

# Biofilm algorithm for global numerical optimization

R. Vasundhara Devi, S. Siva Sathya

**Abstract:** *Swarm intelligence algorithms are based on the behavior and intelligence of living organisms that exist in nature. Amidst living organisms, Bacteria is a micro-organism that exhibit intelligent behavior by the development of biofilms to overcome harsh and adverse environment such as antibiotics and other bacteria. This survival behavior of bacteria forms the basis for the design of Biofilm (Bifi) algorithm in this paper. Biofilm algorithm uses three important characteristics of biofilm forming bacteria viz., conjugation, transformation and quorum sensing for solving real-world optimization problems. Biofilm algorithm is applied to global numerical benchmark test functions and compared with known state-of-the-art optimization algorithms.*

**Index Terms:** *Bacteria behavior, Biofilm, Swarm intelligence algorithm, Single objective optimization.*

## I. INTRODUCTION

Optimization problems in real life are often complex to solve as they have a number of objectives, constraints and decision variables. Bio-inspired algorithms have been proven as the best choice for such problems. Evolutionary algorithms and Swarm intelligence algorithms are important sub-categories of Bio-inspired algorithms; examples include the Genetic algorithm, which is inspired by the concept of natural evolution, particle swarm optimization, artificial bee colony algorithm, etc.[1]. A few of the swarm intelligence algorithms are discussed under section 2.

Living organisms learn from nature, and adapt and evolve slowly based upon the knowledge gained from the environment. Usually, organisms exhibit intelligent and coordinated behavior when they are in groups, such as a swarm of birds, a group of monkeys and a group of microorganisms like bacteria. The bacterial foraging optimization algorithm [2] is one such widely used algorithm based on bacterial behaviors. Bacteria exhibits extraordinary behavior by creating a Biofilm over themselves to survive on solid to semi-solid surfaces and protect themselves from life threatening circumstances, such as exposure to antibiotics or other competing bacteria [3]. Interestingly, the bacterial behavior inside the biofilm is distinct and unique in the sense that the bacterium is better equipped to explore and exploit the environment within the biofilm. The aim of this paper is to design a new meta-heuristic algorithm based on the behavior of bacteria within the biofilm and use the algorithm to solve

global numerical optimization problems. The rest of the paper is organized as follows: Section 2 offers some of the related work in the field of swarm intelligence algorithms, Section 3 presents the life cycle of bacteria with respect to Biofilm formation. Section 4 describes the proposed Biofilm algorithm. Implementation and experimental results of Bifi algorithm for global numerical benchmark test functions are analyzed and discussed in Section 5. Finally, section 6 concludes the paper with future directions.

## II. RELATED WORK

Swarm intelligence algorithms are based on the behavior and adaptation of animals, birds and other living organisms in nature. These simulated algorithms are categorized under artificial intelligence algorithms and are used to solve most of the real-world optimization problems. In recent times, there are several interesting swarm intelligence algorithms in literature. After Evolutionary algorithms, came the era of swarm intelligence algorithm and many hybrid algorithms. Some swarm intelligence algorithms are worth mentioning in this section. They include Particle swarm optimization [4] based on the social behavior of bird flocking or fish swarming, ant colony optimization by Dorigo is based on the foraging behavior of ants [5], Cuckoo search by Yang based on the Cuckoo's behavior of laying eggs in Crow's nest [6], Bat algorithm based on the echolocation of bats [7], Monkey algorithm based on the monkey behavior of searching food [8], bacterial foraging optimization simulates the foraging behavior of E. Coli bacteria [9]. Flower pollination algorithm simulates the pollination process in flowers [10]. Grey wolf optimizer based on the preying behavior of grey wolves [11]. The recent one to swarm intelligence algorithm category is the Whale optimization algorithm based on the behavior of whales [12]. Apart from these, there are a number of hybrid algorithms [13, 14, 15], proposed to carryout complex optimization problems.

## III. BACTERIA BEHAVIOR IN BIOFILMS

### A. Biofilm Characteristics

Biofilm[16] is a protective environment created by microorganisms such as bacteria, algae and fungi in a moist surface. The bacteria forming the biofilm and their behavior within the biofilm have been considered for designing a new algorithm in this paper.

