COMPREHENSIVE ANALYSIS OF STOCHASTIC ALGORITHMS: AN ALTERNATIVE IN OPTIMIZATION

Syed Asadullah Hashmi
Department of Mathematics, Deccan College of Engineering and Technology,
Hyderabad, India.
hashmisyedasad@gmail.com

ABSTRACT

Global optimization is an important branch of computational mathematics that finds the applications in every walk of life. The real-world science and engineering optimization applications are becoming more and more complex in nature and are generally multimodal. The conventional optimization methods fail to optimize such complex multimodal problems. Hence there is always an increasing demand for efficient and robust optimization strategies. In recent years, algorithms based on random process have become popular and are an alternative to conventional methods in optimization. The Stochastic Algorithms are based on random process and most of them are the extraction of natural phenomenon for problem solving. The objective of this paper is to investigate and analyze stochastic algorithms on complex multimodal optimization problems. The comprehensive analysis of different stochastic algorithms is carried out on a set of standard benchmark problems with 10, 30 and 50 dimensions. The algorithmic suitability, robustness and convergence rate of each will be investigated. Finally the dependency of Stochastic Algorithms on problem dimensions are also discussed..

KEYWORDS

Evolutionary algorithm, swarm intelligence, multimodal

1. INTRODUCTION

Global optimization is the branch of computational mathematics and it's objective is to select the best possible decision for a given set of circumstances. In recent years the subject of optimization has matured and is widely used in numerous applications, e.g., petroleum refining, routes for aircrafts, missile trajectories, business, physical, chemical and biological sciences, engineering, architecture, economics, and management. The range of techniques available to solve them is nearly as wide. The conventional optimization methods are based on the mathematical structure of the problem under consideration. The real-world science and engineering optimization applications are becoming more and more complex in nature and are generally non-linear multimodal. The conventional optimization algorithms fail to optimize such complex multimodal problems that are characterized by multiple local optimums. Hence there is a requirement of efficient and robust optimization strategies that can handle complex problems with multiple local optimums

.

In recent years, algorithms based on uncertainly or random process has become popular and is an alternative to conventional optimization techniques. These are called stochastic algorithms. Most of the stochastic algorithms are nature inspired, where the computing machines are being drawn from the aspects featuring the capabilities of evolution, growth and adaptation of living organisms. The well-known fields of stochastic algorithm research are Artificial Neural Networks

DOI : 10.5121/ijsc.2011.2301

(ANN) [6], Evolutionary Algorithms (EA) [7], Swarm Intelligence (SI) [8], Artificial Immune Systems (AIS) [9], Fractal Geometry (FG) [10], DNA computing [11], and Quantum Computing (QC) [12-13]. Though there is variety of stochastic algorithm, there is not a single algorithm for solving all kind of problems, no free lunch [14]. Some algorithms are suitable for one kind of problem and not for other. This paper provides an overview of the fundamentals of stochastic algorithms, emphasizing the motivation, implementation details, and their scope of applications. The paper is organized as follows: Section 2 briefly describes the optimization. Section 3 presents the stochastic algorithms. Section 4 briefly describes stochastic algorithms that are used for compressive analysis. Section 5 presents the benchmark functions and experimental setup adopted for performance comparison. Section 6 describes the simulation results. Section 7 concludes the paper.

2. OPTIMIZATION

The following formatting rules must be followed strictly. This (.doc) document may be used as a template for papers prepared using Microsoft Word. Papers not conforming to these requirements may not be published in the conference proceedings.

$$\begin{aligned}
&\text{Minimize } f(x), x = (x_1, x_2, \ldots, x_D) \in R^D \\
&\text{subject to: } \mathrm{g}_{\mathrm{i}}(x) < 0, i = 1,2,\cdots,q. \\
&\qquad h_i(x) = 0, i = q+1, q+2, \ldots, m \qquad (1) \\
&\qquad l_i \le x_i \le u_i, i = 1,2,\ldots,D
\end{aligned}$$

The functions f, g_i, and h_i are objective function, inequality constraint and equality constraints respectively. These are generally linear or nonlinear real-valued functions. The values l_i and u_i are the lower and upper bounds defining the search space (R).

3. STOCHASTIC ALGORITHMS

The real-world science and engineering optimization applications are becoming more and more complex in nature and are generally non-linear and multimodal. The classical optimization techniques viz. nonlinear programming techniques may fail to solve such problems because these problems usually contain multiple local optima. Therefore, using derivative free global search methods are in demand in order to achieve acceptable solutions. Thus global search methods should be invoked in order to deal with such problems. In recent years, algorithms based on uncertainly or random process (stochastic algorithms) has become popular and is an alternative to conventional optimization techniques. Most of the stochastic algorithms are nature-inspired, where the computing capabilities are being drawn from the biological process of evolution, growth and adaptation of living organisms.

The stochastic algorithms encompass three classes of methods [1-3]:

(1) Computing inspired by nature: The main idea of this branch is to develop computational tools (algorithms) by taking inspiration from nature for the solution of complex problems. The special features from complex living organisms are extracted that are required for problem solving.

(2) The simulation of nature by means of computing: This method is basically used to mimic various natural phenomena, thus increasing our understanding of nature and insights about computer models. It is a synthetic process for creating patterns, forms, behaviours, and organisms that (do not necessarily) resemble life-as-we-know-it.

