Fish School Search

Carmelo J.A. Bastos Filho, Fernando B. de Lima Neto, Anthony J.C.C. Lins, Antônio I.S. Nascimento, and Marília P. Lima

Abstract. Real world problems are packed with complex issues often hard to be computed. Searching for parameters or candidate solutions is frequently associated with these complexities. The reason for that is chiefly related to the large dimensionalities of some search spaces. In general, problems involving large search spaces use traditional computer intensive methods that are, quite often, expensive (i.e. resource consuming). Nature-inspired algorithms, on the other hand, are able to deal reasonably well with the abovementioned difficulties. In this chapter, we provide an overview of a novel approach for searching in high-dimensional spaces based on the behaviors of fish schools. As any other intelligent technique based on population, Fish School Search (FSS) greatly benefits from the collective emerging behavior that increases mutual survivability. Broadly speaking, FSS is composed of operators that can be grouped in the following categories: feeding, swimming and breeding. Together, these operators provide computing behavior such as: (i) high-dimensional search ability, (ii) automatic selection between exploration and exploitation, and (iii) self-adaptable guidance towards sought solutions. This chapter seeks to explain the main ideas behind FSS to researchers and practitioners. In addition, we include examples and simulations aimed at clarifying the simplicity and potentials of FSS.

1 Introduction

Several oceanic fish species, as with other animals, present social behavior. This phenomenon's main purpose is to increase mutual survivability and may be viewed in two ways: (i) for mutual protection and (ii) for synergistic achievement of other collective tasks. By protection we mean reducing the chances of being caught by predators; and by synergy, we refer to an active means of achieving collective goals such as finding food.

Carmelo J.A. Bastos Filho · Fernando B. de Lima Neto · Anthony J.C.C. Lins · Antônio I.S. Nascimento · Marília P. Lima
Department of Computing and Systems, Polytechnic School of Pernambuco, University of Pernambuco (UPE), Recife, Brazil
e-mail: {cjabf,fbln,ajccl,aisn,mpl}@dsc.upe.br

R. Chiong (Ed.): Nature-Inspired Algorithms for Optimisation, SCI 193, pp. 261–277.
springerlink.com

Apart from debating whether the emergent behavior of a fish school is due to learning or genetic reasons, it is important to note that some fish species live their entire lives in schools. This reduces individual freedom in terms of swimming movements and increases competition in regions with scarce food. However, fish aggregation is a fact and the benefits largely outweigh the drawbacks. This chapter aims at presenting a novel computational intelligent search technique inspired by the abovementioned behavior.

Along with the development of this technique we have taken great care not to depart from the original inspiration source, but FSS contains a few abstractions and simplifications that have been introduced to afford efficiency and usability to the algorithm. The main characteristics derived from real fish schools and incorporated into the core of our approach are sound. They are grouped into two observable categories of behaviors as follows:

- Feeding: inspired by the natural instinct of individuals (fish) to find food in order to grow strong and to be able to breed. Notice that food here is a metaphor for the evaluation of candidate solutions in the search process. We have considered that an individual fish can lose as well as obtain weight, depending on the regions it swims in;
- Swimming: the most elaborate observable behavior utilized in our approach. It aims at mimicking the coordinated and the only apparent collective movement produced by all the fish in the school. Swimming is primarily driven by feeding needs and, in the algorithm, it is a metaphor for the search process itself.

2 Background

2.1 Search Problems and Algorithms

Although there are several approaches for searching, there is, unfortunately, no general optimal search strategy [1]. Thus, solving search problems is sometimes more of an art form than an engineering practice. Although custom-made algorithms are valuable options for specific problems, a more generalized automatic search engine would be a great bonus for tackling problems of high dimensionality.

Search problems can be highly varied. For example, they can be classified into two groups with regard to the structure of their search-space: structured or unstructured. For the former, there are many traditional techniques that are, on average, quite efficient. The same observation does not apply to the latter, that is, there is no overall good approach for search spaces on which there is no prior information.

We think that FSS might be a valuable option for searching in high dimensional and unstructured spaces.

2.2 Population-Based Algorithm (PBA)

Many nature-inspired algorithms such as genetic algorithms (GA) [2, 3], artificial immune systems (AIS) [4], ant colony optimization (ACO) [5, 6] and particle swarm optimization (PSO) [7, 8, 9] are based on the concept of populations. In all these approaches, the computing discrimination power and memorization ability of past experiences are distributed among the individuals of the population in varying degrees.

Distributed representation and computation are interesting features to incorporate into search algorithms because of the parallelization they provide. The obvious trade-off is the cost of control (*i.e.* communication among individuals), as opposed to the lower costs associated with centralized control.

In recent years, PSO has produced good results for search problems with high dimensionality. It is an intelligent computational technique proposed by Kennedy and Eberhart in 1995 [7]. This technique is commonly used to solve optimization problems of nonlinear functions. It is inspired by the social behavior of bird flocks. The idea behind PSO is to create particles that simulate the movements of birds to achieve a specific goal within the search space. It explores the social behavior of an organized group of individuals and the group's communication capacity. Each particle represents a solution in a high-dimensional space. The entire swarm uses a specific communication mechanism. The candidate solutions emerge by flocking behavior around more successful individuals. Particles in PSO utilize the notion of adjustable speed according to the degree of success achieved. In the most common PSO implementations, particles move through the search space using a combination of the attraction to the best solution that they have found individually, and the attraction to the best solution that any particle in the neighborhood has found. A neighborhood is the part of the swarm which a particle is able to communicate with. Bratton *et al.* [9] proposed a standard for performance comparison of PSO implementations. Many velocity equations and communication mechanisms were proposed in recent years [10, 11, 12, 13, 14]. However, the PSO technique struggles in some multimodal problems.

