Memetic Algorithm: Hybridization of Hill Climbing with Selection Operator

Rakesh Kumar, Sanjay Tyagi, Manju Sharma

***Abstract*— Genetic Algorithms are the population based search and optimization technique that mimic the process of natural evolution. Premature Convergence and genetic drift are the inherent characteristics of genetic algorithms that make them incapable of finding global optimal solution. A memetic algorithm is an extension of genetic algorithm that incorporates the local search techniques within genetic operations so as to prevent the premature convergence and improve performance in case of NP-hard problems. This paper proposes a new memetic algorithm where hill climbing local search is applied to each individual selected after selection operation. The experiments have been conducted using four different benchmark functions and implementation is carried out using MATLAB. The function's result shows that the proposed memetic algorithm performs better than the genetic algorithm in terms of producing more optimal results and maintains balance between exploitation and exploration within the search space.**

***Index Terms*—benchmark functions, hybrid genetic algorithms, hill climbing, memetic algorithms.**

I. INTRODUCTION

Genetic algorithms are the search technique based on the evolutionary ideas of natural selection and genetics [1]. Genetic algorithms use the principles inspired by natural population genetics to evolve solutions to problems. They follow the principle of survival of fittest [2], for better adaptation of species to their environment. For more than four decades, they have been applied on wide range of optimization problems.

The performance of genetic algorithms depends on the balancing between the exploitation and exploration techniques. Exploitation means to use the already available knowledge to find out the better solution and Exploration is to investigate new and unknown area in search space. The power of genetic algorithms comes from their ability to combine both exploration and exploitation in an optimal way [3]. Genetic algorithms are inspired from biological genetics model and most of its terminology has been borrowed from genetics.

Each allele has a unique position on chromosome called locus. Genetic algorithm uses an iterative process to create a population. The algorithm stops, when the population converges towards the optimal solution. It consists of following steps:-

- INITIALIZATION: Randomly generate a population of N chromosomes.
- SELECTION: Individuals are selected to create mate pool for reproduction according to selection methods.
- REPRODUCTION: Crossover and mutation operators applied on the mate pool individuals.
- REPLACEMENT: Individuals from old population are replaced by new ones according to replacement strategies.

In practice, the population size is finite that influences the performance of genetic algorithm and leads to the problem of genetic drift that occurs mostly in case of multimodal search space. Incorporating a local search method within the genetic operators can introduce new genes than can overcome the problem of genetic drift and accelerate the search towards global optima [4]. A combination of genetic algorithm and a local search method is called as hybrid genetic algorithm or memetic algorithm. In hybrid genetic algorithms, knowledge and local search can be incorporated at any stage like initialization, selection, crossover and mutation. This paper incorporates hill climbing based local search after selection step, the new algorithm is proposed called as hybrid genetic and hill climbing algorithm (HGHCA).The proposed HGHCA is compared with Genetic algorithm(GA) on standard benchmark multimodal functions.

The paper is organized in the following sections. In section 2, literature review is given on different researches related to hybrid genetic algorithms. In section 3, memetic algorithm approach and hill climbing search along with their pseudo codes are discussed. In section 4, benchmark test functions considered for implementation are described. Implementation details and computational results are specified in section 5 and conclusion and future work are given in section 6.

II. RELATED WORK

Holland [3] and David Goldberg [1] by using k armed bandit analogy showed that both exploration and exploitation are used by genetic algorithm at the same time. Due to certain parameters, it has been observed that, stochastic errors occur in genetic algorithm that leads to genetic drift [5,6]. Rakesh Kumar et al. proposed a novel crossover operator that uses the principle of Tabu search. They compared the proposed crossover with PMX and found that the proposed crossover yielded better results than PMX [7].

H.A. Sanusi et al. investigated the performance of genetic algorithm and memetic algorithm for constrained optimization knapsack problem. The analysis results showed that memetic algorithm converges faster than genetic algorithm and produces more optimal result [8]. A comparative analysis of memetic algorithm based on hill climbing search and genetic algorithm has been performed for

Manuscript received on May, 2013.

Rakesh Kumar, DCSA, Kurukshetra University, Kurukshetra, Haryana, India.

Sanjay Tyagi, DCSA, Kurukshetra University, Kurukshetra, Haryana, India.