**Manuscript published on 30 March 2019.**

\*Correspondence Author(s)

**R. Vasundhara Devi**, PhD. Degree, Department of Computer Science, Pondicherry University, (Pondicherry), India.

**Dr. S. Siva Sathya**, Associate Professor, Department of Computer Science, Pondicherry University, (Pondicherry), India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

Bacteria, initially moved with flagella in search of food and a better environment to live in. When such an environment is found, they adhere to that environment and develop a protective coverage called 'Biofilm' with the help of extracellular polymeric substances called as EPS [3] matrix. There are 3 phases in the life cycle of Bacteria regarding biofilm formation. They are attachment phase, maturation phase and dispersal and spreading phase. Fig. 1 shows the life cycle of bacteria and the different stages of biofilm formation.

- Attachment Phase

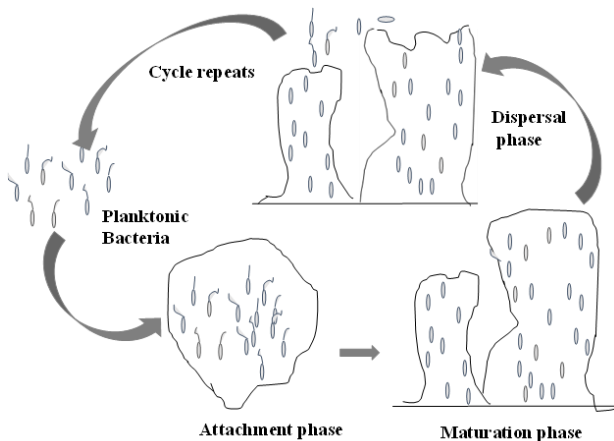
In the attachment phase, the planktonic bacteria with flagella attach itself to surfaces with a conducive environment for its growth. The bacteria move towards the food sources by chemotactic steps (swimming and tumbling) and attach themselves to the new environment. The beginning of the attachment phase is reversible which paves the way to the irreversible attachment phase and formation of the protective biofilm. In the Bifi algorithm, this behavior is simulated as an attachment phase where the bacteria move with the help of chemotactic steps.

- Maturation phase

When the environment is conducive, the bacteria within the biofilm enter the maturation phase. In the maturation phase, the bacteria reproduce asexually using fission reproduction and duplicates within the biofilm. In this phase, reproduction also takes place through horizontal gene transfer mechanisms such as conjugation [17] and transformation [17], and through limited mutations [18]. In Bifi algorithm, the characteristics such as fission reproduction, conjugation, transformation and mutation are simulated.

- Dispersal and spread phase

This phase is initiated when the bacterial population reaches a threshold limit due to insufficient food source for an



**Fig. 1. Life cycle of Bacteria in Biofilm**

increasing bacterial population within the biofilm. The bacteria uses Quorum sensing communication to determine the threshold limit and start the burst of biofilm to spread the bacteria to a new conducive environment. The bacteria inside the biofilm will have their own burst rate to disperse and spread. Once the biofilm bursts, the bacteria becomes

planktonic and start a new cycle of attachment phase in the adjacent biofilm-friendly environment. In Bifi algorithm, the quorum sensing and burst rates are used to simulate dispersal and spread phase.

### IV. BIOFILM (BIFI) ALGORITHM

#### A. Steps involved in Biofilm algorithm

The steps involved in the proposed Biofilm algorithm are given below:

**Population initialization:** The bacterial population of size 'nb' is initialized to random positions within the search space of the problem mentioned in Table I.

**Attachment phase:** When the bacteria are initialized in random starting positions, they move using chemotactic steps in the presence of chemical attractants and get attached to the surface using a step number 'n<sub>cs</sub>' and step size 'ss(i)' during the run; where, i = 1,2,..nb. The new position of the bacteria, 'NewB<sub>pos</sub>', is determined using equation-1. Here, B<sub>pos</sub> is the current position, Δ(i) represents the movement of the i<sup>th</sup> bacterium in random directions lying in the range of [-1,1].

$$NewB_{pos} = B_{pos} + ss(i) * \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (1)$$

**Fitness function evaluation:** At the end of the attachment phase, the numerical objective functions are evaluated for the new positions of the bacteria.

