Chapter 3

An Extensive Study and Performance Analysis of Chemo-GA over the Benchmark Functions

In this Chapter, the efficiency and robustness of the algorithm is studied in an extensive manner by taking a wide set of benchmark problems. In the previous Chapter, the recommendation of parameters of CGA has been done over 4 benchmarking functions. CGA has been compared with GA and its hybridized version i.e. Genetic Algorithm Hybridized Bacterial Foraging Optimization (GA-BF). In the current Chapter, to test the further efficiency and robustness of CGA over other popular hybridized algorithms, the experiments are performed by taking 22 problems having various types of complexities and nonlinearities.

The objective of this Chapter is twofold. They are defined as follows. (i)The algorithm is being compared with other popular hybridized algorithm i.e. Quadratic Approximation Hybridized Genetic Algorithm (QGA), where Quadratic Approximation operator has been inserted in GA cycle. (ii) The second objective is to study the parameters of CGA in an extensive and effective manner to satisfy a wide set of problems. The efficacy of the algorithm is realized by comparing CGA with QGA in terms of better objective function value, success rate, standard deviation which are shown in tabular form.

3.1 INTRODUCTION

In real life situations the nature of the objective function is highly non linear, multimodal and high dimensional. There arises several types of difficulties in finding the global minima (or maxima) of a given function to be optimized. This has been illustrated in Fig. 3.1. Looking through the figure, it is observed that P is the global minima of the given function which is in a deep valley surrounded by tall hills. Q is the promising local minima occurring in the neighborhood of the point P. There is chance of getting trapped into Q before reaching P. R is the shallow basin. Excessive iterations are needed to reach the point P and Q for crossing the basin. There is another local minima situated at point S which may mislead the search as it is far away from the global minima.

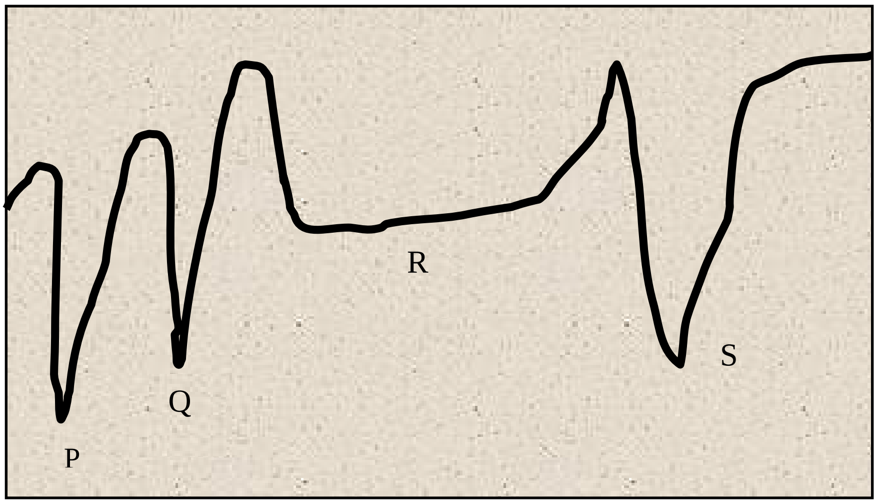

Fig.3.1: Pictorial diagram of a multi modal non linear function

Finding the global minima of such complex function is highly challenging task. So, there is need of robust optimization techniques to deal such complexities. According to Wolpert and Macready (1997), all algorithms designed for finding the extremum of the cost function performs equally well when averaged over all the cost functions. According to the authors, if algorithm A outperforms algorithm B in some optimization problems, there exist as many other problems where algorithm B outperforms algorithm A. Hence, from the problem solving perspective no single state-of-the-art algorithm can handle all sorts of optimization problems.

That's why; Now-a-days researchers are more focused on hybridized techniques. In this regard, already in (Chapter 2), the hybridized algorithm CGA has been designed. So, in this regard, the current Chapter is an extension of Chapter 2 which aims at handling all sorts of complexities lying in the benchmark problems.

The current Chapter is organized as follows. In the following Section the description of Chemo-GA (CGA) is discussed in brief.

The experimental setup of CGA, the result comparison of CGA with Quadratic Approximation Hybridized Genetic Algorithm (QGA) and the corresponding result discussion are explained in section 3.3. The conclusion of the Chapter is drawn in section 3.4.

