

Application of Fractional Order ABC and GA for Neural Network Training and Clustering Process

G. Lavanya

*Research Scholar, Anna University Regional Centre, Madurai
Tamilnadu, India
lavanya0283@gmail.com*

S. Srinivasan

*Professor & Head, Department of Computer Science and Engineering,
Anna University Regional Centre, Madurai
Tamilnadu, India
sriniss@yahoo.com*

Received 29 July 2014

Accepted 4 May 2015

Abstract

Literature presents several search algorithms to find an item with specified properties from a search space defined by a mathematical formula or procedure. One of the widely accepted algorithms is optimization algorithm which can find the optimal element within a certain period of time if the search space is defined. Recent works formulate several problems as optimization problems which were then solved by many optimization algorithms. Accordingly, in a previous paper, a hybrid optimization algorithm, called FAGA was proposed using fractional order Artificial Bee Colony (ABC) and Genetic Algorithm (GA) for optimization to solve the existing benchmark problems. In this paper, we have planned to apply the FAGA algorithm to well defined-real time problems of neural network training and the clustering process. Through neural network training, data classification will be done by making use of FAGA algorithm as neural network training procedure. Similarly, medical image segmentation will be done using clustering process through FAGA algorithm. The performance of the FAGA algorithm in those two processes will be evaluated with different evaluation metrics and the comparison of the FAGA algorithm will be also carried out with the existing ABC and genetic algorithm.

Keywords: FAGA, Neural Network, Image Segmentation, Data Classification

1. Introduction

As the extent of advanced in data collection becomes exponentially huge volumes of crude information need to be extricated. The most well-known strategy is to utilize Data classification. The procedure of data classification is called as the methodology of knowledge disclosure which has been utilized as a part of most inter-disciplinary, for example, database, artificial intelligence statistics, visualization, parallel figuring and different fields. One new and to a great degree capable calculation utilized as a part of data classification is Evolutionary algorithm. This incorporates science motivated calculations, for example, Genetic

Algorithms and swarm based methodologies like Ant Colonies, and Particle Swarm Optimizations. Image segmentation is an essential venture in image analysis used to discrete the input picture into significant regions. Optimization is a method used to look for qualities for a set of parameters that amplify or minimize target capacities subject to specific limitations. Artificial Bee Colony calculation is a novel optimization calculation motivated by the regular conduct of honey bees in their quest methodology for the best sustenance sources.

Swarm Intelligence (SI), which is a part of evolutionary computing, is quite popular among the researchers. The swarm intelligence's algorithm is very

efficient for solving problems of optimization.¹ Swarm Intelligence is a theory that deals with the significance of team-work regarding social insect colonies, different animal societies, or the association of species to their environment, to propose methods or to distributed problem-solving devices. Acquiring the features as well as manners of organisms, many methods of improving the optimization deals with swarm intelligence, are introduced one after another. Also, many applications of optimization algorithms, related to swarm intelligence are introduced continuously,^{2, 3, and 4} along with several methods for solving optimization issues have been developed.⁵ Furthermore, several orthodox and experimental approaches have become quite significant. Within some of the experimental approach, a significant theory, which is a method related to social behavior of few creatures.^{6, 7} Swarms inherently utilize forms of decentralized control and self-organization to accomplish their goals.⁸ SI is the combined behavior of decentralized, self-organized natural or artificial systems.

SI systems are consisted of a group of simple agents communicating within them and with the environment.⁹ The agents do not get any instruction from any control system, they follow simple protocol, by interacting within themselves and results in the emergence of intelligence global behavior which is unknown to the individual agents.^{10, 11} SI examined the combined behavior of systems composed of several individuals communicating locally with each other and with their environment. Recent studies have aimed on the meta-experimental theory, such as, Ant Colony Optimization (ACO),¹² Particle Swarm Optimization (PSO) and ABC.¹³ It has been proved, that these algorithms have the ability to offer enhanced solutions than conventional algorithms. By borrowing ideas with respect to nature swarming behavior in computer methodologies, methods obtained for hard optimization problems become robust, scalable and easily distributed.^{14, 15} Some of the essential swarm methods are: PSO, ACO, ABC, Consultant-Guided Search (CGS) and many more. Again, the complete swarm indicates intelligent behavior, serving ultimate solutions for complex problems like, predator evasion and shortest path finding.^{16, 17}

GA is an orthodox optimization algorithm related to natural selection and the mechanism of population genetics.^{18, 19} These techniques for solving optimization

problem performs better than the orthodox techniques in an efficient way. GA evolve as a superior from orthodox methods and optimization methods in four following ways: a) GA search is quite distinguished from a population of points. So, it will not get blocked in local optimal solution like in orthodox techniques, which make search with respect to a single point. b) GA applies probabilistic method of selection rather than deterministic ones. c) GA deals with Chromosome, hence is encoded type of potential solutions' parameters, not the parameters themselves. d) GA depends on fitness score that is acquired from objective functions, without other derivative or auxiliary information.²⁰

In recent times, ABC algorithm has been suggested regarding optimization problems. According to the ABC algorithm, bee's job is to find appropriate flowers and to locate suitable place of flowers. This total procedure is divided in an asynchronous process, which means the bees communicate with each other, after one find the food source, by dancing on the located source of food.²¹ Some significant benefit of this algorithm is: a) the formation of the algorithm is ideal for distinguished approach, hence less time consuming, b) Dependable, since it enables adjustments and considers specific knowledge of the problem by observing nature.²² A hybrid optimization algorithm FAGA is presented in Ref. 23 using fractional order ABC and GA. The FAGA solves the existing problems faced by the exiting optimization problem such as convergence problem, local minimization problem, low optimal speed and time.