**Maturation phase:** In the maturation phase, the bacteria matures with the help of asexual or fission reproduction, conjugation and transformation techniques. In fission reproduction, genetically identical offsprings of the fittest parents are produced. Around 3% of total population size is made to reproduce asexually. In this algorithm, under bacterial conjugation reproduction process, P<sub>c</sub>, the conjugation probability is applied to fifty percent of the population size. Here, two bacterial parents are selected using tournament selection to exchange genetic information and form two offspring. Around, thirty percent of the total bacteria undergo the transformation. In this process, it produces the offspring from a single fit individual with the transformation probability, P<sub>t</sub>. Then for each generation of the bacteria, it evaluates the fitness function. Mutation process occurs with a probability of P<sub>m</sub> in the maturation phase to improve the diversity among the biofilm population.

**Dispersal and Spread phase:** In this phase, the sessile phase of the bacteria is over with the burst or the explosion of the biofilm, to release the bacteria. This is initiated through environmental cues such as quorum sensing rate (QS\_rate\_ind and QS\_rate\_gp). Burst speed 'B<sub>s</sub>' of the bacteria is generated randomly for each bacterium and plays an important role in the dispersal and spread of the bacteria to a new favorable environment. The dispersed bacteria again goes to the planktonic state and form the biofilm in a new environment.

Fig. 2 represents the steps of the Bifi algorithm schematically. Table II provides the pseudo-code of Biofilm algorithm and Table III shows their sub functions.

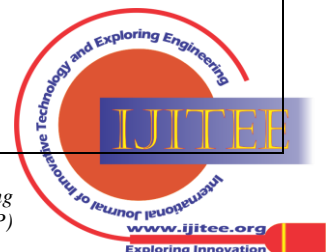


Table I. Numerical test functions and details

S. No	Function name	Mathematical Functions	Range search	of	Global optimum
1	Ackley	$f(x) = 20 + e - 20 \exp \left[ -2.0 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right] - \exp \left[ \frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right]$	[-32,32]		0
2	Mathyas	$f(x) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$	[-10,10]		0
3	Rastringin	$f(x) = 10d + \sum_{i=1}^d x_i^2 - 10 \cos(2\pi x_i)$	[-5.12,5.12]		0
4	Sphere	$f(x) = \sum_{i=1}^d x_i^2$	[-100,100]		0

Table II. Pseudo code of Biofilm algorithm

Algorithm – Biofilm algorithm
<p><b>Begin</b></p> <p><b>Input:</b> <math>f_o(x)</math> :Objective function; nb: Number of Bacteria; c: step size; <math>n_{cs}</math>: Number of Chemotactic step; <math>P_c</math>: Conjugation probability; <math>P_t</math>: Transformation probability; <math>P_m</math>: Mutation probability; br: Burst_rate; Termination condition : Number of dispersals (no_dis)</p> <ol style="list-style-type: none"> <li>1. Initialize the Bacteria(nb) positions</li> <li>2. Calculate the fitness of the bacteria based on the objective function <math>f_o(x)</math></li> <li>3. While (Termination condition not met) //Attachment phase//</li> </ol> <p>For each chemotactic step (<math>n_{cs}</math>) and each bacteria{</p> <p>Calculate the position increment using Equation (1)</p> <p>New_position = current_position + position increment</p> <p>Move the bacteria using the chemotactic steps</p> <p>Return ( new position)}</p> <p>//Maturation Phase://</p> <p>For each generation</p> <p>Generate random_probability ‘rand_prob’</p> <p>For each individual bacteria,{</p> <p>Return Parent 1 and 2 using Tournament_selection()</p> <ol style="list-style-type: none"> <li>a. Create duplicates for the two best fit Bacteria by fission reproduction</li> <li>Add these 4 individuals to the new generation</li> <li>b. If <math>P_c &gt; \text{rand\_prob}</math></li> </ol> <p>Conjugation_HGT(Parent1,Parent2)</p> <p>Add Children to New population</p> <p>Evaluate the fitness of the children</p> <ol style="list-style-type: none"> <li>c. If <math>P_t &gt; \text{rand\_prob}</math></li> </ol>



## Biofilm algorithm for global numerical optimization

```

Transformation_HGT(Parent1,Parent2)
Add Children to New population
Evaluate the fitness of the children}
If  $P_m > \text{rand\_prob}$ {
Call mutate()
Add mutated individual to New population
Evaluate the mutated individual}
Sort the new population and trim the list to the size of nb, discard the least fit //Dispersal phase starts: //
For each individual bacteria, {Generate burst rate in a random manner
    Call Disperse_Bifi()
    Evaluate the fitness function for each bacteria}
4. End While
    
```