(3) Computing with natural materials: It corresponds to the use of novel natural materials to perform computation, thus constituting a true novel computing paradigm that comes to substitute or supplement the current silicon-based computers. The problem solving capabilities of stochastic algorithms are illustrated with the help of a variant of Swarm Intelligence (SI) called Particle Swarm Optimization (PSO). PSO is a population based optimization algorithms introduced in 1995 by Kennedy and Eberhart[15-16]. PSO is almost similar to whole swarm reaches the goal. Since the particle's position in a swarm represents the potential solution, hence is evaluated based on the fitness function to be optimized. The value of fitness function extrapolates the quality of solution. Thus the social behaviour of the swarm can be extracted and a model can be built for solving the problems. When the stochastic algorithms be used: The stochastic algorithms are the excellent tools for global optimization. These algorithms are very easy to understand and simple to implement. These algorithms especially dominates the classical optimization methods when;

1) The problem to be solved is complex.
2) The problem involves a large number of variables.
3) The problem is highly dynamic and nonlinear.
4) The problem has multiple objectives.
5) The problem to be solved cannot be (suitably) modelled.
6) A single solution is not good enough or when diversity is important. As most standard techniques are deterministic and these methods are stochastic.

The main research fields of the stochastic algorithms are:

1) Artificial Neural Networks (ANN)
2) Artificial Immune Systems (AIS)
3) Evolutionary Computation (EC):
Genetic Algorithms (GA), Genetic Programming (GP), Evolution Strategies (ES), Evolutionary Programming (EP), Tabu Search (TS), Harmony Search (HS)
4) Swarm Intelligence (SI):
Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC)

3.1. Artificial Neural Networks

The research of MC Culloch and Pitts [18] is a milestone in the field of Artificial Neural Networks (ANN). This introduced the first mathematical model of a neuron that gave rise to the ANNs [6, 19]. The ANNs can be defined as information processing systems designed with inspiration taken from the nervous system, in most cases the human brain. Currently, most works on ANNs place particular emphasis on problem solving Artificial Neural Networks

3.2. Artificial Immune System

This field of research that emerged around the mid 1980s. Artificial Immune Systems (AIS) is a system inspired by theoretical and experimental immunology of the organism with the goal of solving problems [9,20]. They encompass any system or computational tool that extracts ideas from the biological immune system in order to solve problems.

3.3. Evolutionary Computation

Evolutionary Computing draws ideas from evolutionary biology in order to develop, search and optimization techniques for solving complex problems [7, 21-25]. These basically states that a population of individuals capable of reproducing, subjected to genetic variation followed by selection. This results in new population of individuals comparatively more fit to adapt to their environment. This special characteristics of organism which follows Darwinism helps in solving problems.

3.4. Swarm Intelligence

The term Swarm Intelligence (SI) was coined in the late 1980s to refer to cellular robotic systems in which a collection of simple agents in an environment interact according to local rules [26-28]. In different terms SI can be defined as, it is a property of systems of unintelligent agents of limited individual capabilities exhibiting collective intelligent behavior [28-29]. This collective behavior of simple agent's results in enormous decision making and helps in developing problem solving technique.

4. STOCHASTIC ALGORITHMS

4.1. Evolutionary Algorithms

Evolution Algorithms are based on evolutionary biology, the Darwinism. All the EA algorithms work on same common principle, though with some differences. The common idea is given a population of individuals, the environmental pressure causes natural selection (survival of the fittest) and this causes a rise in the fitness of the population [7, 21-25]. For a function to be optimized we can randomly create a set of candidate solutions, and evaluate the goodness of the candidate based on function fitness measure. Based on this fitness, some of the better candidates are chosen to seed the next generation by applying recombination and/or mutation to them. Recombination is an operator applied to two or more selected candidates (the so-called parents) and results one or more new candidates (the children). Mutation is applied to one candidate and results in one new candidate. Executing recombination and mutation leads to a set of new candidates (the offspring). These compete based on their fitness with the old ones for a place in the next generation. This process can be iterated until a candidate with sufficient quality (a solution) is found or a previously set computational limit is reached. The general scheme of EAs is as follows

```
BEGIN
    INITIALIZE population with random solutions;
    EVALUATE each candidate;
    REPEAT UNTIL (CONDITION is satisfied) DO
    1 SELECT parent;
    2 RECOMBINE pairs of parents;
    3 MUTATE the resulting offspring;
```

```
        4 EVALUATE new candidates;
        5 SELECT individuals for the next generation;
    END
```

4.1.1 Genetic Algorithm:

GA: Genetic Algorithms (GA) well known EA that are modeled after mechanisms of natural selection [23-24]. Each optimization parameter (x_n) is encoded by a gene using an appropriate representation, such as a real number or a string of bits. The corresponding genes for all parameters x_1,... x_n form a chromosome capable of describing an individual design solution. A set of chromosomes representing several individual design solutions comprises a population where the fittest are selected to reproduce. Mating is performed using crossover to combine genes from different parents to produce children. The children are inserted into the population and the procedure starts over again, thus creating an artificial Darwinian environment. The basic steps of GA algorithm are almost similar to EA explained above.

4.1.2 Evolutionary Strategy:

ES: Evolutionary Strategy (ES) is also an EA which is almost all similar to GA. ES differs from GA only in representation of a candidate solution. The representation of a candidate solution for GA is real number or a string of bits, but for ES it is real valued vectors. It uses the probability density function to self adapt its parameters [29]. The basic steps of ES are also similar to EA explained above.

4.1.3 Differential Evolution:

DE: Differential Evolution (DE)[30] is one of the successful EA. It shares similarities with other traditional EAs, but it does not use binary encoding as in simple GA [23-25] and probability density function to self adapt its parameters as in ES [29]. Instead DE extracts the differential information about distance and direction from the current population to guide further search. DE maintains a population of N points in every generation, where each point is a potential solution. In each generation a new population is generated based on the current population. To generate offspring's for the new population; the algorithm extracts distance and direction information from the current population and adds random deviation for diversity. If off springs are better then they replace current population. This process continues until stopping criteria is met.