The FAGA algorithm has four phases such as, employee bee, onlooker bee, mutation and scout bee. In employee bee phase, neighbor solution is generated based on ABC algorithm. Then, in onlooker bee, the probability is used to select a solution and new solution is generated based on fractional calculus-dependent neighbor solution. The mutation operation of GA is used in the mutation module and then the scout bee phase is carried out. A genetic algorithm is generally said to converge when there is no critical change in the estimations of fitness of the population from one generation to the next. Both ABC and GA have local optimization problem. The main features of FAGA are that it directly influences the speed of convergence of the algorithm and it also permits to understand the local and search capability. FAGA has potential for

interpreting the evolution of the algorithm and to control its convergence and for the entire optimization problem which existed in the conventional method; FAGA has the ability to provide results.

There are several works available in the literature but some of those significant research works are reviewed here. One can deduce from Ref. 24 that the efficiency of differential evolution (DE), PSO, and Evolutionary Algorithms (EAs), which can be applied as numerical optimization methods. A well-organized approach for adapting control parameter settings related with DE has been introduced by Janez Brest *et al.*²⁵ The efficiency of the ABC with GA, PSO and PS-EA, which were swarm intelligence and population related methods as the ABC algorithm were verified by Dervis Karaboga and Bahriye Basturk.²⁶ A Discrete Particle Swarm Optimization (DPSO) technique to solve the NP-hard single machine total weighted tardiness scheduling problem in presence of sequence-dependent setup times was suggested by Davide Anghinolfi *et al.*²⁷ Wu Deng *et al.* have introduced a dual-stage hybrid swarm intelligence optimization algorithm known as GA-PSO-ACO algorithm, which unites the theory of GA, PSO and ACO related to evaluating the traveling salesman problem.² A hybrid optimization algorithm related to Cat Swarm Optimization (CSO) and ABC was suggested by Pei-Wei Tsai *et al.*¹ Min-Yuan Cheng and Li-Chuan Lien have suggested an optimization hybrid swarm algorithm the particle bee algorithm (PBA), that take knowledge from the intelligent swarming behavior of honeybees and birds, to integrate BA global search ability along with the local search advantages of PSO.²⁸

In this paper, we apply the FAGA algorithm for image segmentation and data classification. The image segmentation is done by converting the RGB image into grey and generating number of solutions randomly and by processing each solution using FAGA algorithm. The FAGA give a best solution based on best fitness and the Centroid in the best solution is used to cluster the pixels. The clustered pixels would be formed as segmented image. The data classification is done by training the neural network. The neural network uses weights between the neurons in each layer. The weights in the neural networks are assigned, while training is based on the FAGA algorithm. The neural network trained using FAGA algorithm would classify the data in a dataset. The structure of the paper is organized as follows: the second section explained our application of FAGA on

image segmentation and data classification and the third section delineates the results obtained for our technique compared with the existing techniques and the last section concludes our technique.

2. Proposed Application of Fractional Order ABC and GA for neural network training and Clustering Process

This section shows the proposed application of Fractional Order ABC and GA (FAGA) for neural network training and clustering process. The clustering process based on FAGA is done for image segmentation; and the training of neural network based on FAGA is done for data classification. The FAGA [is a hybrid algorithm of fractional order Artificial Bee Colony (ABC) and Genetic Algorithm (GA)].²³ The FAGA has four phases namely, employee bee phase, onlooker bee phase, mutation phase and scout bee phase. In Ref. 23 we proposed FAGA and obtained better minimization and convergence rate using different functions as fitness calculation. Here, the FAGA is applied for data classification and image segmentation. The process of FAGA is as follows:

Employee Bee Phase

The colony of artificial bees contains three groups of bees: employed bees, onlookers and scouts. A bee waiting on the dance area for making decision to choose a food source is called an onlooker and a bee going to the food source visited by it previously is employed bee. A bee carrying out random search is called a scout. For every food source, there is only one employed bee. In other words, the number of employed bees is equal to the number of food sources around the hive. The employed bee whose food source is exhausted by the employed and onlooker bees becomes a scout. In each cycle, the search consists of four steps: sending the employed bees onto the food sources and then measuring their nectar amounts; selecting of the food sources by the onlookers after sharing the information of employed bees and determining the nectar amount of the foods; mutation operation; determining the scout bees and then sending them onto possible food sources. Here, the position of a food source represents a possible solution of the optimization problem and the nectar amount of a food source corresponds to the fitness of the associated solution. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. In employee bee phase, neighbor solution is generated based on ABC algorithm.

At the initialization stage, a set of food source positions are randomly selected by the employed bees and their nectar amounts are determined. Then, these

bees come into the hive and share the nectar information of the sources with the onlooker bees waiting on the dance area within the hive. Initially, ABC generates a randomly distributed initial population represented by pop_{ini} having N_p solutions where each solution is the food source position and N_p is the population size. Each solution is represented by g_j , where $1 \leq j \leq N_p$ is a M-dimensional vector, where M is the number of optimization parameters taken into consideration. After initialization, the population of the positions is subjected to repeated cycles of the search processes of the employed bees, the onlooker bees and scout bees.

Onlooker Bee Phase

In this phase, selection of the food sources by the onlookers after receiving the information of employed bees and generation of new solution based on fractional calculus is carried out. The onlooker bee prefers a food source area depending on the nectar information distributed by the employed bees on the dance area. As the nectar amount of a food source increases, the probability with which that food source is chosen by an onlooker increases, too. Hence, the dance of employed bees carrying higher nectar recruits the onlookers for the food source areas with higher nectar amount. An onlooker bee chooses a food source depending on the probability value associated with that food source P_j given by the expression:

$$P_j = \frac{F_j}{\sum_{k=1}^{N_p} F_k} \tag{1}$$

In the above equation F_j is the fitness value of the j^{th} solution evaluated by its employed bee, which is proportional to the nectar amount of the food source in the position j and N_p is the number of food sources which is equal to the number of employed bees. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source with a probability related to its nectar amount and then, she produces a modification on the position (solution) in her memory and checks the nectar amount of the candidate source (solution). Providing that its nectar is higher than that of the previous one, the bee memorizes the new position and forgets the old one. In other words, a greedy selection mechanism is employed as the selection operation between the old and the current food sources. The position update is made with the use of fractional calculus. Let the old position be represented

by $z_{j,k}$ and the new position is represented by $y_{j,k}$, which is defined by the equation below:

$$z_{j,k} = y_{j,k} + \varphi_{j,k} (y_{j,k} - y_{i,k}), i \neq j \tag{2}$$

Where, $i = \{1,2,\dots,N_p\}$ and $k = \{1,2,\dots,M\}$. $\varphi_{j,k}$ is a random number in the range $(-1, 1)$. Which controls the production of a neighbor food source position around $y_{j,k}$ and the modification represents the comparison of the neighbor food positions visually by the bee. The position update equation shows that as the difference between the parameters of the $y_{j,k}$ and $y_{i,k}$ decreases, the perturbation on the position $y_{j,k}$ also decreases. Thus, as the search approaches to the optimum solution in the search space, the step length is adaptively reduced.

