2, no. 1, pp. 55–70, 2000.
- [13] K. Deb, "Introduction to Selection," in *Evolutionary Computation 1: Basic Algorithms and Operators*, T. Bäck, D. B. Fogel, and T. Michalewicz, Eds., pp. 166–171, Institute of Physics Publishing, Bristol, 2000.
- [14] H. T. Lim, "Enhanced Evolutionary algorithm with cuckoo search for nurse scheduling and rescheduling problem," *Bus driver scheduling system via genetic algorithm with enhanced crossover mechanism*, 2015.
- [15] L. M. Figueiredo, *An overview of the main machine learning models-from theory to algorithms. Unpublished Doctoral Thesis*, NOVA University Lisbon, Campolide, Lisbon, 2020.
- [16] P. Xu, W. Luo, X. Lin, and Y. Qiao, "Evolutionary continuous constrained optimization using random direction repair," *Information Sciences*, vol. 566, no. 1, pp. 80–102, 2021.
- [17] R. Sivaraj and T. Ravichandran, "A review of selection methods in Genetic Algorithm," *International Journal of Engineering Science and Technology*, vol. 3, no. 5, pp. 3792–3797, 2011.
- [18] R. Kumar and Jyotishree, "Novel encoding scheme in genetic algorithms for better fitness," *International Journal of Engineering and Advanced Technology*, vol. 1, no. 6, pp. 214–218, 2012.
- [19] T. Back, U. Hammel, and H. P. Schwefel, "Evolutionary computation: comments on the history and current state," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 3–17, 1997.
- [20] D. E. Goldberg, "A note on Boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing," *Complex Systems*, vol. 4, no. 4, pp. 445–460, 1990.
- [21] D. E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," *Foundations of Genetic Algorithms*, vol. 1, pp. 69–93, 1991.
- [22] T. Blicke and L. Thiele, "A comparison of selection schemes used in genetic algorithms," *Computer Engineering and Communication Networks Lab (TIK)-report*, 1995.
- [23] T. Blicke and L. Thiele, "A comparison of selection schemes used in evolutionary algorithms," *Evolutionary Computation*, vol. 4, no. 4, pp. 361–394, 1996.
- [24] U. K. Chakraborty, K. Deb, and M. Chakraborty, "Analysis of selection algorithms: a Markov chain approach," *Evolutionary Computation*, vol. 4, no. 2, pp. 133–167, 1996.
- [25] B. L. Miller and D. E. Goldberg, "Genetic algorithms, selection schemes, and the varying effects of noise," *Evolutionary Computation*, vol. 4, no. 2, pp. 113–131, 1996.
- [26] N. Veerapen, J. Maturana, and F. Saubion, "An exploration-exploitation compromise-based adaptive operator selection for local search. In," in *Proceedings of the 2012 Genetic and Evolutionary Computation Conference GECCO*, pp. 1277–1284, Newyork, NY, USA, 2012.
- [27] A. R. Rahman and R. Ramli, "Roulette-tournament selection for shrimp diet formulation problem," in *Proceedings of the 4th International Conference on Computing and Informatics (ICOCI)*, pp. 28–29, Sarawak Malaysia, August, 2013.
- [28] M. J. Craven and H. C. Jimbo, "An EA for portfolio selection over multiple investment periods with exponential transaction costs," in *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*, pp. 115–116, ACM, Newyork, NY, USA, 2013.
- [29] S. L. Yadav and A. Sohal, "Comparative study of different selection techniques in genetic algorithm," *International Journal of Energy Sector Management*, vol. 6, no. 3, pp. 1–7, 2017.
- [30] E. S. Ari and C. Gencer, "The use and comparison of A deterministic, A stochastic, and A hybrid multiple-criteria decision-making method for site selection of wind power plants: an application in Turkey," *Wind Engineering*, vol. 44, no. 1, pp. 60–74, 2020.
- [31] W. De Corte, P. R. Sackett, and F. Lievens, "Robustness, sensitivity, and sampling variability of Pareto-optimal selection system solutions to address the quality-diversity trade-off," *Organizational Research Methods*, vol. 23, no. 3, pp. 535–568, 2020.