4.2. Swarm Intelligence: SI

The term Swarm Intelligence (SI) was coined in the late 1980s to refer to cellular robotic systems in which a collection of simple agents in an environment interact according to local rules [26-28]. In different terms SI can be defined as, it is a property of systems of unintelligent agents of limited individual capabilities exhibiting collectively intelligent behavior [28].

4.2.1 Particle Swarm Optimization: PSO:

As stated above PSO was introduced in 1995 by Kennedy and Eberhart[15-16]. PSO simulates the social life, such as a swarm of birds or school of fish [17], and harnesses the searching capability of a swarm. During the search process, every particle remembers its best position found so far called personal best (pbest). And some how learns the best position found by any particle in the swarm called the global best (gbest). These two factors decide the flying trajectory of the particle from its current location. Since the particle's position in a swarm represents the potential solution, it is represented in d-dimensional search space as $X_1,\ldots,X_d$ and the velocity as $V_1,\ldots,V_d$. The PSO algorithm updates the velocity and position of each particle by the following equations respectively.

$$V_{new} = V_{old} + c_1 * rand * (pbest - X) + c_2 * rand * (gbest - X)$$
$$X_{new} = X_{old} + V_{new}$$

Where, c_1 and c_2 are the learning factors which determine the relative influence of cognitive and social component respectively. "rand" is the random numbers in the range [0,1].

4.2.2 Artificial Bee Colony:

ABC: Karaboga has introduced artificial bee colony (ABC) algorithm [31]. The colony of artificial bees consists of three groups of bees: employed bees, onlookers and scouts. The first half of the colony consists of the employed artificial bees and the second half includes the onlookers. For every food source, there is only one employed bee. The number of employed bees is equal to the number of food sources around the hive. The employed bee whose food source has been exhausted by the bees becomes a scout. The information exchange among bees is through dancing area called waggle dance. Since information about all the current rich sources is available to an onlooker on the dance floor, she probably could watch numerous dances and choose to employ herself at the most profitable source. There is a greater probability of onlookers choosing more profitable sources since more information is circulating about the more profitable sources. Employed foragers share their information with a probability, which is proportional to the profitability of the food source, and the sharing of this information through waggle dancing is longer in duration. Hence, the recruitment is proportional to profitability of a food source. The main steps of the algorithm can be summarized as: Send the scouts and employed bees onto the initial food sources, then determine their nectar amounts. Calculate the probability value of the sources. Stop the exploitation process of the sources abandoned by the bees and send the scouts into the search area randomly. Memorize the best food source found so far.

5. SIMULATION

5.1. PC Configuration

The simulation of Bioinspired Computational Algorithms is carried out on Pentium 4 2.0GHz, Windows XP Professional with 2GB of RAM. The coding is done in Matlab 7.2

5.2. Parameters settings

Experiments are carried out with the population size of 25, the number of iterations 1000. The results recorded are the average of 20 trials. The stopping criteria for all the algorithms are set to e^{-100}. The algorithms are tested for convergence on a set of standard benchmark functions with dimension 10. The dimensions of 10, 30 and 50 were used for testing quality of results. For PSO we have used social and cognitive factors $c_1 = c_2 = 2.0$. Velocity limit is set 20 percent of the search range. For GA we have used Elite Count and Crossover Fraction to be 2.0 and 0.8 respectively. For Classical DE the crossover probability and step size to be 0.5 and 0.3 is used. For ES $\beta = 0.04086$. For ABC we have used the Number of Employed and Onlooker Bees is 50% of population and the scout bees to be 1.

5.3. Benchmark functions

Stochastic Algorithms experience difficulty in optimizing complex multimodal optimization problems with multiple minima; therefore we focus on well-known standard benchmark problems [32] which have many local minima as shown in Table I. The benchmark functions are numbered as f1 to f6 with their names. The Table I give complete details of the benchmark function's definition and their optima. Figure 1 shows the pictorial details of the problems. The Griewank function f1 is a multimodal with multiple minima. It has a product term, introducing interdependency between the variables. The characteristic of Rastrigin function f2 is the existence of many suboptimal peaks whose values increase as the distance from the global optimum point increases. It is also a multimodal with multiple minima. The Rosenbrock function f3 is characterized by an extremely deep valley along the parabola that leads to the global minimum. Due to the non-linearity of the valley, many algorithms converge slowly because they change the direction of the search repeatedly. The function has a long gully with very steep walls and almost flat bottom. The Schwefels function f4 is also a multimodal function. The most important property of Schwefel function is that it traps all other algorithms in local optima. Sphere function f5 is simple, smooth, symmetrical and strongly convex and can be considered as unimodal or multimodal. Zakharov functions f6 is simple, strongly convex and can be considered as unimodal or multimodal.

6. RESULTS

The simulation experiments were conducted on a set of six complex multimodal benchmark problems of dimensions 10, 30 and 50. The performance comparisons are done with four different aspects a) mean results b) convergence rate c) stability and d) problem dimensional dependency.

6.1. Mean results:

The average results of 20 trials and 1000 iterations by each of the algorithm over six standard benchmark problems with dimension of 10, 30 and 50 are presented in Table II to Table IV.