Fractional Calculus (FC) is a natural extension of the classical mathematics and extends the possibility of taking real number powers or even complex number powers of the differentiation operator or the integration operator. In our case, fractional calculus is used for position updation step. Rearranging the position updation step, we have equation as:

$$z_{j,k} - y_{j,k} = \varphi_{j,k} (y_{j,k} - y_{i,k}) \tag{3}$$

In the above equation $z_{j,k}$ is the position update from $y_{j,k}$ in the previous step, representing in the time domain, we can write $y_{j,k}$ as z_t when $z_{j,k}$ is taken as

z_{t+1} . Hence the above equation is formed as:

$$z_{t+1} - z_t = \varphi_{j,k} (y_{j,k} - y_{i,k}) \tag{4}$$

The left side $z_{t+1} - z_t$ is the discrete version of the derivative of order $\alpha = 1$. Hence the equation is formed as:

$$D^\alpha [z_{t+1}] = \varphi_{j,k} (y_{j,k} - y_{i,k}) \tag{5}$$

Here, by discrete time approximation (taking first four terms), the equation is formed as:

$$z_{t+1} - \alpha z_t - \frac{1}{2} \alpha z_{t-1} - \frac{1}{6} \alpha (1-\alpha) z_{t-2} - \frac{1}{24} \alpha (1-\alpha)(2-\alpha) z_{t-3} = \varphi_{j,k} (y_{j,k} - y_{i,k}) \tag{6}$$

Rearranging, we have the updated position equation as:

$$z_{t+1} = \alpha z_t + \frac{1}{2} \alpha z_{t-1} + \frac{1}{6} \alpha (1-\alpha) z_{t-2} + \frac{1}{24} \alpha (1-\alpha)(2-\alpha) z_{t-3} + \varphi_{j,k} (y_{j,k} - y_{i,k}) \tag{7}$$

Mutation Module

In this module, mutation operation is carried out to have better solution. The mutation operator is one of the operators used in GA. GAs are adaptive heuristic search algorithm premised on the evolutionary ideas of natural selection and genetic which has been widely studied, experimented and applied in many fields in engineering worlds. GA belong to the larger class of EA, which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated; multiple individuals are stochastically selected from the current population based on their fitness and modified using recombination and mutation to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

In mutation, individuals are perturbed probabilistically to bring a change in the individuals. Using mutation operator, there is a probability that some new features might appear due to change in the chromosome. Mutation is a genetic operator that alters one or more gene values in a chromosome from its initial state. This can result in entirely new gene values being added to the gene pool. With these new gene values, the genetic algorithm may be able to arrive at better solution than was previously possible. Mutation is an important part of the genetic search as it helps to prevent the population from stagnating at any local optima. Mutation occurs during evolution according to a user-definable mutation probability. This probability should usually be set fairly low and if it is set to high, the search will turn into a primitive random search.

Scout Bee Phase

The employed bee whose food source is exhausted by the employed and onlooker bees becomes a scout and it carries out random search. The food source whose nectar is abandoned by the bees is replaced with a new food source by the scouts. This is simulated by randomly producing a position and replacing it with the abandoned one. Here, if a position cannot be improved further through a predetermined number of cycles called limit then that food source is assumed to be abandoned.

The control parameters used in the algorithm consist of the number of the food sources which is equal to the

number of employed or onlooker bees, the value of limit, mutation operation and the maximum cycle number. Here, the onlookers and employed bees carry out the exploitation process in the search space, the scouts control the exploration process and mutation betters the solution.

2.1. Application of FAGA for Image Segmentation

This section delineates the application of FAGA algorithm to segment an image. Fig.1 shows the process of segmenting an image using FAGA.

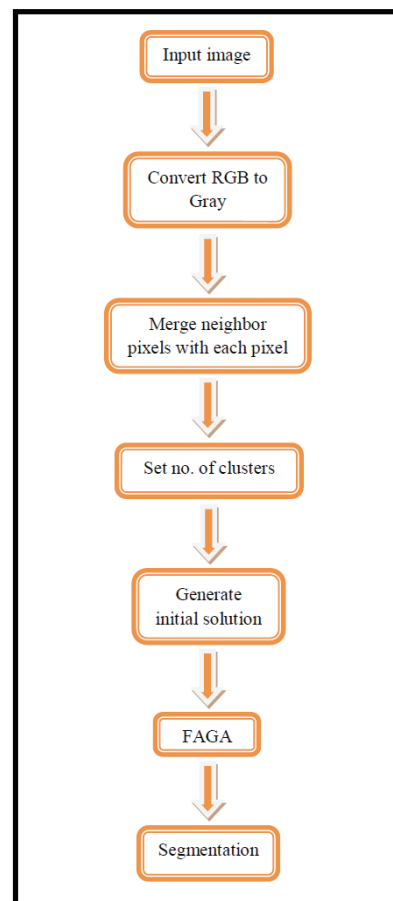


Fig.1 Process of Segmenting an Image

According to Fig.1, the image segmentation process is explained as follows: an image is given as input to the system. In general, the image would be in RGB color format i.e. each pixel of the image would have three dimensional values. Therefore to convert it to one

dimension value, the RGB image is converted to grey image. The adjoining pixels are merged (by assumption) with each pixel of the image for clustering process. Initially, the number of clusters required is set by the user and based on it the initial solution is generated to process the FAGA. The image given as input is then segmented using FAGA.