- [32] A. Hussain and Y. S. Muhammad, "Trade-off between exploration and exploitation with genetic algorithm using a novel selection operator," *Complex & intelligent systems*, vol. 6, no. 1, pp. 1–14, 2020.
- [33] W. B. Owais, I. W. Alkhazendar, and D. Saleh, "Evaluating the impact of different types of crossover and selection methods on the convergence of 0/1 Knapsack using Genetic Algorithm," *arXiv preprint arXiv*, vol. 2010, Article ID 03483, 2020.
- [34] J. Li, H. Kang, G. Sun et al., "IBDA: improved binary dragonfly algorithm with evolutionary population dynamics and adaptive crossover for feature selection," *IEEE Access*, vol. 8, pp. 108032–108051, 2020.
- [35] M. A. Dulebenets, "A comprehensive evaluation of weak and strong mutation mechanisms in evolutionary algorithms for truck scheduling at cross-docking terminals," *IEEE Access*, vol. 6, pp. 65635–65650, 2018.
- [36] M. Kavooosi, M. A. Dulebenets, and M. Mikijeljević, "A universal island-based metaheuristic for effective berth scheduling," in *Proceedings of the XXIII international conference on material handling, constructions and logistics (MHCL 2019)*, Bar (Montenegro), 2019.
- [37] A. R. Rahman and R. Ramli, "Average concept of crossover operator in real coded genetic algorithm," *International Proceedings of Economics Development and Research (IPEDR)*, vol. 63, no. 15, pp. 73–77, 2013.
- [38] N. R. Sabar, A. Bhaskar, E. Chung, A. Turkey, and A. Song, "A self-adaptive evolutionary algorithm for dynamic vehicle routing problems with traffic congestion," *Swarm and Evolutionary Computation*, vol. 44, pp. 1018–1027, 2019.
- [39] J. Pasha, M. A. Dulebenets, M. Kavooosi, O. F. Abioye, H. Wang, and W. Guo, "An optimization model and solution algorithms for the vehicle routing problem with a "factory-in-a-box", " *IEEE Access*, vol. 8, pp. 134743–134763, 2020.
- [40] M. A. Dulebenets, "An adaptive island evolutionary algorithm for the berth scheduling problem," *Memetic Computing*, vol. 12, no. 1, pp. 51–72, 2020.
- [41] C.-J. Soong, R. Razamin, and A. R. Rosshairy, "A standard deviation selection in evolutionary algorithm for grouper fish feed formulation," in *Proceedings of the 4th international conference on quantitative sciences and its application (ICOQSIA)*, putrajaya Malaysia, august 2016.
- [42] C.-J. Soong, R. Razamin, and A. R. Rosshairy, "Potential grouper feed formulation based on evolutionary algorithm concept with A unique selection operator," *Journal of Engineering Science & Technology*, vol. 13, no. 2, pp. 332–346, 2018.
- [43] J. Gottlieb and T. Kruse, "Selection in evolutionary algorithms for the traveling salesman problem," in *Proceedings of the 2000 ACM symposium on Applied computing*, vol. 1, pp. 415–421, 2000.
- [44] Y. Shuai, S. Yunfeng, and Z. Kai, "An effective method for solving multiple travelling salesman problem based on NSGA-II. Systems Science & Control Engineering," *Systems Science & Control Engineering*, vol. 7, no. 2, pp. 108–116, 2019.
- [45] S. Zhou, L. Xing, X. Zheng, N. Du, L. Wang, and Q. Zhang, "A self-adaptive differential evolution algorithm for scheduling a single batch-processing machine with arbitrary job sizes and release times," *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1430–1442, 2021.
- [46] M. Florian, J. K. Lenstra, and A. H. G. Rinnooy Kan, "Deterministic production planning: algorithms and complexity," *Management Science*, vol. 26, no. 7, pp. 669–679, 1980.
- [47] S. Pathumnakul, M. Ittiphalin, K. Piewthongngam, and S. Rujikietkumjorn, "Should feed mills go beyond traditional least cost formulation?" *Computers and Electronics in Agriculture*, vol. 75, no. 2, pp. 243–249, 2011.