3 Fish-School Search (FSS)

3.1 FSS Computational Principles

The search process in FSS is carried out by a population of limited-memory individuals – the fish. Each fish represents a possible solution to the problem. Similar to PSO or GA, search guidance in FSS is driven by the success of some individual members of the population.

The main feature of the FSS paradigm is that all fish contain an innate memory of their successes – their weights. In comparison to PSO, this information is highly relevant because it can obviate the need to keep a log of the best positions visited by all individuals, their velocities and other competitive global variables.

Another major feature of FSS is the idea of evolution through a combination of some collective swimming, *i.e.* "operators" that select among different modes of operation during the search process, on the basis of instantaneous results.

As for dealing with the high dimensionality and lack of structure of the search space, the authors believe that FSS should at least incorporate principles such as the following:

(i) Simple computation in all individuals;
(ii) Various means of storing distributed memory of past computation;
(iii) Local computation (preferably within small radiuses);
(iv) Low communication between neighboring individuals;
(v) Minimum centralized control (preferably none); and
(vi) Some diversity among individuals.

A brief rationale for the above-mentioned principles is given, respectively: (i) this reduces the overall computation cost of the search; (ii) this allows for adaptive learning; (iii), (iv) and (v) these keep computation costs low as well as allowing some local knowledge to be shared, thereby speeding up convergence; and finally, (vi) this might also speed up the search due to the differentiation/specialization of individuals. These principles incorporated in FSS lead the authors to believe that FSS can deal with multimodal problems better than the PSO approaches.

3.2 Overview of the New Approach

The inspiration mentioned in Section I, together with the principles just stated above, are incorporated in our approach in the form of two operators that comprise the main routines of the FSS algorithm. To understand the operators, a number of concepts need to be defined.

The concept of food is related to the function to be optimized in the process. For example, in a minimization problem the amount of food in a region is inversely proportional to the function evaluation in this region. The "aquarium" is defined by the delimited region in the search space where the fish can be positioned.

The operators are grouped in the same manner in which they were observed when drawn from the fish school. They are as follows:

- Feeding: food is a metaphor for indicating to the fish the regions of the aquarium that are likely to be good spots for the search process;
- Swimming: a collection of operators that are responsible for guiding the search effort globally towards subspaces of the aquarium that are collectively sensed by all individual fish as more promising with regard to the search process.

3.3 FSS Operators

3.3.1 The Feeding Operator

As in real situations, the fish of FSS are attracted to food scattered in the aquarium in various concentrations. In order to find greater amounts of food, the fish in the school can move independently (see individual movements in the next section).

As a result, each fish can grow or diminish in weight, depending on its success or failure in obtaining food. We propose that fish's weight variation be proportional to the normalized difference between the evaluation of fitness function of previous and current fish position with regard to food concentration of these spots. The assessment of 'food' concentration considers all problem dimensions, as shown in 1,

$$W_i(t+1) = W_i(t) + \frac{f[x_i(t+1)] - f[x_i(t)]}{\max\{|f[x_i(t+1)] - f[x_i(t)]|\}}, \tag{1}$$

where $W_i(t)$ is the weight of the fish i, $\mathbf{x}_i(t)$ is the position of the fish i and $f[\mathbf{x}_i(t)]$ evaluates the fitness function (*i.e.* amount of food) in $\mathbf{x}_i(t)$.

A few additional measures were included to ensure rapid convergence toward rich areas of the aquarium, namely:

- Fish weight variation is evaluated once at every FSS cycle;
- An additional parameter, named weight scale (W_{scale}) was created to limit the weight of a fish. The fish weight can vary between "1" and W_{scale}.
- All the fish are born with weight equal to $\frac{W_{scale}}{2}$.

3.3.2 The Swimming Operators

A basic animal instinct is to react to environmental stimulation (or sometimes, the lack of it). In our approach swimming is considered to be an elaborate form of reaction regarding survivability. In FSS, the swimming patterns of the fish school are the result of a combination of three different causes (*i.e.* movements).

For fish, swimming is directly related to all the important individual and collective behaviors such as feeding, breeding, escaping from predators, moving to more livable regions of the aquarium or, simply being gregarious.

This panoply of motivations to swim away inspired us to group causes of swimming into three classes: (i) individual, (ii) collective-instinct and (iii) collective-volition. Below we provide further explanations on how computations are performed on each of them.

3.3.2.1 Individual Movement

Individual movement occurs for each fish in the aquarium at every cycle of the FSS algorithm. The swim direction is randomly chosen. Provided the candidate destination point lies within the aquarium boundaries, the fish assesses whether the food density there seems to be better than at its current location. If this is not the case or if the step-size is not possible (i.e. it lies outside the aquarium or is blocked by, say, reefs), the individual movement of the fish does not occur. Soon after each individual movement, feeding occurs, as detailed above.