2.1.1. Merge Neighbor Pixels

This process is done after converting the RGB image into grey image. The merging is done by

PX ₁₁	PX ₁₂	PX ₁₃	PX ₁₄	PX ₁₅	.	.	PX _{1m}
PX ₂₁	PX ₂₂	PX ₂₃	PX ₂₄	PX ₂₅	.	.	PX _{2m}
PX ₃₁	PX ₃₂	PX ₃₃	PX ₃₄	PX ₃₅	.	.	PX _{3m}
PX ₄₁	PX ₄₂	PX ₄₃	PX ₄₄	PX ₄₅	.	.	PX _{4m}
PX ₅₁	PX ₅₂	PX ₅₃	PX ₅₄	PX ₅₅	.	.	.
.
.
PX _{n1}	PX _{n2}	PX _{n3}	PX _{n4}	PX _{n5}	.	.	PX _{nm}

Fig.2. merging of adjoining pixels

According to Fig.2, the merging process explains as follows: consider we take the centroid as the pixel PX₃₃ of the image. While taking the PX₃₃ to process, we also take the neighboring pixels (i.e. PX₂₂, PX₂₃, PX₂₄, PX₃₂, PX₃₄, PX₄₂, PX₄₃ and PX₄₄) with it. Therefore the merging of PX_{nm} is shown by an equation below:

$$Mer(PX_{nm}) = \left\{ \begin{array}{l} PX_{(n-1)(m-1)}, PX_{(n-1)m}, PX_{(n-1)(m+1)}, \\ PX_{n(m-1)}, PX_{nm}, PX_{n(m+1)}, PX_{(n+1)(m-1)} \\ , PX_{(n+1)m}, PX_{(n+1)(m+1)} \end{array} \right\} \tag{8}$$

2.1.2. Initial Solution

The initial solution is generated by setting upper bound and lower bound based on the pixels in the image taken for segmentation. The initial solution would have N_p number of solutions and each solution would have the centroids which are chosen randomly from the pixels of the image taken for segmentation. The number of centroids on each solution is set by the user. Fig.3 shows a sample initial solution generated from the pixels of image taken for segmentation.

assumption but actually we are not merging. This is done because while considering a pixel we have to take the neighboring pixels too. Fig.2 shows the merging of adjoining pixels of an image.

	C ₁	C ₂	C ₃
N ₁	Mer(PX ₂₁)	Mer(PX ₄₅)	Mer(PX ₈₅)
N ₂	Mer(PX ₃₅)	Mer(PX ₆₂)	Mer(PX _{9m})
N ₃	Mer(PX ₅₃)	Mer(PX _{n5})	Mer(PX ₈₇)
.	.	.	.
.	.	.	.
N _p	Mer(PX ₄₉)	Mer(PX ₇₅)	Mer(PX ₉₄)

Fig.3. Sample initial solution

In Fig.3 {N₁, N₂, ..., N_p} represents the number of solutions and we set three centroids which are {C₁, C₂ and C₃}. If we consider the solution N₁, the {Mer(PX₂₁), Mer(PX₄₅) and Mer(PX₈₅)} are the centroids generated for solution N₁. Each solution is then given to the FAGA to process it. While processing FAGA, the fitness calculation is an essential step to identify the nectar amount.

2.1.3. Nectar Calculation of FAGA for Image Segmentation

The fitness (i.e. nectar amount) calculation of FAGA is an essential step to decide which solution is best to segment the image. In Ref. 23 different functions are used separately to calculate the fitness of FAGA and

compared the minimization and convergence rate with other algorithms. In this paper, the fitness is calculated as follows:

$$F_j(IS) = \sum_{k=1}^3 \text{dist}(C_k, PX_{nm}) \tag{9}$$

In the above equation F_j is the fitness of j^{th} solution where $j = N_1 \text{ to } N_p$; and C_k is the k^{th} centroid of j^{th} solution and k varies from 1 to 3 since we take three centroids; and PX is the pixels in the image; and IS represents image segmentation. The above equation is explained as follows: if we consider the first solution, the distance between each centroid of the first solution and each pixel of the image is taken separately and summed to get the fitness value of it.

2.1.4. Segmentation

The FAGA algorithm has different phases and the solutions generated are subjected to all the phases of the FAGA algorithm. In each phase, the solutions would get

changed and the fitness is calculated for all the solution in each phase. The best solution identified at each phase is stored for further process. The best solution is based on best fitness (nectar amount). The solution that obtained least fitness value is judged as best solution since a good cluster requires less distance between the centroid and the elements in it. The FAGA is then repeated using the solutions obtained at the final phase until the number of iterations set. At the end of the iteration, the best solutions stored are compared with each other and the centroids of the best solution among them are chosen to cluster the pixels of the image. The clustered pixels are then formed as segmented image.

2.2. Application of FAGA for Data Classification

This section delineates the application of FAGA for data classification. Here a neural network which is trained based on FAGA algorithm is used to classify the dataset. In neural network the weight values between the neurons of each layer is fixed by the FAGA algorithm. Fig.4 shows a sample neural network.

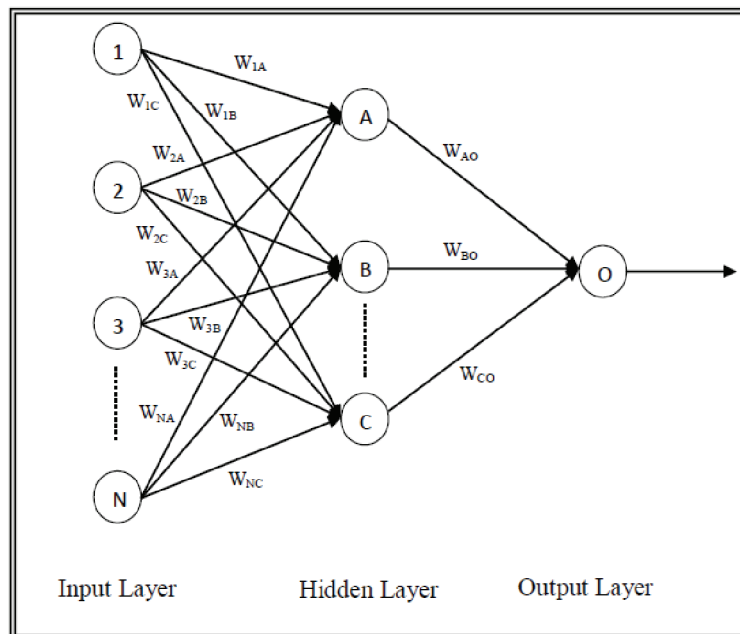


Fig.4. Sample neural network

The neural networks would have three layers which are input layer, hidden layer and output layer. The inputs are given to the neurons in the input layer and the output is obtained from the neuron in the output layer.