- [48] A. N. Sihananto, M. S. Burhan, and W. F. Mahmudy, "Chicken feed optimization using evolution strategies and firefly algorithm," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 1, pp. 585–592, 2019.
- [49] A. A. Altun and M. A. Sahman, "Cost optimization of mixed feeds with the particle swarm optimization method," *Neural Computing & Applications*, vol. 22, no. 2, pp. 383–390, 2013.
- [50] A. Hertz and D. Kobler, "A framework for the description of evolutionary algorithms," *European Journal of Operational Research*, vol. 126, pp. 1–12, 2000.
- [51] J. Bai, M. Tian, and J. Li, "An elite adaptive niche evolutionary algorithm for duty clustering problem in SoWSN," *Journal of Physics: Conference Series*, vol. 1769, no. 1, pp. 1–7, 2021.
- [52] H. Zhu, N. Luo, and X. Li, *A Quantum-Inspired Cuckoo Co-evolutionary Algorithm for No-Wait Flow Shop Scheduling IET Collaborative Intelligent Manufacturing*, John Wiley & Sons Ltd, New Jersey, 2021.
- [53] D. B. Fogel, *Evolutionary computation: towards a new philosophy of machine intelligence*, IEEE Press, New York, 1995.
- [54] D. B. Fogel, *Evolving artificial intelligence. Doctoral Thesis*, University of California, San Diego, La Jolla, 1992.
- [55] D. B. Fogel, "Applying evolutionary programming to selected traveling salesman problems," *Cybernetics & Systems*, vol. 24, no. 1, pp. 27–36, 1993.
- [56] N. M. A. Al-Salami, "Evolutionary algorithm definition," *American Journal of Engineering and Applied Sciences*, vol. 2, no. 4, pp. 789–795, 2009.
- [57] A. Slowik and H. Kwasnicka, "Evolutionary algorithms and their applications to engineering problems," *Neural Computing & Applications*, vol. 32, no. 16, pp. 12363–12379, 2020.
- [58] L. Zhou, L. Feng, A. Gupta, and Y. S. Ong, "Learnable evolutionary search across heterogeneous problems via kernelized autoencoding," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 3, pp. 567–581, 2021.
- [59] K. Deb, "Genetic Algorithm In Search And Optimization: The Technique And Applications," in *Proceedings of the international workshop on soft computing and intelligent systems*, pp. 58–87, 1998.
- [60] C. Grosan and A. Abraham, "Hybrid evolutionary algorithm: methodologies, architectures, and reviews," *Studies in Computational Intelligence (SCI)*, vol. 75, pp. 1–17, 2007.
- [61] R. Ojsterek, M. Brezocnik, and B. Buchmeister, "Multi-objective optimization of production scheduling with evolutionary computation: a review," *International Journal of Industrial Engineering Computations*, vol. 11, no. 3, pp. 359–376, 2020.
- [62] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 122–128, 1986.
- [63] X. Yao, Y. Liu, and G. M. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82–102, 1999.
- [64] D. Whitley, "An overview of evolutionary algorithms: practical issues and common pitfalls," *Information and Software Technology*, vol. 43, no. 14, pp. 817–831, 2001.
- [65] D. Ashlock, *Evolutionary Computation for Modeling and Optimization*, Springer, USA, 2005.

- [66] B. Maenhout and M. Vanhoucke, "An evolutionary approach for the nurse rostering problem," *Computers & Operations Research*, vol. 38, no. 10, pp. 1400–1411, 2011.
- [67] W. Zhang, W. Xu, and M. Gen, "Multi-objective evolutionary algorithm with strong convergence of multi-area for assembly line balancing problem with worker capability," *Procedia Computer Science*, vol. 20, pp. 83–89, 2013.
- [68] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: overview and conceptual comparison," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, 2003.
- [69] E. K. Burke and G. Kendall, "Introduction," in *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, E. K. Burke and G. Kendall, Eds., pp. 5–18, Springer, New York, 2005.
- [70] L. T. K. Hien and N. Gillis, "Algorithms for nonnegative matrix factorization with the Kullback-Leibler divergence," *arXiv*, vol. 2010, Article ID 01935, 2020.