**Table III. Sub functions of Biofilm algorithm**

Sub Functions	
<p><b>Function Conjugation_HGT</b>  <b>Input:</b> Two parents, crossover point  <b>Output:</b> Two children                      Begin                      Calculate crossover point                      Based upon the crossover point                      Split parent a into a1,a2                      Split parent b into b1,b2                      Apply crossover of a and b at crossover points                      Child 1 &lt;----- Combination of a1 &amp; a2 (Same as parent a)                      Child 2 &lt;----- Combination of b1 &amp; a2 or a1 &amp; b2                      End</p>	<p><b>Function Transformation_HGT</b>  <b>Input:</b> Two parents, one best fit bacteria  <b>Output:</b> One bacteria                      Begin                      Randomly pick parent 1, pick the elite bacteria as parent2                      Calculate the crossover point                      Based upon the crossover point                      Split parent a into a1,a2                      Split parent b into b1,b2                      Child &lt;----- a1 &amp; b2                      End</p>
<p><b>Function mutate()</b>  <b>Input:</b> One individual // any x or y position chosen randomly to induce mutation//  <b>Output:</b> One new bacteria                      Begin                      Choose a random point as a mutation point                      Make random changes at the mutation point                      Return the new bacteria                      End</p>	<p><b>Function Disperse_Bifi()</b>  <b>Input:</b> One bacteria, burst speed with which the bacteria travel after the dispersal. Number of bacteria, Set burst speed minimum and maximum  <b>Output:</b> New position to the bacteria in the global domain                      Begin                      Initialize burst speed (Bs) to zero.                      Generate random values for the burst speed by <math>Bs = \text{rand} * br</math>  <math>\text{new } Bs = Bs + QS\_rate\_ind * QS\_rate\_gp * \text{current\_bacteria\_pos}</math>  <math>\text{new\_bacteria\_pos} = \text{current\_bacteria\_pos} + \text{new\_Bs}</math>                      Return new_bacteria_pos                      End</p>

### V. EXPERIMENTAL RESULTS AND DISCUSSION

The Biofilm optimization algorithm is implemented using MATLAB [19] software. This work is done in HP laptop with 4.0 GB RAM, Intel(R) core™ i3- 5005U CPU @ 2.00 GHz processor.

#### A. Mathematical benchmark test functions

The Biofilm algorithm's performance is illustrated with the help of four global numerical test functions [20] . The mathematical benchmark test functions are used as an indicator to study the performance of the new algorithm.

Table I provides the details of the benchmark test functions. The benchmark test functions used are: Ackley, Matyas, Rastrigin and Sphere functions.

#### B. Mathematical benchmark test functions

An initial bacterial population of size 60 is taken in the Biofilm algorithm. The lower and upper bound values corresponding to the benchmark function is utilized to generate the bacterial population.

The number of chemotactic steps is 200 before irreversible attachment to the surface. While the number of generations for the maturation phase is fixed at 5.