In between each neuron on each layer, there would have weight values which are used to train the neural network. Initially, the weight values are random numbers and later the weight values are changed during

training using certain algorithm. Here we used the FAGA to fix the weight values between each layer. To apply FAGA initially numbers of solutions are

generated that have different weight values. Fig.5 shows the sample initial solution generated for assigning the weight values of neural network using FAGA.

	W _{1A}	W _{1B}	W _{1C}	W _{2A}	W _{2B}	W _{2C}	W _{3A}	W _{3B}	W _{3C}	W _{NA}	W _{NB}	W _{NC}	W _{AO}	W _{BO}	W _{CO}
N ₁	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V _{1m}
N ₂	V ₂₁	V ₂₂	V ₂₃	V ₂₄	V _{2m}
N ₃	V ₃₁	V ₃₂	V ₃₃	V ₃₄	V _{3m}
.
.
N _p	V _{p1}	V _{p2}	V _{p3}	V _{p4}	V _{pm}

Fig.5. Sample initial solution generated to train neural network

Fig.5 explains as follows: the {N₁, N₂, ..., N_p} represents the number of solutions generated to train the neural network; and the {W_{1A}, W_{1B}, ..., W_{CO}} represents the weights in the neural network; and V represents the corresponding weight values for N and W. These solutions are processed using FAGA and the best solution is chosen based on FAGA.

The calculation of fitness is done on each phase in FAGA process and the best solution is saved. The FAGA is repeated until the iteration numbers set using the solutions obtained in last phase as input to next iteration. The weight values in the best solution obtained at last iteration is assigned to the neural network for data classification.

2.2.1. Nectar Calculation of FAGA for Neural Network Training

In FAGA, the nectar amount (fitness) calculation is an essential step to identify best solution. The neural network is trained based on FAGA to classify the data in a dataset. The fitness calculation for neural network training is based on mean square error (MSE). It is calculated as follows:

$$F_j(NN) = (Tar - O)^2 \tag{10}$$

In the above equation F_j is the fitness of j^{th} solution where j varies from N₁ to N_p; and NN represents neural network training; and Tar denotes the target we need to get; and O represents actual output we get using j^{th} solution. The output O is obtained by applying the weight values of j^{th} solution to the neural network. The fitness is calculated for each solution and the solution with best fitness is stored and the entire solution is given to next phase of the FAGA.

2.2.2. Classification of Data

After training the neural network based on FAGA, the dataset is given as input to the neurons in the input layer of neural network. The weight values assigned between the neurons in the input layer and the neurons in the hidden layer are processed with the input and the solutions are then processed with the weight values between the neurons in the hidden layer and the neuron in the output layer. The weight values between the neurons in the input layer and neurons in the hidden layer and the weight values between the neurons in the hidden layer and neuron in the output layer are assigned based on the best solution obtained using FAGA algorithm. The output layer of the neural network would then give the classified data. Fig.6 shows the algorithm for the application of FAGA on image segmentation and Fig.7 shows the algorithm for the application of FAGA on data classification.