3.2 DESCRIPTION OF CHEMO-GA (CGA)

Brief on GA and BFO:

Since the present study deals with the hybridization of some operators of BFO and GA; a brief outline about them are presented. GA is a population based method consisting of 4 major steps namely Selection, Crossover, Mutation and Elitism. Based on the Darwin's principle of survival of fittest, GA selects the better individuals in the population with letting the worse die off. Similarly, BFO also works with a population of individuals through its major steps like Chemo taxis, Reproduction, and Elimination-Dispersal and Swarming effect. However, each of them attempts to explore the search space and results with a near optimal solution, if not the optimal one. Each one has some pitfalls in their inherent mechanism. Researchers (Chen et al., 2007; Kim et al., 2007, Long et al., 2010; Panda and Naik, 2012) tried to overcome them by the process of hybridization in solving unconstrained optimization problems.

Inspired by the hybridization (Kim et al., 2007), an attempt has been made in (Chapter 2) to design hybridized algorithm of GA and BFO. The motivation of this hybridization is already discussed in Section 2.3(Chapter 2) that can be referred.

So, the cycle of the proposed algorithm CGA is given (in brief) in Fig 3.2.

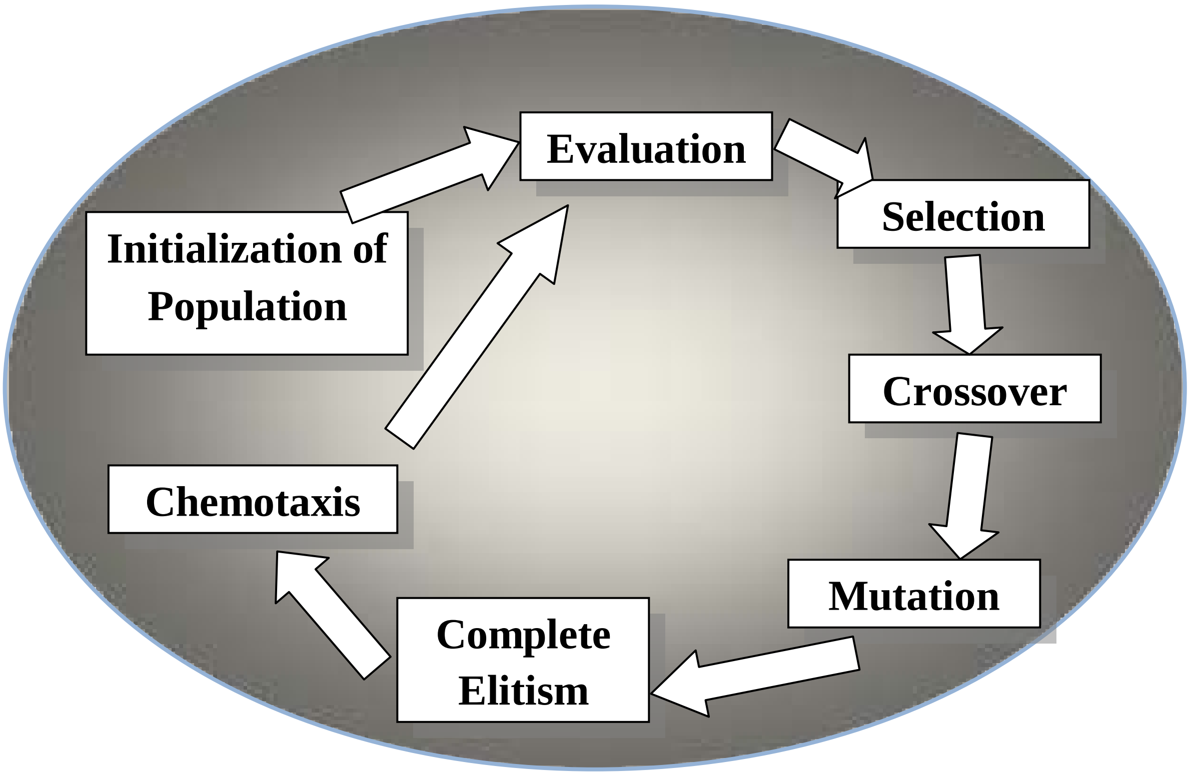

Fig. 3.2: CGA Cycle

The stepwise working principle of CGA is given in detail in (Section 2.4(subsection 2.4.1)) which can be reffered.