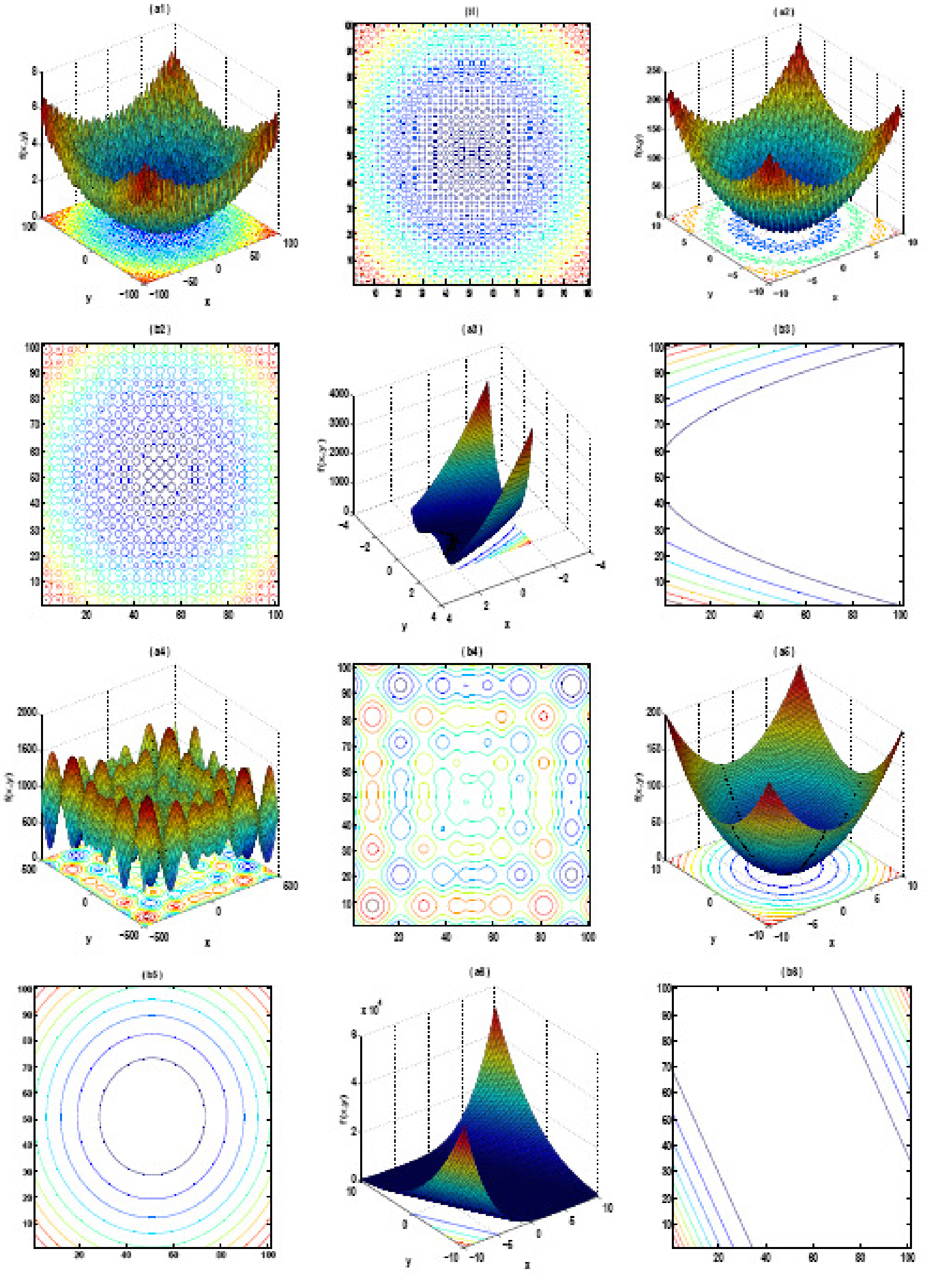

Fig. 1. (a1, a2, a3, a4, a5, a6) and (b1, b2, b3, b4, b5, b6) Surface and Contour plots of Griewank, Rastrigin, Rosenbrock, Schwefel,Sphere and Zakhar problems respectively.

The first and second columns in these tables show the functions and algorithms. These tables give the complete information about the results achieved by the stochastic algorithms on each of the selected problems. The best mean result and best standard deviations achieved by the algorithms are shown in bold.

6.1.1. Dimension 10:

From Table II it can be observed that the mean results of ES performs well on problems f1 and f6. GA performs well on problem f4. ABC does well on problem f3 and DE on f2 and f5 respectively.

TABLE I
BENCHMARK FUNCTIONS DEFINITION

Function Name	Function Definition	Search Range	Minimum Value
Griewank	$f_1(x)=\sum_{i=1}^{D}\frac{x_i^2}{4000}-\prod_{i=1}^{N}cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	[-100, 100]	$f_1(\vec{0})=0$
Rastrigin	$f_2(x)=\sum_{i=1}^{D}\left(x_i^2-10cos(2\pi x_i)+10\right)$	[-5.12 5.12]	$f_2(\vec{0})=0$
Rosenbrock	$f_3(x)=\sum_{i=1}^{D-1}\left[100\left(x_i^2-x_{i+1}\right)^2+(x_i-1)^2\right]$	[-100 100]	$f_3(\vec{1})=0$
Schwefel	$f_4(x)=\sum_{i=1}^{D}(\sum_{j=1}^{i}x_i)^2$	[-100 100]	$f_4(420.\vec{9}687)=0$
Sphere	$f_5(x)=\sum_{i=1}^{D}x_i^2$	[-10, 10]	$f_5(\vec{0})=0$
Zakharov	$f_6(x)=\sum_{i=1}^{n}x_i^2+\left(\sum_{i=1}^{n}0.5ix_i\right)^2+\left(\sum_{i=1}^{n}0.5ix_i\right)^4$	[-10, 10]	$f_6(\vec{0})=0$