For this movement, we define a parameter to determine the fish displacement in the aquarium called individual step ($step_{ind}$). Each fish moves $step_{ind}$ if the new position has more food than the previous position. Actually, to include more randomness in the search process we multiply the individual step by a random number generated by a uniform distribution in the interval [0,1]. In our simulation we decrease the individual step linearly in order to provide exploitation abilities in later iterations.

Fig. 1 shows an illustrative example of this swimming operator. One can note that just the fish that found spots with more food had moved.

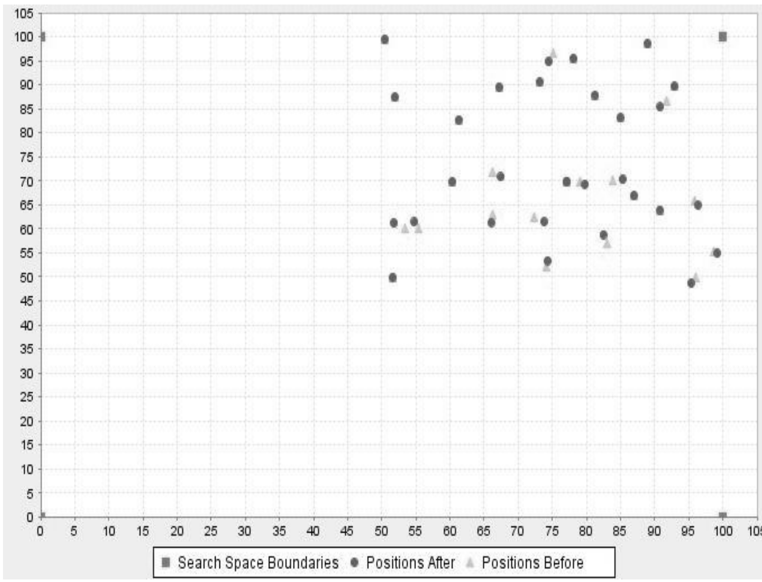

Fig. 1 Individual movement is illustrated here before and after its occurrence; circular dots are fish positions after and triangular dots are the same fish before individual movement

3.3.2.2 Collective-Instinctive Movement

After all fish have moved individually, a weighted average of individual movements based on the instantaneous success of all fish of the school is computed. This means that fish that had successful individual movements influence the resulting direction of movement more than the unsuccessful ones. When the overall direction is computed, each fish is repositioned. This movement is based on the fitness evaluation enhancement achieved, as shown in 2.

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \frac{\sum_{i=1}^{N} \Delta\mathbf{x}_{indi}\,\{f[x_i(t+1)] - f[x_i(t)]\}}{\sum_{i=1}^{N}\{f[x_i(t+1)] - f[x_i(t)]\}} \tag{2}$$

where $\Delta\mathbf{x}_{indi}$ is the displacement of the fish i due to the individual movement in the FSS cycle.

Fig. 2 shows the influence of the collective-instinctive movement in the example presented in Fig. 1. One can note that in this case all the fish had their positions adjusted.

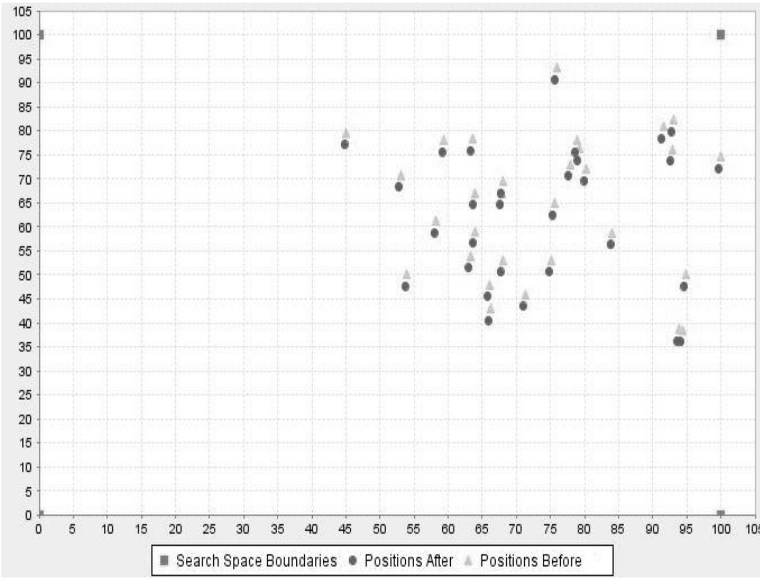

Fig. 2 Collective-instinctive movement is illustrated here before and after its occurrence; circular dots are fish positions after and triangular dots are the same fish before collective-instinctive movement

3.3.2.3 Collective-Volitive Movement

After individual and collective-instinctive movements are performed, one additional positional adjustment is still necessary for all fish in the school: the collective-volitive movement. This movement is devised as an overall success/failure evaluation based on the incremental weight variation of the whole fish school. In other words, this last movement will be based on the overall performance of the fish school.