3.3 BENCHMARK PROBLEMS

A set of 22 benchmark problems has been taken from literature (Deep and Das, 2008) and listed in Table 3.1. Each function included in this set is a scalable problem i.e. the dimesion of the problem depends upon the user choice.

Table 3.1: List of test functions with the bounds of the decision variables

S. N.	Name of Function	Function	Bounds
1	**Ackley**	$-20\exp\left(-0.02\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)$ $+20+e$	[-30, 30]
2	**Cosine Mixture**	$0.1n+\sum_{i=1}^{n}x_i^2-0.1\sum_{i=1}^{n}\cos(5\pi x_i)$	[-1, 1]
3	**Exponential**	$1-\left(\exp\left(-0.5\sum_{i=1}^{n}x_i^2\right)\right)$	[-1, 1]
4	**Griewank**	$1+\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)$	[-600,600]
5	**Levy & Mantalvo-1**	$\frac{\pi}{n}\left(10\sin^2(\pi y_1)+\sum_{i=1}^{n-1}\begin{matrix}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})]+\\(y_n-1)^2\end{matrix}\right.$ $y_i=1+\frac{1}{4}(x_i+1)$	[-10,10]
6	**Levy & Montalvo-2**	0.1 $\left(\left(\sin^2(3\pi x_1)+\sum_{i=1}^{n-1}\begin{matrix}(x_i-1)^2[1+\sin^2(3\pi x_{i+1})]\\+(x_n-1)^2[1+\sin^2(2\pi x_n)]\end{matrix}\right)\right)$	[-5, 5]
7	**Paviani**	$45.778\sum_{i=1}^{10}[(\ln(x_i-2))^2+(\ln(10-x_i))^2]-\left(\prod_{i=1}^{10}x_i\right)^{0.2}$	[2, 10]
8	**Rastrigin**	$10n+\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)]$	[-5.12,5.12]
9	**Rosenbrock**	$\sum_{i=1}^{n-1}[100(x_{i+1}-x_i^{\ 2})^2+(x_i-1)^2]$	[-30,30]
10	**Schwefel**	$418.9829n-\sum_{i=1}^{n}x_i\sin\left(\sqrt{\lvert x_i\rvert}\right)$	[-500,500]
11	**Sinusoidal**	$3.5-[2.5\prod_{i=1}^{n}\sin(x_i-\pi/6)+\prod_{i=1}^{n}\sin(5(x_i-\pi/6))$	[0, π]
12	**Zakharov's**	$\sum_{i=1}^{n}x_i^{\ 2}+\left(\sum_{i=1}^{n}\frac{i}{2}x_i\right)^2+\left(\sum_{i=1}^{n}\frac{i}{2}x_i\right)^4$	[-5.12,5.12]

13	**Sphere**	$\sum_{i=1}^{n} x_i^2$	[-5.12,5.12]
14	**Axis parallel hyper ellipsoid**	$\sum_{i=1}^{n} i x_i^2$	[-5.12,5.12]
15	**Schewefel-3**	$\sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	[-10,10]
16	**Neumaier-3**	$\frac{n(n+4)(n-1)}{6} + \sum_{i=1}^{n} (x_i - 1)^2 - \sum_{i=2}^{n} x_i x_{i-1}$	$\left[-n^2, n^2\right]$
17	**Salomon**	$1\text{-}\cos\left(2\pi \lVert x \rVert\right) + 0.1 \lVert x \rVert, \lVert x \rVert = \sqrt{\sum_{i=1}^{n} x_i^2}$	[-100, 100]
18	**Ellipsoidal**	$\sum_{i=1}^{n} (x_i - i)^2$	$[-n, n]$
19	**Schaffer-1**	$0.5 + \left(\left(\sin \sqrt{\sum_{i=1}^{n} x_i^2}\right)^2 - 0.5\right) / \left(1 + 0.001\left(\sum_{i=1}^{n} x_i^2\right)\right)^2$	[-100,100]
20	**Brown-3**	$\sum_{i=1}^{n-1} [(x_i^2)^{(x_{i+1}^2 + 1)} + (x_{i+1}^2)^{(x_i^2 + 1)}]$	[-1, 4]
21	**New function**	$\sum_{i=1}^{n} \left(0.2 x_i^2 + 0.1 x_i^2 \sin 2x_i\right)$	[-10,10]
22	**Cigar**	$x_1^2 + 100{,}000 \sum_{i=2}^{n} x_i^2$	[-10,10]