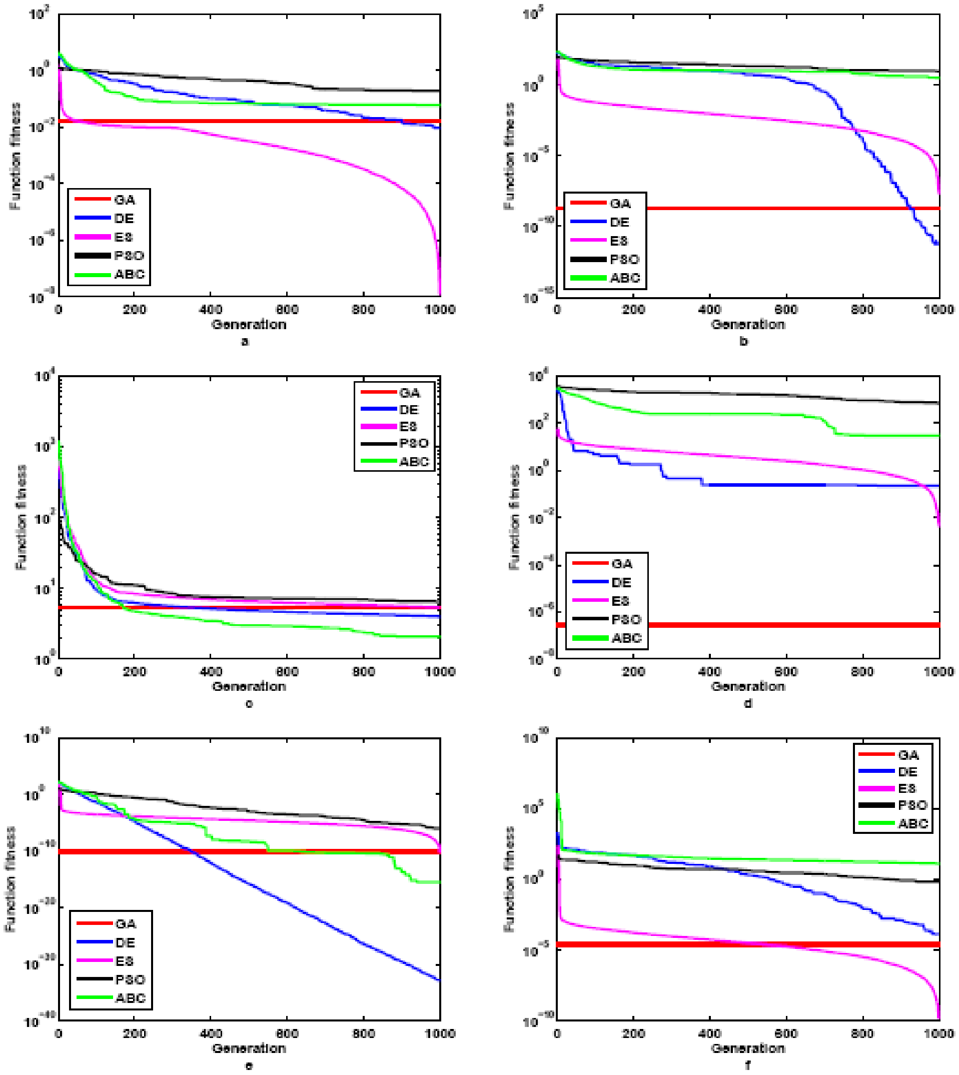

Fig. 2. Convergence graphs of (a) $Griewank$, (b) $Rastrigin$, (c) $Rosenbrock$, (d) $Schwefel$, (e) $Sphere$, (f) $Zakharov$

6.1.2. Dimension 30:

The results of 30 dimensions are shown in Table III. The mean results from Table III shows that ES attains good results on problems f1, f5 and f6. The GA outperforms on problems f2, f3 and f4.