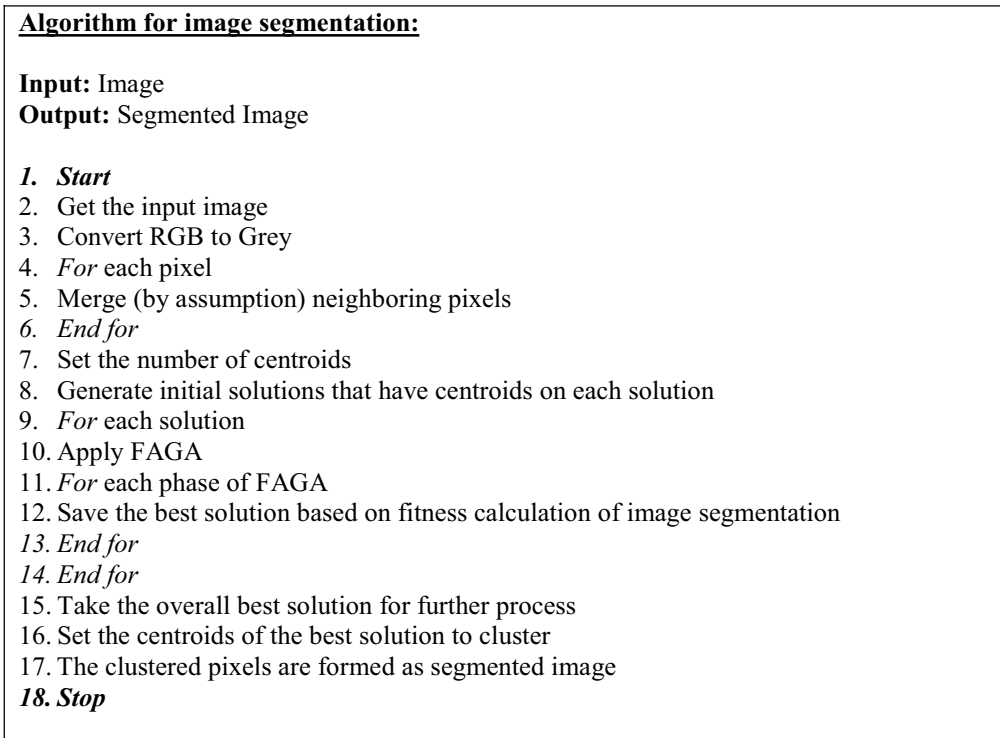


Fig.6.Algorithm for the application of FAGA on Image Segmentation

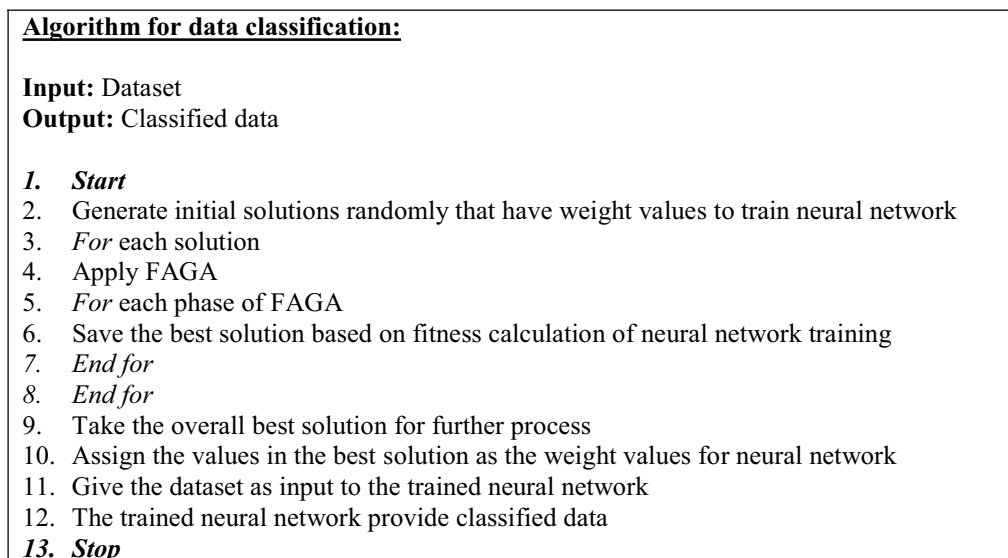


Fig.7. Algorithm for the application of FAGA on Data Classification

3. Result and Discussion

This section delineates the results obtained for our technique FAGA compared to the existing ABC, Genetic Algorithm (GA) and ABC-GA techniques. The

performance of our technique is compared by applying it for both, image segmentation and data classification. The experimentation is done on the system that has the configuration as follows: Intel i5 processor with 4GB

RAM and Windows 7 OS. The implementation is done on Matlab R2013a.

3.1. Dataset Description

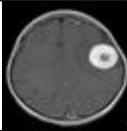



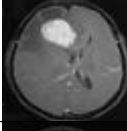

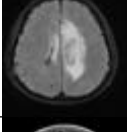

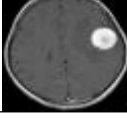
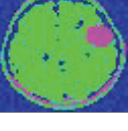
This section explains the datasets taken for our experimentation. We took five tumor affected MRI brain images to test the image segmentation process of our technique compared to the existing techniques. The tumor affected MRI brain images are taken from publically available source. To experiment the data classification process, we took three datasets which are Pima Indians Diabetes (PID), Heart Disease and Lung Cancer from UCI Machine Learning Repository (<https://archive.ics.uci.edu/ml/datasets.html>). The PID dataset has eight attributes and seven hundred and sixty eight instances. The Heart Disease dataset has seventy five attributes and three hundred and three instances.

The Lung Cancer dataset has fifty six attributes and thirty two instances.

3.2. Performance based on Image Segmentation

This section shows the performance of our technique FAGA compared to the existing techniques in terms of accuracy based on image segmentation. The images used for our experimentation are brain tumor images. Here we compared the accuracies of non-tumor region (brain region), tumor region and background region of the images obtained for our technique with the existing techniques. We used five brain tumor images for our experimentation. Table.1 shows the five normal brain tumor images and its corresponding segmented images using our technique.

Table 1. Normal and Segmented Brain Tumor Images

Image Numbers	Normal Brain Tumor Images	Segmented Brain Tumor Images
1		
2		
3		
4		
5		

In Table 1, the black colored region in normal brain tumor images is the background region; and the grey colored region in normal brain tumor images is the brain region; and the white colored region in the brain is the tumor. In the segmented brain tumor images of Table.1, the blue colored region is the segmented background

region; and the green colored region is the segmented brain region; and the pink colored region in the brain is the segmented tumor region. Table.2 shows the performance comparison using first image in terms of accuracy.

Table 2. Performance comparison using first image in terms of accuracy

	Non-Tumor	Tumor	Background	Average
ABC	0.975574	0.555668	1	0.843747
GA	0.987517	0.602227	1	0.863248
FAGA	0.992184	0.791498	0.942568	0.90875

The accuracies obtained for our technique compared to the other existing techniques is shown in Table.2 and the accuracy is evaluated for three regions which are brain region (non tumor region), tumor region and background region. Here our technique FAGA is compared with the existing techniques which are ABC, GA and ABC-GA. While comparing the performance

using first image, our technique performed better in terms of non-tumor segmentation and tumor segmentation compared to the other existing techniques. Overall if we take the average of the accuracy of the three regions (non-tumor, tumor and background), our technique performed better than the other techniques.

Table 3. Performance comparison using second image in terms of accuracy

	NonTumor	Tumor	Background	Average
ABC	0.970947	0.561659	1	0.844202
GA	1	0.756303	1	0.918768
FAGA	1	0.798319	0.962382	0.920234

Table 3 shows the accuracies obtained using second image. Here in terms of non-tumor region segmentation, our technique and GA technique performed better than the other techniques; and in terms

of tumor region segmentation, our technique performed better compared to other techniques. While comparing the average accuracy of the three regions, our technique performed better than the other algorithms.