- [71] A. Bashab, A. O. Ibrahim, E. E. AbedElgabar et al., "A systematic mapping study on solving university timetabling problems using meta-heuristic algorithms," *Neural Computing & Applications*, vol. 32, no. 23, pp. 17397–17432, 2020.
- [72] F. Zhao, R. Ma, and L. Wang, "A self-learning discrete jaya algorithm for multiobjective energy-efficient distributed no-idle flow-shop scheduling problem in heterogeneous factory system," *IEEE Transactions on Cybernetics*, pp. 1–12, 2021.
- [73] F. W. Glover and G. A. Kochenberger, *Handbook of Metaheuristics*, Springer Science & Business Media, New York, 2006.
- [74] H. Zhang, S. Liu, S. Moraca, and R. Ojstersek, "An effective use of hybrid metaheuristics algorithm for job shop scheduling problem," *International Journal of Simulation Modelling*, vol. 16, no. 4, pp. 644–657, 2016.
- [75] N. Siddique, *Intelligent Control: a hybrid approach based on fuzzy logic, neural networks and genetic algorithms*, Springer, New York, 2013.
- [76] S. Sundar, P. N. Suganthan, C. T. Jin, C. T. Xiang, and C. C. Soon, "A hybrid artificial bee colony algorithm for the job-shop scheduling problem with no-wait constraint," *Soft Computing*, vol. 21, no. 5, pp. 1193–1202, 2017.
- [77] J. q. Li, Q. k. Pan, and P. y Duan, "An improved artificial bee colony algorithm for solving hybrid flexible flowshop with dynamic operation skipping," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp. 1311–1324, 2016.
- [78] Y. Marinakis and M. Marinaki, "A hybrid particle swarm optimization algorithm for the open vehicle routing problem," in *Swarm Intelligence: 8th International Conference, ANTS 2012*, pp. 180–187, Springer, Brussels, 2012.
- [79] F. Goodarzian, H. Hosseini-Nasab, and M. B. Fakhrazad, "A multi-objective sustainable medicine supply chain network design using a novel hybrid multi-objective metaheuristic algorithm," *International Journal of Engineering*, vol. 33, pp. 1986–1995, 2020.
- [80] F. Goodarzian, D. Shishebori, H. Nasserli, and F. Dadvar, "A bi-objective production-distribution problem in a supply chain network under grey flexible conditions," *RAIRO - Operations Research*, vol. 55, pp. 1287–1316, 2021.
- [81] F. Zhao, L. Zhang, J. Cao, and J. Tang, "A cooperative water wave optimization algorithm with reinforcement learning for the distributed assembly no-idle flowshop scheduling problem," *Computers & Industrial Engineering*, vol. 153, p. 107082, 2021.
- [82] W. L. Liu, Y. J. Gong, W. N. Chen, Z. Liu, H. Wang, and J. Zhang, "Coordinated charging scheduling of electric vehicles: a mixed-variable differential evolution approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 12, pp. 5094–5109, 2020.
- [83] R. Ramli, *An evolutionary algorithm for the nurse scheduling problem with circadian rhythms. Unpublished doctoral thesis*, Universiti Sains Malaysia, Malaysia, 2004.
- [84] H. P. Schwefel, "Advantages (and disadvantages) of evolutionary computation over other approaches," in *Handbook Of Evolutionary Computation*, K. De-Jong, L. Fogel, and H. P. Schwefel, Eds., pp. 1–10, IOP Publishing Ltd and Oxford University Press, England, 1997.
- [85] R. Kumar and P. K. Singh, "Pareto evolutionary algorithm hybridized with local search for biobjective TSP," *Studies in Computational International (SCI)*, vol. 75, pp. 361–398, 2007.
- [86] M. Akif Şahman, M. Cunkas, S. Inal, F. Inal, B. Coskun, and U. Taskiran, "Cost optimization of feed mixes by genetic algorithms," *Advances in Engineering Software*, vol. 40, no. 10, pp. 965–974, 2009.
- [87] F. Zhao, X. He, and L. Wang, "A two-stage cooperative evolutionary algorithm with problem-specific knowledge for energy-efficient scheduling of no-wait flow-shop problem," *IEEE Transactions on Cybernetics*, vol. 51, no. 11, pp. 5291–5303, 2021.