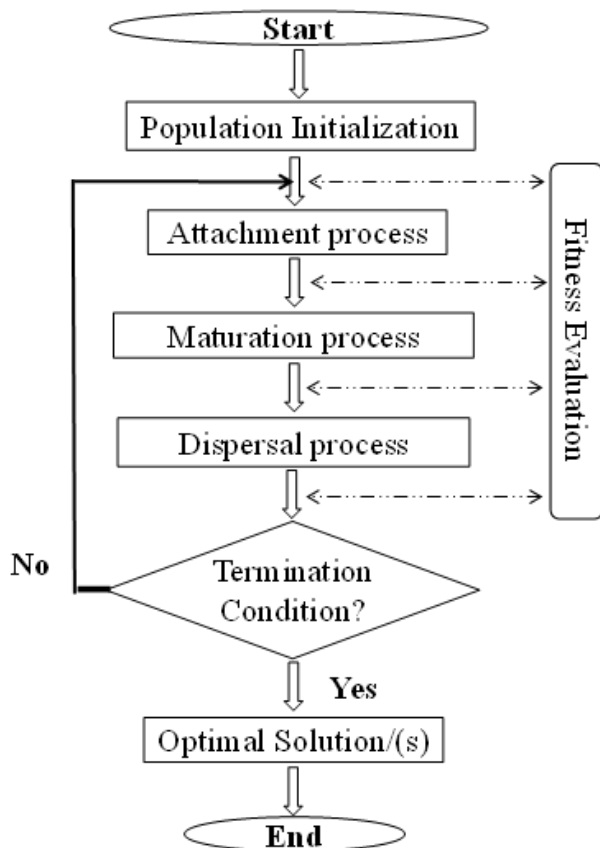


Fig. 2. Flowchart of Biofilm algorithm

In the maturation phase, the number of individuals that have to undergo fission reproduction are set as two fittest bacteria; conjugation, transformation and the intragenic mutation are taken as 0.5, 0.3 and 0.1 respectively of the entire population. The number of dispersal for the biofilm is fixed as 3 for the dispersal and spread phase. The burst speed is generated randomly and assigned to each individual. The range of distance the bacteria can travel after the dispersal depends upon the burst speed. This is limited within the range of maxdist and mindist to limit the bacteria within the problem space and improve exploitation and exploration

within the problem space. The mean time taken by the Bifi algorithm for solving the test functions from 50 runs is shown in Table IV.

Table IV. Time taken by Bifi algorithm for benchmark test functions

S.No	Benchmark test functions	Time taken (Mean ± SD) in Seconds
1	Ackley	1.6951 ± 0.0390
2	Matyas	2.9162 ± 0.1231
3	Rastrigin	3.4048 ± 0.1289
4	Sphere	2.6713 ± 0.0413

With the current experimental setup, the time taken shows that Bifi algorithm is comparatively competitive with its counterparts.

### C. Comparison of Bifi algorithm with PSO, BFO, CS and BA

The results from Bifi algorithm were compared with the well-known state-of-art algorithms such as Particle swarm optimization (PSO), Bacterial foraging optimization algorithm (BFO), Cuckoo search (CS) and Bat algorithms (BA). Table V compares the fitness values obtained for various benchmark test functions using different algorithms from 50 runs. The basic particle swarm optimization parameters taken are as follows: inertia weight is 1, personal and global learning coefficients are 1.5 and 2.0 respectively. The parameter values of BFO are fixed as follows: reproductive steps = 4, elimination steps = 4 and elimination probability= 0.5. The parameters of CS, viz., rate of discovery for an alien egg is 0.25. The parameter values of BA such as loudness rate and the pulse rate are taken as 0.5; the frequency range is taken as 1 and 2 that depicts the minimum and maximum frequency values. Fig. 3 shows the convergence of fitness for various benchmark test functions when different algorithms were in use.

Table V. Comparison of test results of Bifi algorithm with PSO, BFO, CS and BA.

Test functions	PSO	BFO	CS	BA	Bifi
<b>Ackley</b>					
Iterations	600	600	600	600	600
Fitness (Mean ± SD)	0.13021 ± 0.08946	1.61E-04 ± 0.00014	1.8622 ± 0.9782	1.2641 ± 1.3835	2.56E-05 ± 0
<b>Matyas</b>					
Iterations	300	300	300	300	300
Fitness (Mean ± SD)	2.03E-04 ± 0.00020	1.01E-06 ± 8.33E-07	0.00223 ± 0.00278	0.00164 ± 0.00308	1.6E-09 ± 0
<b>Rastrigin</b>					
Iterations	600	600	600	600	600

## Biofilm algorithm for global numerical optimization

Fitness (Mean $\pm$ SD)	1.0446 $\pm$ 0.85598	-9.9563 $\pm$ 0.19433	0.89825 $\pm$ 0.67277	1.62354 $\pm$ 1.22036	4.16E-10 $\pm$ 0
<b>Sphere</b>					
Iterations	900	900	900	900	900
Fitness (Mean $\pm$ SD)	2.70E-07 $\pm$ 4.84E-07	7.33E-10 $\pm$ 1.00E-09	0.04085 $\pm$ 0.06930	0.00844 $\pm$ 0.02473	8E-10 $\pm$ 0