The rationale is as follows: if the fish school is putting on weight (meaning the search has been successful), the radius of the school should contract; if not, it should dilate. This operator is deemed to help greatly in enhancing the exploration abilities in FSS. This phenomenon might also occur in real swarms, but the reasons are as yet unknown.

The fish-school dilation or contraction is applied as a small step drift to every fish position with regard to the school's barycenter. The fish-school's barycenter is obtained by considering all fish positions and their weights, as shown in 3.

Collective-volitive movement will be inwards or outwards (in relation to the fish-school's barycenter), according to whether the previously recorded overall weight of the school has increased or decreased in relation to the new overall weight observed at the end of the current FSS cycle.

$$Bari(t) = \frac{\sum_{i=1}^{N} \mathbf{x}_i(t) W_i(t)}{\sum_{i=1}^{N} W_i(t)} \tag{3}$$

For this movement, we also define a parameter called volitive step ($step_{vol}$). We evaluate the new position as in 4 if the overall weight of the school increases in the FSS cycle; if the overall weight decreases, we use 5.

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) - step_{vol}.rand.\left[\mathbf{x}_i(t) - Bari(t)\right], \tag{4}$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + step_{vol}.rand.\left[\mathbf{x}_i(t) - Bari(t)\right], \tag{5}$$

where *rand* is a random number uniformly generated in the interval [0,1]. We also decreased the linear $step_{vol}$ along the iterations.

Fig. 3 shows the influence of the collective-volitive movement in the example presented in Fig. 1 after individual and collective-instintive movements. In this case, as the overall weight of the school had increased, the radius of the school diminished.

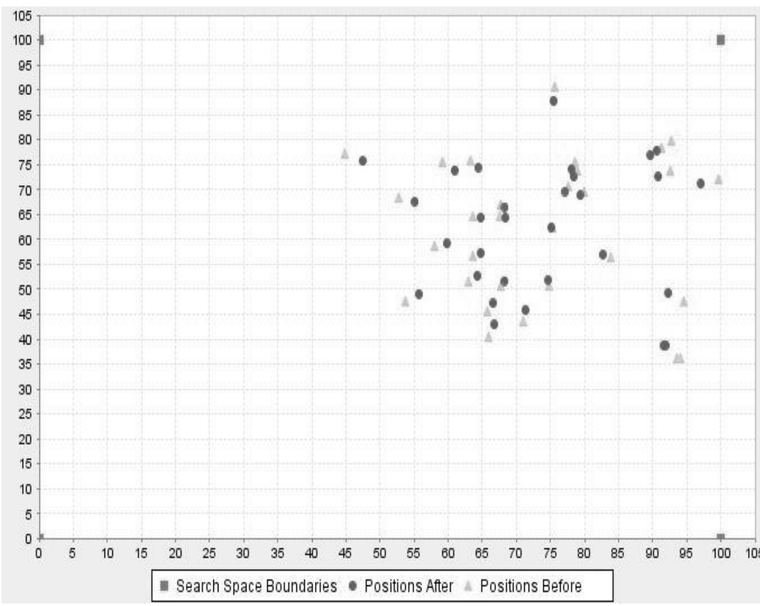

Fig. 3 Collective-volitive movement is illustrated here before and after its occurrence; circular dots are fish positions after and triangular dots are the same fish before collective-volitive movement

3.4 FSS Cycle and Stop Conditions

The FSS algorithm starts by randomly generating a fish school according to parameters that control fish sizes and their initial positions.

Regarding dynamics, the central idea of FSS is that all bio-inspired operators perform independently of each other across the three conceived classes.

The search process (*i.e.* FSS at work) is enclosed in a loop, where invocations of the previously presented operators will occur until at least one stop condition is met.

As of now, stop conditions conceived for FSS are as follows: limitation of the number of cycles (the stopping condition of all experiments in this chapter), time limit, maximum school radius, minimum school weight and maximum fish number.

We present below the pseudo-code for the Fish School Search Algorithm. In the initialization step, each fish in the swarm has its weight initialized with the value $\frac{W_{scale}}{2}$ and its position in each dimension initialized randomly in the search space.

Algorithm Fish School Search

1. Initialize fish in the swarm
2. **While** maximum iterations or stop criteria is not attained **do**
3. **for** each fish *i* in the swarm **do**
   a. **update position applying the individual operator**

$$\Delta \mathbf{x}_i(t+1) = step_{\text{ind}}(t) \cdot 2 \cdot rand \cdot direction$$

$$\overrightarrow{temp}_i = \mathbf{x}_i(t) + \Delta \mathbf{x}_i(t+1)$$

calculate fish fitness $f_i(\overrightarrow{temp}_i)$
if $f(\overrightarrow{temp}_i) < f(\mathbf{x}_i(t))$

$$\mathbf{x}_i(t+1) = \overrightarrow{temp}_i$$

$$f_i^{(t+1)} = f_i(\overrightarrow{temp}_i)$$

else

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t)$$

$$f_i^{(t+1)} = f_i^{(t)}$$

   b. **apply feeding operator**
   update fish weight according to 1
   c. **apply collective-instinctive movement**
   update fish position according to 2
   d. **apply collective-volitive movement**
   if overall weight of the school increases in the cycle
   update fish position using 4
   elseif overall weight of the school decreases in the cycle
   update fish position using 5

end for decrease the individual and volitive steps linearly

end while

4 Illustrative Example

This section presents an illustrative example aimed at better understanding how FSS can be used and, ultimately, how it works. The selected example considers a small school and a very simple problem that is three fish are set to find the global optimum of the sphere function in two dimensions. The sphere function is presented in 6 and its parameters are: (i) feasible space [-10,10], (ii) number of iterations equal to 10, (iii) w_{scale}= 10, (iv) initial $step_{ind}$ = 1, (v) final $step_{ind}$ = 0.1, (vi) initial $step_{vol}$ = 0.5, (vii) final $step_{vol}$ = 0.05. Table 1 includes initial values associated with the experimental fish school; Fig. 4a presents start-up loci of all fish.