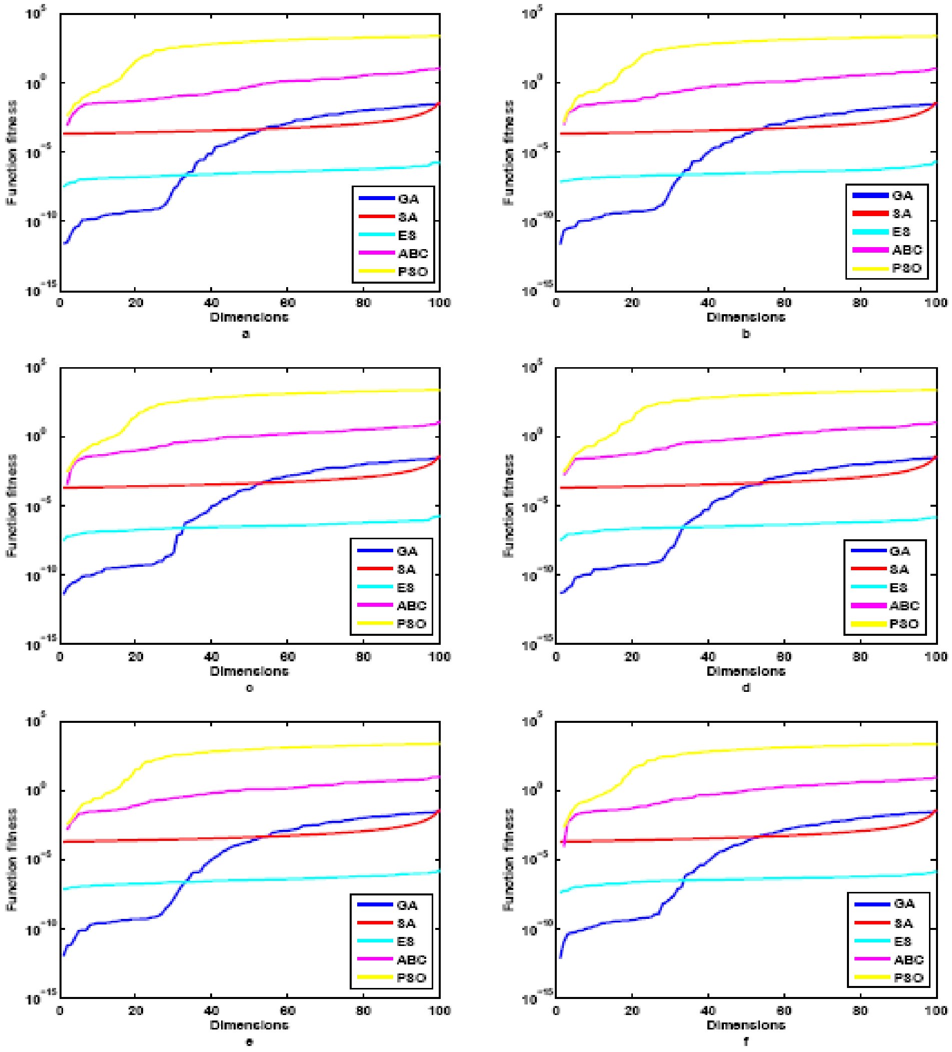

Fig. 3. Dimensional dependency graphs of (a) *Griewank*, (b) *Rastrigin*, (c) *Rosenbrock*, (d) *Schwefel*, (e) *Sphere*, (f) *Zakharov*

6.1.3. Dimension 50:

The Table IV records the results of 50 dimensions. From Table IV it is observed that ES surpasses all other algorithms on f1, f2, f5 and f6, leaving only f3 and f4 for ABC and GA respectively.

6.2. Convergence:

Convergence characteristics of the algorithms over iteration are shown in Fig 2. The results for the showing the convergence of different algorithms are also the mean of 20 trials over 1000 iterations. The Convergence characteristics are shown only for 10 dimensional problems. Through the experiments it is observed that the convergence characteristics of the algorithms for higher dimensions does not vary considerably and hence are not included for analysis. From the Fig 2 it is seen that ES converges faster on problems f1 and f6. The DE converges faster on f2 and f5. The faster convergence rate of ABC and GA are seen on problem f3 and f4 respectively.

TABLE II
RESULTS FOR 10 DIMENSION FUNCTIONS MEAN(STD)

fun	Alg	best	worst	mean	median	std
f_1	GA	5.64e-11	3.95e-2	1.73e-2	7.40e-3	2.05e-2
	DE	8.33e-14	3.23e-2	9.42e-3	7.40e-3	1.33e-2
	ES	1.51e-10	4.88e-8	1.23e-8	1.97e-9	2.08e-8
	PSO	2.46e-2	3.77e-1	1.97e-1	2.30e-1	1.32e-1
	ABC	6.23e-3	1.31e-1	6.01e-2	4.77e-2	4.71e-2
f_2	GA	7.18e-12	1.05e-8	2.23e-9	2.74e-10	4.60e-9
	DE	0.0	1.48e-11	3.10e-12	9.95e-14	6.56e-12
	ES	6.30e-10	6.86e-8	2.04e-8	8.28e-9	2.79e-8
	PSO	4.16	1.97e+1	9.69	7.36	5.99
	ABC	1.27	9.45	3.27	2.05	3.47
f_3	GA	4.50	5.71	5.30	5.55	4.92e-1
	DE	3.01	5.08	4.02	4.25	9.01e-1
	ES	4.06	6.86	5.43	5.73	1.22
	PSO	6.02	6.86	6.48	6.48	3.28e-1
	ABC	9.29e-1	2.99	2.09	2.30	9.20e-1
f_4	GA	1.16e-10	8.06e-7	2.80e-7	1.11e-7	3.46e-7
	DE	6.07e-2	3.64e-1	2.26e-1	3.07e-1	1.42e-1
	ES	3.86e-4	1.46e-2	4.33e-3	1.89e-3	5.86e-3
	PSO	3.02e+2	1.43e+3	7.05e+2	6.28e+2	4.26e+2
	ABC	2.03e-4	1.48e+2	2.99e+1	7.63e-4	6.61e+1
f_5	GA	9.11e-12	3.85e-10	9.42e-11	2.03e-11	1.63e-10
	DE	2.48e-35	7.14e-33	1.83e-33	4.56e-34	3.02e-33
	ES	9.68e-14	1.98e-10	5.26e-11	1.87e-11	8.25e-11
	PSO	5.38e-8	4.56e-6	1.05e-6	2.41e-7	1.97e-6
	ABC	2.18e-16	5.25e-16	3.29e-16	3.00e-16	1.16e-16
f_6	GA	3.52e-7	1.25e-4	2.68e-5	2.08e-6	5.51e-5
	DE	2.41e-5	5.10e-4	1.40e-4	4.67e-5	2.09e-4
	ES	2.45e-13	3.71e-10	1.48e-10	6.00e-11	1.78e-10
	PSO	1.02e-1	2.38	6.85e-1	2.38e-1	9.61e-1
	ABC	1.54	2.47e+1	1.38e+1	1.53e+1	1.08e+1

TABLE III
RESULTS FOR 30 DIMENSION FUNCTIONS MEAN(STD)