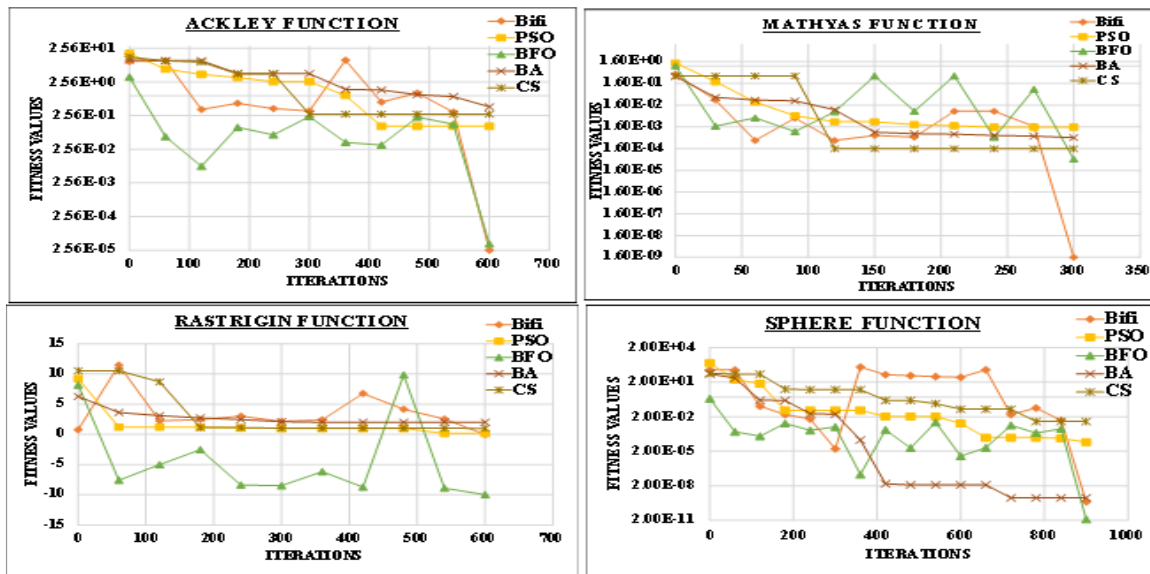
The unimodal and multi-modal benchmark test functions in the two-dimensional space were taken for experimentation. To do statistical analyses and compute the mean fitness and standard deviation (SD), each of the algorithms was executed 50 times. The number of iterations for the respective benchmark test functions are maintained the same in this comparison. The benchmark test functions have a global optimum value of 0. In every run of Bifi algorithms, the attained values are very close to the global optimal values in a minimal number of iterations when compared to PSO, BFO, CS and BA algorithms. In Fig. 3, for Ackley function, the Bifi algorithm and the BFO provide optimal values in the range of  $10^{-05}$ ; however, the Bifi algorithm show better mean fitness value. For Matyas function, the fitness value obtained using Bifi algorithm is in the range of  $10^{-09}$  and is better than that

obtained using other algorithms. In the case of the Rastrigin function, except for BFO, the fitness value reaches close to global optimal value. Also for the sphere function, the Bifi algorithm provides better mean fitness value in the range of  $10^{-10}$ , as compared to other algorithms. The results show that the Bifi algorithm is robust and better compared to other algorithms studied. It could even be improved further by fine-tuning the parameters. Hence, the Bifi algorithm can be considered a useful addition to the existing repertoire of existing swarm intelligence algorithms.

### VI. CONCLUSION

This paper proposed a new swarm intelligence algorithm called Biofilm (Bifi) algorithm that simulates intelligent bacterial behavior within biofilms. This meta-heuristic algorithm has been implemented and applied to global numerical benchmark test functions. The results of Bifi algorithm are compared with a few popular algorithms such as Particle swarm optimization, Bacterial foraging algorithm, Cuckoo search algorithm and Bat algorithm. The comparison shows that Bifi is better than other meta-heuristic algorithms used in this study. In the future, it is planned to tune the parameters of Bifi algorithm to apply to large scale, multi-dimensional real-life problems. Also, the development of a multi-objective version of Bifi algorithm by simulating a few more characteristics of the bacteria in Biofilm is under process.