$$F_{sphere}(x) = \sum_{i=1}^{n-1} (x_i)^2 , \tag{6}$$

Table 1 Initial conditions for the three fish in the sphere example

Fish	Initial conditions		
	weight	***position***	***fitness***
# 1	5	(9,7)	130
# 2	5	(5,6)	61
# 3	5	(8,4)	80

After initialization, all fish are free to check for new candidate positions that are generated by the individual movement operator. Lets assume that these positions are $\mathbf{x}_1 = (9.6, 6.2)$, $\mathbf{x}_2 = (4.6, 4.4)$ and $\mathbf{x}_3 = (6.2, 4.2)$, and the associated fitnesses are $f(\mathbf{x}_1) = 130.6$, $f(\mathbf{x}_2) = 40.52$ and $f(\mathbf{x}_3) = 56.08$. One should notice that fish #2 and fish #3 found best positions, whereas fish #1 did not move. The positions after the individual movement are then $\mathbf{x}_1 = (9,7)$, $\mathbf{x}_2 = (4.6, 4.4)$ and $\mathbf{x}_3 = (6.2, 4.2)$. Fig. 4b illustrates the individual movement of the three fish in search space for the sphere problem.

According to our model, the next operator to be computed is feeding. As fish #1 remained in the same position, it will not change its weight. The weight of fish #2 and fish #3 will change according to 1. The weight variation depends on the maximum fitness change. The maximum fitness variation in this case was achieved by fish #3 and is equal to 23.92. As a result, fish #3 increased its weight by 1 unit and its new weight became 6. The fitness variation of fish #2 was 20.48. Dividing the fitness variation of fish #2 by maximum fitness change, we conclude that the weight variation of fish #2 is 0.86. The new weight of fish #2 is then 5.86.

Following our model, the third operator to be computed is the collective-instinctive one. This operator evaluates the collective displacement of the fish school considering the individual fitness variations and the individual movement according to 2. As fish #1 stayed in the same position, it will not influence the overall calculation. Considering the values obtained in this iteration, the displacement is (-1.2,-0.6). This vector is applied to all the fish (including fish #1), so the new positions, after third operator computations, are $\mathbf{x}_1 = (7.8, 6.4)$, $\mathbf{x}_2 = (3.4, 3.8)$ and $\mathbf{x}_3 = (5, 3.6)$.

Then, the fitnesses regarding new positions recalculations are 101.8, 26 and 37.96 for fish #1, #2 and #3, respectively. The individual displacement of all fish due to collective-instinctive operator is presented in Fig. 4c. Reader may find it interesting to compare Fig. 4b and Fig. 4c.

The last operator to be considered in this example is the collective-volitive one. For that, one has to obtain the instantaneous value of the barycenter of the fish school according to 3. In this case, the barycenter is (4.96,4.25). Notice that the weight of whole school has increased, therefore a contraction instead of a dilatation is the implicit decision of the school (*i.e.* collective-volitive). By means of using 4, the new positions are $\mathbf{x}_1 = (5.81, 4.89)$, $\mathbf{x}_2 = (4.02, 3.98)$ and $\mathbf{x}_3 = (4.98, 3.92)$. The barycenter and the collective-volitive movement for this step are presented in Fig. 4d.

At this point the algorithm tests if valid stop-conditions are met. Obviously it is not the case yet, thus a new cycle begins as explained above. If one compares the initial and final positions illustrated in Fig. 4, after this first iteration, the reader can observe that all fish are closer to the optimum point (0,0).

Of course the optimum point is unknown to the algorithm. However, in a very peculiar manner the FSS model assures fast convergence towards it (*i.e.* the goal for the search process) because of the above mentioned natural principles instantiated in the FSS algorithm.