fun	Alg	best	worst	mean	median	std
f_1	GA	4.79e-14	3.95e-2	1.58e-2	9.33e-10	2.16e-2
	DE	5.28e-9	6.94e-7	1.49e-7	1.51e-8	3.05e-7
	ES	6.45e-10	7.38e-9	4.36e-9	5.60e-9	3.23e-9
	PSO	1.39	4.13	2.89	3.02	9.78e-1
	ABC	4.99e-5	1.29e-1	6.02e-2	3.50e-2	5.58e-2
f_2	GA	1.82e-11	1.84e-8	4.43e-9	9.50e-10	7.87e-9
	DE	1.36e+2	1.54e+2	1.46e+2	1.50e+2	8.05
	ES	1.08e-9	4.44e-8	1.81e-8	1.71e-8	1.80e-8
	PSO	2.79e+2	3.28e+2	3.06e+2	3.11e+2	1.93e+1
	ABC	4.15	5.03e+1	2.06e+1	1.38e+1	1.78e+1
f_3	GA	2.45e+1	2.65e+1	2.57e+1	2.63e+1	9.41e-1
	DE	2.50e+1	2.86e+1	2.63e+1	2.59e+1	1.39
	ES	3.38e+1	3.86e+1	3.54e+1	3.49e+1	1.88
	PSO	3.51e+2	5.28e+2	4.71e+2	5.00e+2	7.33e+1
	ABC	1.22e+1	4.38e+1	2.67e+1	2.13e+1	1.31e+1
f_4	GA	5.26e-9	7.92e-8	2.63e-8	1.55e-8	2.99e-8
	DE	1.18e-1	2.74	1.18	7.94e-1	1.05
	ES	2.65e-3	3.03e-2	1.08e-2	6.27e-3	1.12e-2
	PSO	9.58e+3	1.18e+4	1.08e+4	1.13e+4	9.38e+2
	ABC	3.26e+2	1.27e+3	8.22e+2	8.39e+2	3.37e+2
f_5	GA	1.02e-11	1.54e-7	3.25e-8	5.70e-10	6.80e-8
	DE	4.26e-10	3.07e-9	1.49e-9	1.31e-9	1.07e-9
	ES	4.14e-12	4.45e-10	2.44e-10	3.10e-10	1.87e-10
	PSO	3.70e+1	9.50e+1	6.90e+1	8.43e+1	2.87e+1
	ABC	5.96e-10	1.34e-6	4.90e-7	3.03e-7	5.78e-7
f_6	GA	2.73e-5	1.28e-3	6.42e-4	4.94e-4	5.13e-4
	DE	1.13e+3	2.03e+3	1.59e+3	1.79e+3	4.22e+2
	ES	1.09e-12	1.13e-9	3.06e-10	2.15e-11	4.84e-10
	PSO	9.21e+1	2.50e+2	1.71e+2	1.63e+2	5.69e+1
	ABC	2.71e+2	3.87e+2	3.37e+2	3.36e+2	4.67e+1

TABLE IV
RESULTS FOR 50 DIMENSION FUNCTIONS MEAN(STD)

fun	Alg	best	worst	mean	median	std.
f_1	GA	8.29e-7	7.37e-1	1.69e-1	9.87e-3	3.19e-1
	DE	2.96e-4	6.66e-2	1.87e-2	8.19e-3	2.77e-2
	ES	2.86e-12	9.16e-9	2.95e-9	1.78e-9	3.67e-9
	PSO	5.34	6.85	6.00	5.70	6.55e-1
	ABC	1.32e-2	2.25e-1	9.93e-2	6.22e-2	8.89e-2
f_2	GA	3.14e-1	1.25e+001	5.58	3.20	6.02
	DE	3.15e+2	3.65e+2	3.38e+2	3.32e+2	2.43e+1
	ES	2.83e-10	4.65e-8	1.59e-8	8.14e-9	1.83e-8
	PSO	5.02e+2	6.47e+2	5.79e+2	5.73e+2	5.71e+1
	ABC	2.29e+1	5.96e+1	3.68e+1	3.07e+1	1.62e+1
f_3	GA	3.51e+1	4.75e+1	4.40e+1	4.61e+1	5.14
	DE	4.27e+1	4.95e+1	4.49e+1	4.35e+1	2.73
	ES	5.77e+1	6.68e+1	6.06e+1	5.87e+1	3.69
	PSO	1.06e+3	1.45e+3	1.22e+3	1.21e+3	1.43e+2
	ABC	9.47e+1	1.60e+2	1.27e+2	1.29e+2	2.53e+1
f_4	GA	5.38e-10	3.88e-8	1.48e-8	1.11e-8	1.60e-8
	DE	5.98e-1	9.92	2.92	1.01	3.97
	ES	2.54e-3	1.46e-2	7.16e-3	6.43e-3	4.49e-3
	PSO	1.95e+4	2.14e+4	2.07e+4	2.09e+4	7.69e+2
	ABC	1.36e+3	2.54e+3	2.12e+3	2.31e+3	4.69e+2
f_5	GA	6.43e-2	8.90e-1	5.13e-1	7.30e-1	3.81e-1
	DE	7.74e-5	6.83e-4	3.18e-4	1.31e-4	2.87e-4
	ES	1.35e-13	2.11e-10	4.56e-11	6.94e-12	9.24e-11
	PSO	1.53e+2	1.92e+2	1.68e+2	1.64e+2	1.47e+1
	ABC	2.22e-4	2.83e-1	5.89e-2	4.20e-3	1.25e-1
f_6	GA	1.54e-2	3.73e-1	1.72e-1	8.10e-2	1.69e-1
	DE	2.84e+3	6.65e+3	4.39e+3	3.04e+3	1.99e+3
	ES	5.09e-11	2.04e-9	8.10e-10	2.07e-10	9.52e-10
	PSO	2.77e+2	4.47e+2	3.43e+2	2.96e+2	7.67e+1
	ABC	6.35e+2	9.07e+2	7.85e+2	8.04e+2	1.03e+2

6.3. Stability:

Since almost all stochastic algorithms are based on random process, they do not produce the same results every time and hence a stability test is required. The stability or robustness of the algorithms are represented by the standard deviation they achieve over 20 trials and 1000 iterations. A 0.0 value of standard deviation indicate very stable nature of the algorithm, any other value indicate the deviation. The best standard deviation is shown in bold.

6.3.1. Dimension 10:

From Table II, it can be observed that ES is well stable on problems f1 and f6. GA is stable on problems f4. The DE does well on problem f2 and f5. The PSO is stable on f3

6.3.2. Dimension 30:

The results of 30 dimensions are shown in Table III. The standard deviation of ES shows that it is stable on problems f1, f5 and f6. The GA outperforms on problems f2, f3 and f4.

6.3.3. Dimension 50:

The Table IV records the results 50 dimension. From Table IV it is observed that ES surpasses all other algorithms on f1, f2, f5 and f6, leaving only f3 and f4 to DE and GA respectively.