Table 4. Performance comparison using third image in terms of accuracy

	NonTumor	Tumor	Background	Average
ABC	0.990476	0.785276	1	0.925251
GA	1	0.619632	1	0.873211
FAGA	1	0.870833	1	0.956944

Table 4 shows the accuracies obtained using third image. Here in terms of non-tumor region segmentation, our technique and the GA technique performed better than other algorithms; and in terms of tumor region

segmentation, our technique performed better compared to other techniques. When the average of all the three regions is compared, our technique performed better than other techniques.

Table 5. Performance comparison using fourth image in terms of accuracy

	NonTumor	Tumor	Background	Average
ABC	0.989494	0.562044	1	0.850513
GA	0.999088	0.446267	1	0.815118
FAGA	0.998444	0.863035	1	0.953826

Table 5 shows the performance comparison using fourth image. Here in terms of non-tumor region segmentation and tumor region segmentation, the accuracy of our proposed technique is better compared to other techniques.

Table 6. Performance comparison using fifth image in terms of accuracy

	NonTumor	Tumor	Background	Average
ABC	0.978529	0.584466	1	0.854332
GA	0.988997	0.616505	1	0.868501
FAGA	0.994096	0.790615	0.943038	0.909249

Table 6 shows the performance comparison using fifth image. Here in terms of non-tumor region segmentation and tumor region segmentation, the accuracy of our proposed technique is better compared to other techniques. When comparing the overall performance, the proposed technique is better than other techniques.

3.3. Performance based on Data Classification

This section shows the performance of our technique based on data classification compared to the existing techniques. To compare the performance based on data classification, we used three datasets which are PID, Heart Disease and Lung Cancer for experimentation. Fig.8 shows the accuracy obtained using PID dataset.

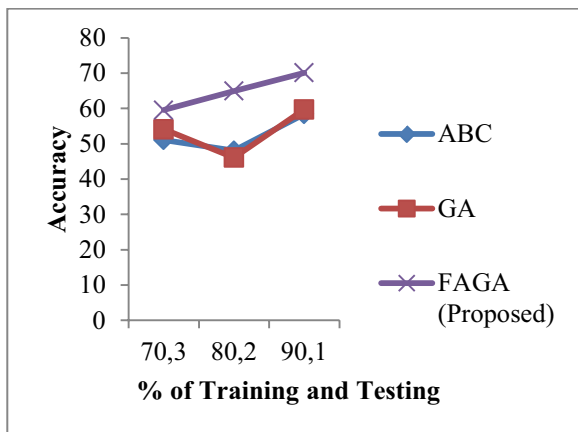


Fig.8. Accuracy using PID dataset

In Fig.8 the accuracy obtained for the proposed technique is compared with the existing techniques by varying the datasets for training and testing. The 70.3 in

the x-axis denotes seventy percentages of the data is used for training and thirty percentages is used for testing; and the 80.2 represents eighty percentages of data is used for training and twenty percentages for testing; and the 90.1 represents ninety percentages of data is used for training and ten percentages is used for testing. From Fig.8, it is clear that when increasing the percentage of testing data, the accuracy of FAGA is increasing from 60% to 70%. Thus it is clear that the accuracy of FAGA is more compared to that of other method. Here in Fig.8 for all the three varied percentages of PID dataset, our technique FAGA performed better compared to other algorithms.

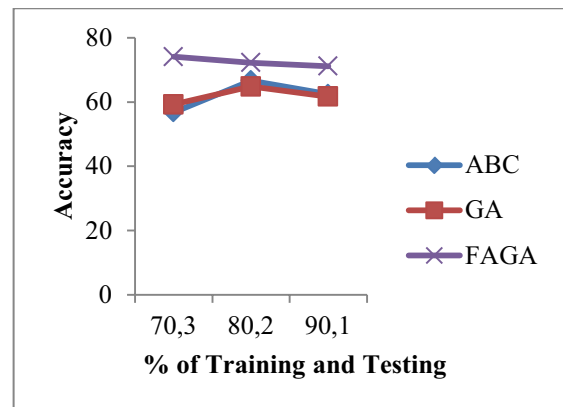


Fig.9. Accuracy using Heart Disease dataset

Fig.9 shows the accuracy obtained for our proposed technique compared to the existing techniques using Heart Disease dataset. Here, the proposed technique classified the dataset better than the other techniques for all the varied percentages of dataset. From Fig.9, it is clear that there is a slight change in the accuracy of FAGA from 70 to 75%, as the training data increases.

The percentage of other method is between 60 to 70%, thus the proposed technique has more accuracy compared to that of all the existing method. Fig.10 shows the accuracy comparison using Lung Cancer dataset.

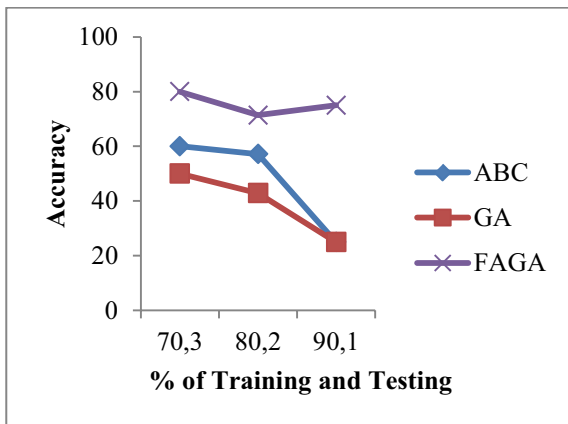


Fig.10. Accuracy using Lung Cancer dataset

In Fig.10, The accuracy of the proposed technique is better compared to the other techniques for all the varied percentages of data taken for training and testing. FAGA has an accuracy percentage between 70 to 80%. The accuracy percentages of all the other percentage are below 60%. This implies that our proposed technique classified data better than the other techniques.

4. Conclusion

The fractional order ABC and GA (FAGA) provides a novel SI-based algorithm for the efficient optimization. Using this algorithm, optimal data can be extracted out in less time. Likewise, it recuperates the required information in less effort when compared to other techniques. Unlike other techniques, this algorithm does not involve more mathematical calculation, at the same time to improve the existing algorithm mathematical validation using fractional calculus is carried out. This work has proved experimentally that proposed technique has obtained better results by acquiring better minimization and convergence rate. This algorithm is well-organized so that, it can be used for real time applications of image segmentation and data classification. The proposed FAGA algorithm can be employed to other applications where optimization leads to best required results.