3.3.1 Experimental Set up

The proposed CGA program code is designed in C++ and the experiment is carried out on a P-IV, 2.8 GHz machine with 512 MB RAM under WINXP platform. In the recent past Deep and Das (2008) designed 4 versions of GAs, depending on all possible combination of selection (Tournament and Roulette wheel) and crossover (One-point and Uniform) GA-operators. They named them as GA1: that uses Tournament selection and one-point crossover; GA2: that uses Tournament selection and uniform crossover; GA3: that uses Roulette wheel selection and one-point crossover; and GA4: that uses Roulette wheel selection and uniform crossover. These 4 versions are further hybridized with

Quadratic approximation to get more 4 versions namely HGA1, HGA2, HGA3 and HGA4 respectively. In (Deep and Das, 2008), for each of 22 benchmark problems (Table 3.1), best version out of above 8 versions/algorithms is reported. In this section, the efficiency of CGA is realized over the best reported version in (Deep and Das, 2008) for each benchmark problems.

For a fair comparison, the ranges of the decision variables remain unchanged. Problem size for each problem is fixed at 10 and the population size is at 60. A total of 100 runs are conducted for each of CGA and HGAs. A run is said to be a success if the value obtained by the algorithm is within 1% accuracy of the known optimal solution. But for Rosenbrock, Schwefel, Neumier-3 and Salomon it is fixed at 3, 0.4, 2, and 0.1 respectively. The stopping criteria are a maximum of 1000 generations or if no improvement is observed in consecutive 100 generations. But for Rosenbrock function, the maximum generation is considered as 10000. The chemo tactic loop is allowed after 100 generations for 6 functions i.e. Ackley, Sinusoidal, Levymontalvo-1, Levymontalvo-11, Griewank and Cigar function and for the remaining functions after 10 generations. In chemo tactic loop, only 4 bacteria are reinitialized within the maximum and minimum range of the search space and allowed to exploit the search operation in a better way.

After a series of hand tuning experiments, the recommended values of the parameters are as follows. Probability of crossover $\left(P_c = 0.9\right)$, probability of mutation $\left(P_m = 0.01\right)$, maximum step size $\left(C_{\max} = 0.1\right)$, minimum step size $\left(C_{\min} = 0.008\right)$ and number of Chemo tactic step $\left(N_c = 40\right)$. But due to the complex behavior, for Neumier-3 function, $N_c = 1500$ is considered. With the recommendation of the above parameter and fixation of above experimental set up, the simulation results are recorded, which is discussed in the next section.

3.3.2 Analysis of Result

The average minimum function value, success rate (S. R), function evaluation, Standard deviation (S.D), CPU time of 100 independent runs both for CGA and HGAs/ GAs are presented in Table 3.2. The best result for solving each problemis highlighted by the bold face figures. To bold the best result, two major parameters have been considered as follows.

i. The average objective function value

ii. If there is a tie in (i), next priority is given to 'success rate'.

Form Table 3.2 it is observed that CGA is better in terms of either better optimal solution or higher success rate than HGAs/GAs. Out of 22 problems under consideration, CGA outperforms with better function values in 16 cases, tie occurs in 3 cases. In rest three, it could not perform well. However, the results are comparable. In the other hand, CGA also solved most of the problems with higher success rate. Except in problem 16, everywhere CGA yields less standard deviation to ensure the stability of the system. Similarly, the CPU time for CGA is better in 15 problems as compared to HGAs/GAs. In return, it takes little more number of function evaluations. The better function values, higher success rates, less standard deviations, less time and less number of function evaluations are reported in bold face letters in Table 3.2.