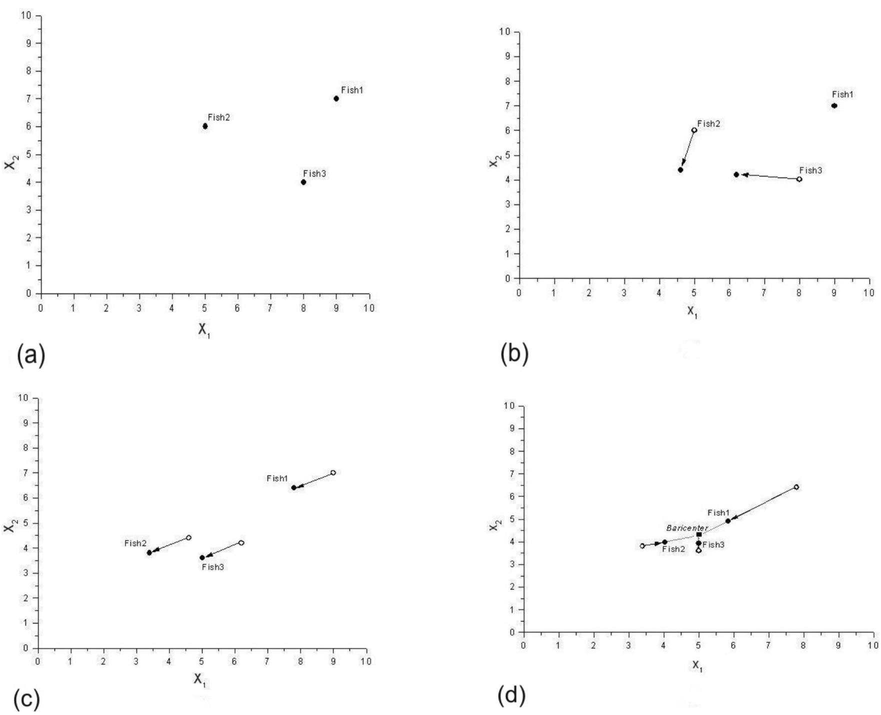

Fig. 4 Example with three fish in the sphere example: (a) Initial position, (b) individual movement, (c) instinctive collective movement and (d) volitive collective movement

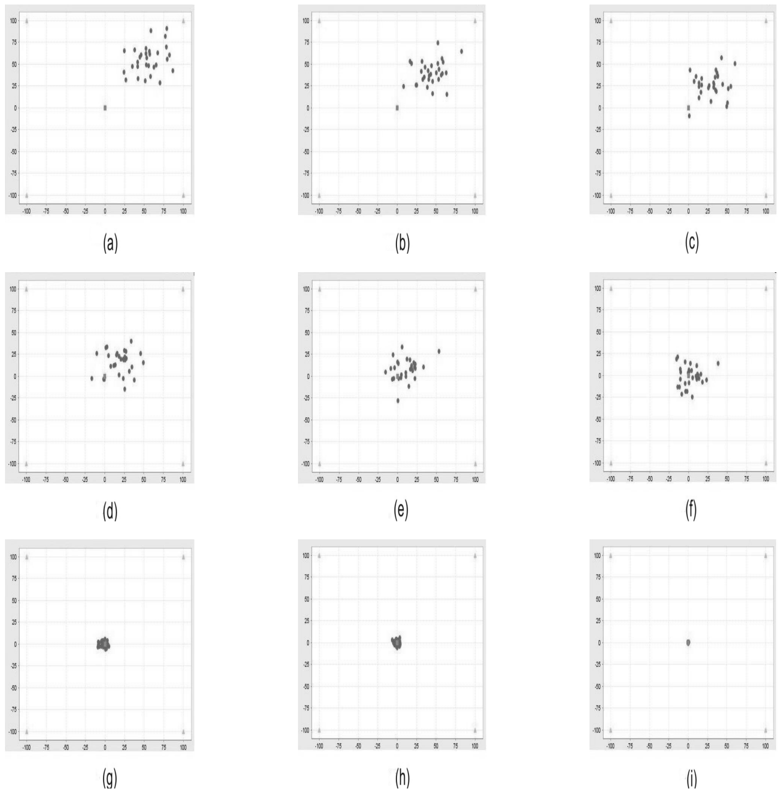

Fig. 5 Fish school evolution after iteration (a) 1, (b) 50, (c) 100, (d) 200, (e) 300, (f) 400, (g) 500, (h) 750 e (i) 1000 for sphere function with 30 fish

In order to illustrate the convergence behavior of the fish school along the iterations, we present the simulation results for the sphere function. In these simulations we used 30 fish, [-100,100] in the two dimensions, initialization range [0,100] in the two dimensions, w_{scale}= 500, initial $step_{ind}$ = 10, final $step_{ind}$ = 0.1, initial $step_{vol}$ = 5, final $step_{vol}$ = 0.5. Fig. 5. shows the fish positions after iteration (a) 1, (b) 50, (c) 100, (d) 200, (e) 300, (f) 400, (g) 500, (h) 750 e (i) 1000, respectively. One can note that the school was attracted to the optimum point (0,0).

5 Comparative Examples

5.1 Experimental Setup

Five benchmark functions were used to carry out simulations and are described in 7, 8, 9, 10, and 11. Table 2 shows the search space, the initialization range, and the optimum for each function. All searches were carried out in 30 dimensions. All five functions are used for minimization problems. Two of these functions namely, Rosenbrock and Schwefel 1.2, represent simple unimodal problems; the other three, Rastrigin, Griewank, and Ackley, are highly complex multimodal functions that contain many local optima. Considered functions in comparisons are:

$$F_{Rosenbrock}(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (1 - x_i)^2\right], \tag{7}$$

$$F_{Rastrigin}(x) = 10n + \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i)\right], \tag{8}$$

$$F_{Griewank}(x) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right), \tag{9}$$

$$F_{Ackley}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e, \tag{10}$$

Table 2 Function parameters

Function	**Parameters**		
	Search space	***Initialization***	***Optima***
Rosenbrock	$-30 \leq xi \leq 30$	$15 \leq xi \leq 30$	1.0D
Rastrigin	$-5.12 \leq xi \leq 5.12$	$2.56 \leq xi \leq 5.12$	0.0D
Griewank	$-600 \leq xi \leq 600$	$300 \leq xi \leq 600$	0.0D
Ackley	$-32 \leq xi \leq 32$	$16 \leq xi \leq 32$	0.0D
Schwefel 1.2	$-100 \leq xi \leq 100$	$50 \leq xi \leq 100$	0.0D

and

$$F_{Schwefel1.2}(x) = \sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2 . \quad (11)$$