Fig. 3. Experimental results of mathematical benchmark test functions over different algorithms



REFERENCES

- G. Amirthalingam and G. Radhamani, "New chaff point based fuzzy vault for multimodal biometric cryptosystem using particle swarm optimization," J. King Saud Univ. - Comput. Inf. Sci., vol. 28, no. 4, pp. 381–394, 2016.
- K. M. Passino, "Bacterial Foraging Optimization," International Journal of Swarm Intelligence Research, vol. 1, no. 1, pp. 1–16, 2010.
- H. H. and S. Lackner, "Modeling of Biofilm Systems: A Review," Adv. Biochem. Eng. Biotechnol., vol. 146, no. July 2015, pp. 127–141, 2014.
- R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization An overview," Swarm Intell., no. 1, pp. 33–57, 2007.
- K. D. Narooei, R. Ramli, M. Nizam, A. Rahman, F. Iberahim, and J. A. Qudeiri, "Tool Routing Path Optimization for Multi-Hole Drilling Based on Ant Colony Optimization," World Appl. Sci. J., vol. 32, no. 9, pp. 1894–1898, 2014.
- S. Yang, X.-S., and Deb, "Engineering Optimisation by Cuckoo Search," Int. J. Math. Model. Numer. Optim., vol. 1, no. 4, pp. 330–343, 2010.
- X. S. Yang, "A new metaheuristic Bat-inspired Algorithm," Stud. Comput. Intell., vol. 284, pp. 65–74, 2010.
- R. V. Devi and S. S. Sathya, "Monkey Behavior Based Algorithms - A Survey," Int. J. Intell. Syst. Appl., vol. 9, no. 12, pp. 67–86, 2017.
- S. Das, A. Biswas, S. Dasgupta, and A. Abraham, "Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Applications," Found. Comput. Intell. Vol. 3, vol. 3, pp. 23–55, 2009.
- A. Fouad and X. Z. Gao, "A novel modified flower pollination algorithm for global optimization," Neural Computing and Applications, pp. 1–34, 2018.
- S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," Adv. Eng. Softw., vol. 69, pp. 46–61, 2014.
- S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," Adv. Eng. Softw., vol. 95, pp. 51–67, 2016.
- N. Y. ,AK Hasan MAKAS, "Balancing exploration and exploitation by using sequential execution cooperation between artificial bee colony and migrating birds optimization algorithms.pdf," Turk J Elec Eng Comp Sci, pp. 4935–4956, 2016.
- S. M. Abd-Elazim and E. S. Ali, "A hybrid Particle Swarm Optimization and Bacterial Foraging for optimal Power System Stabilizers design," Int. J. Electr. Power Energy Syst., vol. 46, no. 1, pp. 334–341, 2013.
- P. Kora and S. R. Kalva, "Hybrid Bacterial Foraging and Particle Swarm Optimization for detecting Bundle Branch Block," Springerplus, vol. 4, no. 1, 2015.
- F. El Moustaid, "Mathematical Modelling of Bacterial Attachment to Surfaces: Biofilm Initiation," no. September, pp. 6–8, 2011.
- K. M. Derbyshire and T. A. Gray, "Distributive Conjugal Transfer: New Insights into Horizontal Gene Transfer and Genetic Exchange in Mycobacteria Keith," vol. 2, no. 1, pp. 1–32, 2014.
- M. B. H. Najafi and P. Pezeshki, "Bacterial Mutation; Types, mechanisms and mutant detection methods: A Review," vol. 4, no.

- December, pp. 628–638, 2013.
- "Matlab software," Matlab, high-performance numeric computation and visualization software: reference guide, 1992. [Online]. Available: [https://in.mathworks.com/downloads/web\\_downloads/?s\\_iid=hp\\_ff\\_t\\_downloads](https://in.mathworks.com/downloads/web_downloads/?s_iid=hp_ff_t_downloads).
- M. Jamil and X. Y. Blekinge, "A Literature Survey of Benchmark Functions For Global Optimization Problems," pp. 1–47, 2013.

AUTHORS PROFILE



**R. Vasundhara Devi** received M. Tech Degree from Pondicherry University. Currently pursuing Ph.D. Degree in Computer Science at Pondicherry University, Pondicherry, India. Her research interests include optimization algorithms, interdisciplinary applications, Databases and Data mining.



**Dr. S. Siva Sathya** is an Associate Professor in the Department of Computer Science, Pondicherry University. Her areas of interests include evolutionary algorithms, bioinformatics, intrusion detection, etc. She has to her credit a number of research papers in international journals and conferences. She is the recipient of Nari Shakti Puraskar award 2018