Table 3.2: Comparative results of CGA and HGAs/GAs (Deep and Das, 2008)

S.I.	Fun. Name	Fun. Value	S.R.	S.D.	Time (Sec.)	Fun. Eval.
1	**HGA3**	1.67E-08	**100**	1.32E-08	4.06	**28757**
	CGA	**7.70e-15**	**100**	**9.85e-15**	**2.62**	112064
2	**HGA3**	2.00E-015	**100**	1.63E-15	2.44	**16749**
	CGA	**2.22E-18**	**100**	**1.56E-17**	**1.8**	109477
3	**HGA3**	2.49E-16	**100**	1.33E-16	2.00	**15849**
	CGA	**4.88E-17**	**100**	**5.51E-17**	**1.33**	71885
4	**GA4**	3.53E-03	**23**	4.43E-03	**8.04**	**26204**
	CGA	**1.23E-03**	8	**3.26E-03**	16.2	197492
5	**HGA**	**1.18E-12**	**100**	9.17E-16	**3.40**	**16023**
	CGA	5.23E-09	98	**1.16E-21**	4.48	219182
6	**HGA3**	**1.89E-13**	**100**	4.62E-17	**2.02**	**13418**
	CGA	1.67E-09	83	**9.52E-22**	3.56	155836
7	**HGA3**	4.70E-04	**100**	8.41E-14	3.00	24872
	CGA	**2.64E-14**	**100**	**2.67E-14**	**0.15**	**18802**
8	**HGA3**	4.73E-13	35	2.64E-13	4.80	**34276**
	CGA	**2.72E-15**	**100**	**1.67E-15**	**3.25**	276187
9	**HGA3**	6.92E-01	**100**	1.75E+00	**88.4**	**309516**
	CGA	**4.42E-05**	78	**4.78E-05**	181	3.82E+06
10	**HGA4**	**1.27E-04**	75	3.45E-12	4.61	**28134**
	CGA	**1.27E-04**	**99**	**3.55E-13**	**4.2**	75782

11	**HGA3** **CGA**	3.02E-15 **5.86E-16**	**100** 91	1.23E-15 **2.70E-16**	2.33 **1.37**	**19580** 58993
12	**HGA3** **CGA**	2.04E-14 **1.33E-22**	**100** **100**	1.01E-14 **4.27E-23**	3.00 **1.17**	**17088** 156966
13	**HGA3** **CGA**	2.69E-15 **6.36E-41**	**100** **100**	1.49E-15 **1.49E-41**	2.27 **1.06**	**16773** 169683
14	**HGA3** **CGA**	2.89e-15 **3.495E-40**	**100** **100**	1.59E-15 **1.32E-40**	2.16 **1.72**	**16703** 321818
15	**HGA3** **CGA**	3.83E-07 **1.71E-08**	**100** **100**	1.21E-07 **1.84E-08**	**3.00** 3.42	**19684** 197641
16	**HGA4** **CGA**	**7.17E-04** 5.007E-01	**100** 92	**2.98E-04** 1.44E-01	14.3 **6.01**	**98728** 1.03E+06
17	**GA4** **CGA**	**9.99E-02** **9.99E-02**	35 **44**	3.37E-07 **1.39E-17**	7.71 **3.83**	**28657** 216620
18	**HGA3** **CGA**	1.06E-14 **1.80E-31**	**100** **100**	6.74E-15 **2.86E-31**	2.94 **0.92**	**17311** 71123
19	**HGA3** **CGA**	**9.72E-03** **9.72E-03**	16 **34**	5.89E-17 **2.62E-17**	**2.81** 3.06	**25799** 350332
20	**HGA4** **CGA**	5.96E-16 **3.05E-29**	**100** **100**	2.12E-16 **7.99E-30**	3.00 **1.72**	**16466** 146612
21	**HGA3** **CGA**	1.85E-15 **4.69E-041**	**100** **100**	9.42E-16 **1.12E-41**	2.47 **1.18**	**16781** 187579
22	**HGA3** **CGA**	6.39E-10 **2.03E-12**	**100** **100**	3.46E-10 **2.94E-12**	**3.00** 11	**18113** 1.17E+08

3.4 CONCLUSION

In this Chapter an attempt has been made to study hybridization of chemo tactic step in GA cycle in an effective manner on a wide set of benchmark functions. The parameter settings of CGA have been recommended based on wide set problems for further use throughout the thesis for unconstrained optimization problem. It is observed from the result and discussion that Chemo-GA gives efficient and challenging result as compared to Quadratic Approximation based hybrid genetic algorithm and simple GA, not only in terms of better objective function value but also in terms of less computational time for most of the functions. Though the function evaluation is more, it consumes less computational time in return. Based on this study, it is concluded that CGA is a better candidate in order to enhance the performance of GA in the field of function optimization.