A factorial planning of experiments was performed to find suitable parametric combination of individual and volitive steps at both initial and final limits (*i.e.* $step_{ind,initial}$, $step_{ind,final}$, $step_{vol,initial}$, $step_{vol,final}$). We have associated the individual and volitive steps as percentages of the actual search space. Percentage values considered for initial and final limits were, respectively, as follows: 10; 1; 0.1 and 0.1; 0.01; 0.001; 0.0001. W_{scale} was set as 5000; this is half of the number of considered iterations.

All FSS simulations were performed using 30 fish and 10,000 iterations. Only after 30 trials the mean and the standard deviation were recorded. All the fish were randomly initialized in areas of the aquarium that are far from the optimal solution regarding every dimension. This initialization process is carried out in order to measure the ability of the fish school in locating the optimum solution outside the initialization space.

We compared our results with PSO simulation results presented in an earlier landmark paper [9]. Three PSO approaches were considered for comparisons: original PSO with the G_{best} topology, constriction PSO with the G_{best} topology and constriction PSO with the L_{best} topology. All the PSO simulations included 30 trials, each of which performed 300,000 evaluations, and simulations that considered 30 dimensions and used 30 particles. Thus, we considered that a fair convergence analysis between the FSS and PSO approaches could be made.

5.2 Simulation Results

Tables 3, 4, 5, 6 and 7 show the best simulation results for function Rosenbrock, Rastrigin, Griewank, Ackley and Schwefel 1.2, respectively. Only the best six results sorted by the fitness average for each function are presented here. The highlighted values are indications of success for all search performed by FSS.

Table 8 presents the comparison between the FSS and PSO approaches. It contains the best results achieved for the five benchmark functions used to evaluate the performance of the four algorithms.

Table 3 Simulation Results for the Rosenbrock Function – Fitness (average and standard deviation) for 30 trials

$step_{ind,initial}$	$step_{ind,final}$	$step_{vol,initial}$	$step_{vol,final}$	$fitness_{(average)}$	$fitness_{(stddev)}$
0.1	**0.001**	**1**	**0.01**	**16.1183**	**0.729559**
0.1	0.0001	1	0.01	16.4036	0.853030
0.1	0.0001	1	0.001	16.4470	0.770458
0.1	0.001	1	0.001	16.4629	0.797471
10	0.01	1	0.001	44.7585	7.785530
10	0.1	0.1	0.001	46.4926	5.676689

Table 4 Simulation Results for the Rastringin Function – Fitness (average and standard deviation) for 30 trials

$step_{ind,initial}$	$step_{ind,final}$	$step_{vol,initial}$	$step_{vol,final}$	$fitness_{(average)}$	$fitness_{(stddev)}$
10	**0.01**	**10**	**0.1**	**13.3868**	**4.005888**
10	0.1	10	0.01	13.7376	2.889882
10	0.1	10	0.1	14.1285	3.680864
10	0.01	10	0.01	14.5193	2.780258
10	0.01	0.1	0.0001	200.225	22.09435
10	0.01	0.1	0.001	200.652	19.42133

Table 5 Simulation Results for the Griewank Function – Fitness (average and standard deviation) for 30 trials

$step_{ind,initial}$	$step_{ind,final}$	$step_{vol,initial}$	$step_{vol,final}$	$fitness_{(average)}$	$fitness_{(stddev)}$
1	**0.001**	**1**	**0.01**	**0.00270**	**0.002291**
0.1	0.0001	1	0.001	0.00373	0.004015
1	0.001	10	0.01	0.00377	0.002375
0.1	0.001	1	0.001	0.00445	0.003982
0.1	0.0001	1	0.01	0.00499	0.004313
0.1	0.001	1	0.01	0.00603	0.004149

Table 6 Simulation Results for the Ackley Function – Fitness (average and standard deviation) for 30 trials

$step_{ind,initial}$	$step_{ind,final}$	$step_{vol,initial}$	$step_{vol,final}$	$fitness_{(average)}$	$fitness_{(stddev)}$
10	**0.01**	**10**	**0.1**	**0.04004**	**0.020568**
10	0.01	10	0.01	0.08393	0.041568
10	0.01	1	0.01	0.15836	0.032344
10	0.01	1	0.001	0.16337	0.031565
10	0.1	10	0.1	0.18650	0.038461
10	0.1	10	0.01	0.20383	0.039025

Table 7 Simulation Results for the Schwefel 1.2 Function – Fitness (average and standard deviation) for 30 trials

$step_{ind,initial}$	$step_{ind,final}$	$step_{vol,initial}$	$step_{vol,final}$	$fitness_{(average)}$	$fitness_{(stddev)}$
1	**0.001**	**1**	**0.01**	**0.08085**	**0.022414**
1	0.001	1	0.001	0.09159	0.032054
1	0.01	1	0.001	0.09478	0.031902
1	0.01	1	0.01	0.09720	0.026483
1	0.001	0.1	0.001	0.37266	0.264715
1	0.001	10	0.01	0.61065	0.139308

Notice that the FSS algorithm outperforms the original PSO in all the cases. Moreover, FSS achieved excellent results for notoriously hard multimodal functions such as the Rastrigin, Griewank, and Ackley.