- [88] S. Legg, M. Hutter, and A. Kumar, "Tournament versus Fitness Uniform Selection In," in *Proceedings of the 2004 congress on evolutionary computation (CEC '04)*, pp. 2144–2151, IEEE, Portland, OR, 2004.
- [89] F. Saeed, F. Mohammed, and N. Gazem, "Emerging trends in intelligent computing and informatics: data science," *intelligent information systems and smart computing*, vol. 1073), 2019.
- [90] A. Al-Naqi, A. T. Erdogan, and T. Arslan, "Fault Tolerance through Automatic Cell Isolation Using Three-Dimensional Cellular Genetic Algorithms In," in *Proceedings of the 2010 IEEE congress on evolutionary computation (CEC)*, pp. 1–8, IEEE, New York, 2010.
- [91] S. Ramachandra and V. Chavan, "Enhanced-Hybrid-age layered population structure (E-Hybrid-ALPS): a genetic algorithm with adaptive crossover for molecular docking studies of drug discovery process," *International Journal of Current Research and Review*, vol. 12, no. 15, pp. 07–15, 2020.
- [92] C. C. Tsai and S. H. Li, "A two-stage modeling with genetic algorithms for the nurse scheduling problem," *Expert Systems with Applications*, vol. 36, no. 5, pp. 9506–9512, 2009.
- [93] F. C. Yang and W. T. Wu, "A genetic algorithm-based method for creating impartial work schedules for nurses," *International Journal of Electronic Business Management*, vol. 10, no. 3, pp. 182–193, 2012.
- [94] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," in *Proceedings of the first IEEE conference on evolutionary computation. IEEE world congress on computational intelligence*, pp. 82–87, 1994.
- [95] D. Srinivasan and L. Rachmawati, "An efficient multi-objective evolutionary algorithm with steady-state replacement model," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 715–722, 2006.
- [96] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.
- [97] G. R. Harik, "Finding multimodal solutions using restricted tournament selection," in *Proceedings of the 6th International Conference on Genetic Algorithms*, pp. 24–31, 1995.

- [98] N. M. Razali and J. Geraghty, "Genetic algorithm performance with different selection strategies in solving tsp," in *Proceedings of the World Congress on Engineering WCE*, London, UK, 2011.
- [99] S. Prayudani, A. Hizriadi, E. B. Nababan, and S. Suwilo, "Analysis effect of tournament selection on genetic algorithm performance in traveling salesman problem (TSP)," *Journal of Physics: conference Series*, vol. 1566, no. 1, p. 012131, 2020.
- [100] J. H. Back, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, Oxford, 1996.
- [101] NRC, *Nutrient Requirements of Fish and Shrimp*. Board on Agriculture and Natural Resources, National Research Council, 2011.
- [102] G. J. Holt, *Larval Fish Nutrition*, John Wiley & Sons, Inc, USA, 2011.
- [103] D. A. Davis, *Feed and Feeding Practices in Aquaculture*, Elsevier, USA, 2015.
- [104] S. A. Ali, *Nutritional Feeding of Fish and Shrimps in India*, MJP Publisher, India, 2019.
- [105] A. A. Muhammadar, A. G. Mazlan, A. Samat, Z. A. Muchlin, and K. D. Simon, "Crude protein and amino acids content in some common feeds of tiger grouper (*Epinephelus fuscoguttatus*) juvenile," *Aquaculture, Aquarium, Conservation & Legislation International Journal of the Bioflux Society*, vol. 4, no. 4, pp. 499–504, 2011.
- [106] P. S. Mann, *Introductory Statistics*, John Wiley & Sons Inc, New Jersey, 2013.
- [107] P. S. Mann, *Introductory Statistics*, John Wiley & Sons Inc, New Jersey, 2016.
- [108] R. Delmas and Y. Liu, "Exploring students' conceptions of the standard deviation," *Statistics Education Research Journal*, vol. 4, no. 1, pp. 55–82, 2005.
- [109] F. Zhao, S. Di, J. Cao, J. Tang, and Jonrinaldi, "A novel cooperative multi-stage hyper-heuristic for combination optimization problems," *Complex System Modeling and Simulation*, vol. 1, no. 2, pp. 91–108, 2021.