6.4. Problem dimensional dependency:

Since the complexity of problems generally increases with increase in dimension, hence are suitable for testing the convergence behavior of the stochastic algorithms. The complexity of the problems is proportional to the number of variables (dimensions) it has. The problems that are selected here are scalable and their dimensions can be set as per the requirement. In our paper the dimension of the problems are increased linearly from 1 to 100 and for every dimension the average result of all the algorithms are recorded over 20 runs and 1000 iterations. The Figure 3.a to Figure 3.f shows the convergence characteristics of the algorithms on selected problems. The quality of solution of almost all the algorithms degrades with the increase in dimension. The ES shows comparatively stable result on selected problems over a wide range of dimension.

7. CONCLUSIONS

This paper presents a comprehensive investigation and analysis of Stochastic Algorithms on complex multimodal optimization problems of higher dimensions. The derivative free nature of the stochastic algorithms is well tested on the said problems as they are multimodal. Among the stochastic algorithms the ES, DE and GA algorithms outperform the other mentioned algorithms in terms of obtaining quality solution, convergence rate and stability. The strength of the algorithms lies in finding the global solution for static environments. Though these algorithms theoretically seem to do well on dynamic environments but one needs to test them and develop efficient algorithms. The authors are working towards developing efficient and adaptive stochastic algorithms for complex, dynamic environments and the application of stochastic algorithms in varied domains.

REFERENCES

[1] de Castro LN. Natural computing. In: Khosrow-Pour M, editor. Encyclopedia of information science and technology, vol. IV. Idea Group Inc; 2005. p. 20804.

[2] de Castro LN. Fundamentals of natural computing: basic concepts, algorithms, and applications. CRC Press LLC; 2006.

[3] Mange D, Tomassini M. Bio-inspired computing machines: towards novel computational architecture. Presses Polytechniques et Universitaires Romandes; 1998.

[4] Paton R, editor. Computing with biological metaphors. Chapman and Hall; 1994.

[5] Flake GW. The Computational beauty of nature. MIT Press; 2000.

[6] Haykin S. Neural networks: a comprehensive foundation. 2nd ed. Prentice-Hall; 1999.

[7] Back T, Fogel DB, Michalewicz Z. Evolutionary computation 1 basic algorithms and operators. Bristol and Philadelphia: Institute of Physics Publishing (IOP); 2000.

[8] Engelbrecht. Fundamentals of computational swarm intelligence. John Wiley and Sons; 2006

[9] Dasgupta D, editor. Artificial immune systems and their applications. Springer; 1999.

[10] Iannaccone PM, Khokha M, editors. Fractal geometry in biological systems. CRC Press; 1996.

[11] Zingel T. Formal models of DNA computing: a survey. Proc of the Est Ac of Sci Phys Math 2000;49(2):909.

[12] Hirvensalo M. Quantum computing. Springer; 2000.

[13] Nielsen MA, Chuang IL. Quantum computation and quantum information. Cambridge University Press; 2000.

[14] Wolpert, D.H.; Macready, W.G. " No free lunch theorems for optimization," IEEE Transactions on Evolutionary Computation, vol.1, no. 1, 1997, pp. 67-82.

[15] J. Kennedy, R. C. Eberhart, " Particle swarm optimization," in Proc. Of IEEE Conference on Neural Networks, Piscataway, NJ, 1995, pp. 1942-1948.

[16] J. Kennedy, R. C. Eberhart," A new optimizer using particle swarm theory," in Proc. of Sixth Int. Symposium on Micro machine and Human Science, Nagoya, Japan, 1995, pp. 39-43.

[17] J. Robinson and Yahya Rahmath Samii, " Particle Swarm Optimization in Electromagnetic," IEEE Transactions on Antenna and Propagation, vol.52, no. 2, 2004, pp. 397-400.

[18] McCulloch W, Pitts WH. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 1943;5:11533.

[19] Fausett L. Fundamentals of neural networks: architectures, algorithms, and applications. Prentice-Hall; 1994.

[20] de Castro LN, Timmis JI. Artificial immune systems: a new computational intelligence approach. Springer; 2002.

[21] Back T, Fogel DB, Michalewicz Z. Evolutionary computation 2 advanced algorithms and operators. Bristol and Philadelphia: Institute of Physics Publishing (IOP); 2000.

[22] Fogel DB, editor. Evolutionary computation: the fossil record. The IEEE Press; 1998.

[23] Goldberg DE. Genetic algorithms in search, optimization and machine learning. Reading, MA: Addison-Wesley; 1989

[24] Michalewicz Z. Genetic algorithms + data structures = evolution programs. Springer; 1996

[25] Mitchell M. An introduction to genetic algorithms. MIT Press; 1996.

[26] Beni G. The concept of cellular robotic systems. In: Proc of the IEEE Int. Symp. on Intelligent Control. 1988. p. 5762.

[27] Beni G, Wang J. Swarm intelligence. In: Proc of the 7th annual meeting of the robotics society of Japan. 1989. p. 4258.

[28] White T, Pagurek B. Towards multi-swarm problem solving in networks. In: Proc of the 3rd Int Conf on multi-agent systems (ICMAS98). 1998. p. 33340.

[29] Beyer H-G. Theory of evolution strategies. Springer; 2001.

[30] R. Storn and K. Price, Differential Evolution A Simplified and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces, Techhnical Report TR-95-012, International Computer Science Institute, Berkely, California, 1995.

[31] D. Karaboga, An Idea Based On Honey Bee Swarm For Numerical Optimization, Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[32] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger and S. Tiwari, " Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization," Technical Report, Nanyang Technological University, Singapore, AND KanGAL Report 2005005, IIT Kanpur, India, May 2005