References

1. P. Tsai, J. Pan, P. Shi and B. Liao, A New Framework for Optimization Based-On Hybrid Swarm Intelligence, *Handbook of Swarm Intelligence* (2011) vol. 8 421–449.
2. W. Deng, R. Chen, B. He, Y. Liu, L. Yin and J. Guo, A novel two-stage hybrid swarm intelligence optimization algorithm and application, *Soft Computing* (2012) vol. 6 1707-1722.
3. S. Kim, I. Kim, V. Mani, V and H.J. Kim, Ant Colony Optimization for SONET Ring Loading Problem, *International Journal of Innovative Computing, Information and Control* (2008) vol. 4 1617–1626.
4. S. Nakajima, H. Arimoto, H. Rensha and T. Toriu, Measurement of a Translation and a Rotation of a Tooth after an Orthodontic Treatment Using GA, *International Journal of Innovative Computing Information and Control* (2007) vol. 3 no. 6 (A) 1399–1406.
5. A. George, B. Rajakumar, D. Binu, Genetic algorithm based airlines booking terminal open/close decision system, *Proceedings of the International Conference on Advances in Computing, Communications and Informatics* (2012) pp. 174-179.
6. M. Subotic, Artificial bee colony algorithm with multiple onlookers for constrained optimization problems, *Proceedings of the 5th European conference on European computing conference* (2011) pp. 251-256.
7. B. Rajakumar, The Lion's Algorithm: A New Nature Inspired Search Algorithm, *Procedia Technology* (2012) vol. 6C 126-135.
8. D. Martens, B. Baesens and T. Fawcett, Editorial survey: swarm intelligence for data mining, *Journal Machine Learning archive* (2011) vol. 82 no. 1 1-42.
9. T. Hashni and T. Amudha, Relative Study of CGS with ACO and BCO Swarm Intelligence Techniques, *International Journal of Computer Technology & Applications* (2012) vol. 3, no. 5 1775-1781.
10. A. Garg, P. Gill, P. Rathi, Amardeep and K.Garg, An Insight into Swarm Intelligence, *International Journal of Recent Trends in Engineering* (2009) vol. 2, no. 8 42-44.
11. M. Mahant., B. Choudhary, A. Kesharwani and K. Rathore, A Profound Survey on Swarm Intelligence, *International Journal of Advanced Computer Research* (2012) vol. 2 no. 1 31-36.
12. A. George A and B. Rajakumar, Fuzzy Aided Ant Colony Optimization Algorithm to Solve Optimization Problem, *Advances in Intelligent Systems and Computing, Intelligent Informatics* (2013) vol. 182 207-215.
13. S. Binitha and S. Sathya, A Survey of Bio inspired Optimization Algorithms, *International Journal of Soft Computing and Engineering* (2012) vol. 2 no. 2 137-151.
14. L. Castro, Fundamentals of natural computing: an overview, *Physics Life Reviews* (2007) vol. 4, no. 1 1–36.
15. X. Gao, S.J. Ovaska and X. Wang, A GA-based Negative Selection Algorithm, *International Journal of Innovative Computing Information and Control* (2008) vol. 4, no. 4 971- 979.
16. J. Ruiz-Vanoye, O. Díaz-Parra, F. Cocón and A. Soto, Meta-Heuristics Algorithms based on the Grouping of

- Animals by Social Behavior for the Traveling Salesman Problem, *International Journal of Combinatorial Optimization Problems and Informatics* (2012) vol. 3 no. 3 104-123.
17. B. Shivakumar and T. Amudha, A Novel Nature-inspired Algorithm to solve Complex Generalized Assignment Problems, *International Journal of Research and Innovation in Computer Engineering* (2012) vol. 2 no. 3 280-284.
 18. J. Holland, *Adaptation in natural and artificial systems* (University of Michigan Press Ann Arbor, Mich. 1975).
 19. D. Goldberg, *Genetic algorithms in search, optimization and machine Learning*, (Addison-Wesley Publishing Co., Inc., Reading, Mass. 1989).
 20. R. Sharapov, *Genetic Algorithms: Basic Ideas, (Variants and Analysis, Vision Systems: Segmentation and Pattern Recognition, 2007)*.
 21. T. Singh and Z.M. Sandhu, An Approach in the Software Testing Environment using Artificial Bee Colony (ABC) Optimization, *International Journal of Computer Applications* (0975 – 8887) (2012) vol. 58, no. 21.
 22. E. Gerhardt E and H. Gomes, Artificial Bee Colony (ABC) Algorithm for Engineering Optimization Problems, *3rd International Conference on Engineering Optimization*, (2012).
 23. G .Lavanya and S. Srinivasan, "FAGA: Hybridization of Fractional Order ABC and GA for Optimization", *International Arab Journal of Information Technology*, Vol. 13, No.3. (Published Online)
 24. J. Vesterstrom and R. Thomsen, A Comparative Study of Differential Evolution, Particle Swarm Optimization, and Evolutionary Algorithms on Numerical Benchmark Problems, *Congress on Evolutionary Computation* (2004) vol. 2 1980-1987.
 25. J. Brest, S.W. Greiner, B. Boskovic, M. Mernik and V. Zumer, Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems, *IEEE Transactions on Evolutionary Computation* (2006) vol. 10 no. 6.
 26. D. Karaboga and B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, *J Glob Optim* (2007) vol. 39 459–471.
 27. D. Anghinolfi and M. Paolucci, A new discrete particle swarm optimization approach for the single-machine total weighted tardiness scheduling problem with sequence-dependent setup times, *European Journal of Operational Research* (2009) vol. 193 no. 1 73–85.
 28. M. Cheng and L. Lien, hybrid swarm intelligence based particle-bee algorithm for construction site layout optimization, *Journal Expert Systems with Applications* (2012) vol. 39 no. 10.