Table 8 Overall comparison of results between algorithms – Fitness (average and standard deviation) for 30 trials

Function	Fitness (average and standard deviation)			
	Orig. PSO	*Constricted PSO (G_{best})*	*Constricted PSO (L_{best})*	*FSS*
Rosenbrock	54.6867 (2.8570)	**8.1579 (2.7835)**	12.6648 (1.2304)	16.118 (0.729)
Rastrigin	400.7194 (4.2981)	140.4876 (4.8538)	144.8155 (4.4066)	**13.386 (4.005)**
Griewank	1.0111 (0.0031)	0.0308 (0.0063)	**0.0009 (0.0005)**	0.0027 (0.002)
Ackley	20.2769 (0.0082)	17.6628 (1.0232)	17.5891 (1.0264)	**0.0400 (0.020)**
Schefel 1.2	5.4572 (0.1429)	**0.0 (0.0)**	0.1259 (0.0178)	0.0808 (0.022)

6 Discussion and Conclusions

In this chapter, we have detailed the general ideas and principles embedded in FSS. This novel search algorithm is quite promising as a search tool for dealing with multimodal high dimensional problems, as it may be concluded from the examples provided in previous sections.

The performance of FSS on some multimodal functions was surprisingly good, especially when compared to monomodal ones.

Although previous works [15, 16, 17] have similar titles and motivations, our approach is quite different as it considers bio-inspired operators to directly guide the search process. Additionally, FSS presents an interesting balance between exploration and exploitation abilities, self-adapts quite swiftly out of local minima (towards sought solutions), and self-regulates the search granularity.

We foresee that FSS will most likely receive a great number of extensions in the near future, namely, sea currents, springs, predators, reefs, corals and other barriers to the school progression; all of them, situations to be avoided or taken advantage of. Altogether, these extensions may allow FSS to deal with noise, attractors, repulsors and no-go regions. Finally, breeding is another bio-inspired feature that ought to be considered further in the near future.

References

1. Michalewicz, Z., Fogel, D.: How to solve it - modern heuristics, 2nd edn. Springer, Heidelberg (2004)
2. Holland, J.: Adaptation in natural and artificial systems. University of State of Michigan Press, Ann Arbor (1975)
3. Eiben, A., Smith, J.: Introduction to evolutionary computing. Springer, Heidelberg (2003)

4. de Castro, L., Timmis, J.: Artificial immune systems - a new computational intelligence approach. Springer, Heidelberg (2002)
5. Dorigo, M., Gambardella, L.: Ant colony systems: a cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation 1(1), 53–66 (1997)
6. Dorigo, M., Blum, C.: Ant colony optimization theory: A survey. Theoretical Computer Science 344, 243–278 (2005)
7. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948 (1995)
8. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: Proceedings of the 6th International Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39–43 (1995)
9. Bratton, D., Kennedy, J.: Defining a standard for particle swarm optimization. In: Proceedings of the IEEE Swarm Intelligence Symposium (SIS 2007), Hawaii, USA, pp. 120–127 (2007)
10. Clerc, M., Kennedy, J.: The particle swarm - explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 6(1), 58–73 (2002)
11. Carvalho, D.F., Bastos-Filho, C.J.A.: Clan particle swarm optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2008), Hong Kong, pp. 3044–3051 (2008)
12. Mendes, R., Kennedy, J., Neves, J.: Watch thy neighbor or how the swarm can learn from its environment. In: Proceedings of the IEEE Swarm Intelligence Symposium (SIS 2003), Indiana, USA, pp. 88–94 (2003)
13. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2002), Hawaii, USA, pp. 1671–1676 (2002)
14. Suganthan, P.N.: Particle swarm optimiser with neighbourhood operator. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 1999), Washington, DC, USA, pp. 1958–1962 (1999)
15. Oboshi, T., Kato, S., Mutoh, A., Itoh, H.: Effectiveness of strays on ability of fish school to escape from predator. In: Proceedings of the 7th Asia-Pacific Conference on Complex Systems, Cairns, Australia (2004)
16. Luo, F.F., Chen, G.L., Guo, W.Z.: An improved 'fish-search' algorithm for information retrieval. In: Proceedings of IEEE International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE 2005), Wuhan, China, pp. 523–528 (2005)
17. Hersovici, M., Jacovi, M., Maarek, Y., Pelleg, D., Shtalhaim, M., Ur, S.: The shark-search algorithm. An application: tailored Web site mapping. Computer Networks and ISDN Systems 30(1-7), 317–326 (1998)