Manju Sharma DCSA, Kurukshetra University, Kurukshetra, Haryana, India.

the cryptanalysis on simplified data encryption standard problem by Poonam Garg [9].She concluded that memetic algorithm is superior for finding number of keys than genetic algorithms.

Antariksha [10], proposed a hybrid genetic algorithm based on GA and Artificial Immune network Algorithm (GAIN) for finding optimal collision free path in case of mobile robot moving in static environment filled with obstacles. She concluded that GAIN is better for solving such kind of problems. E .Burke et al. proposed a memetic algorithm that based on Tabu search technique to solve the maintenance scheduling problem. The proposed MA performs better and can be usefully applied to real problems [11]. Malin et al [12] proposed a memetic algorithm for feature selection in volumetric data containing spatially distributed clusters of informative features in neuroscience application. They concluded that the proposed MA identified a majority of relevant features as compared to genetic algorithm.

III. METHODOLOGY

A. Memetic Algorithm Approach

Incorporating problem specific information in a genetic algorithm at any level of genetic operation form a hybrid genetic algorithm [13].The technique of hybridization of knowledge and global genetic algorithm is memetic algorithm (MA). MA is motivated by Dawkins notation of a meme. A meme is a unit of information that reproduces itself as people exchange ideas [14]. MA binds the functionality of GA with several heuristic's search techniques like hill climbing, simulated annealing, Tabu search etc. A number of issues should be carefully addressed when an effective hybrid genetic algorithm is constructed .Two popular ways of hybridization depends on the concepts of "Baldwin effect" [15] and "Lamarckism" [16]. According to Baldwinian search strategy, the local optimization can interacts and allow the local search to change the fitness of individual but genotype itself remain unchanged. The disadvantage of Baldwinism is that it is slow. According to Lamarckism, the characteristics acquired by individual during its lifetime may become heritable traits. According to this approach both the fitness and genotype of individuals are changed during local optimization phase. Most of the MA is based on Lamarckism approach of hybridization. The proposed memetic algorithm incorporates hill climbing local search after selection process in order to increase exploitation. In the proposed approach, members selected using roulette wheel selection has been used as initial point to carry out hill climbing search. In this approach, each individual is improved using hill climbing before passing to reproduction phase.

Algorithm 1: Proposed Hybrid Genetic Algorithm

```
  Procedure MA(fitfxn, psize, Pc, Pm)
  // fitfxn – fitness function to evaluate chromosome
  // psize – size of population in each generation
  // Pc – crossover probability
  // Pm – mutation probability
  // mxgen –maximum number of generations
  encode solution space
  Initialize population
  gen=1
while (gen <= mxgen)
    evaluate (min(fitfxn))
    for i = 1 to psize
        mate1, mate2=select(population)
   // apply local search to each selected individual
        optmate1=hill climbing (mate1)
        optmate2=hill climbing (mate2)
        If ( rnd(0,1)<=Pc)
          child = crossover(optmate1, optmate2)
        end If
        If ( rnd(0,1)<=Pm)
          mchild = mutation(child)
        End If
     End for
     Add offspring to new generation
     gen=gen+1
End while
return best chromosomes
```

B. Hill climbing local search

Hill climbing is an optimization algorithm for single objective function. In hill climbing algorithm a loop is performed in which the currently known best individual produce one offspring. If the fitness of new individual is better than parent it replaces it, else stop the loop.

Algorithm 2: Hill Climbing Algorithm

```
Procedure Hill climbing (parent)
//parent – currently known best solution
While (termination criteria is not specified) do
   New_solution <– neighbors (parent)
   If (New_Solution is better than parent)
      Parent = New_solution
   End If
 End While
return best solution
```

IV. TEST FUNCTIONS

Different researchers have used different function group to evaluate the genetic algorithm performance. In this paper, the authors examine four different benchmark multimodal functions in order to study the performance of purposed memetic algorithm.

Table 1 Benchmark Test Functions

Function	Name
F1	Rastrigin's Function
F2	Schwefel's Function
F3	Ackley's Function
F4	Griewangk's Function

Rastrigin's function (F1) is highly multimodal and has a complexity of O(n.log(n)), where n is the number of function parameters. It has several local minima [17].
Function definition:

$$f1(x) = 10*n + \sum_{i=1}^{n} [x_i^2 - 10*\cos(2.\Pi x_i)]$$

$-5.12 <= x_i <= 5.12$

global minimum: $fn(x)=0, x_i=0$

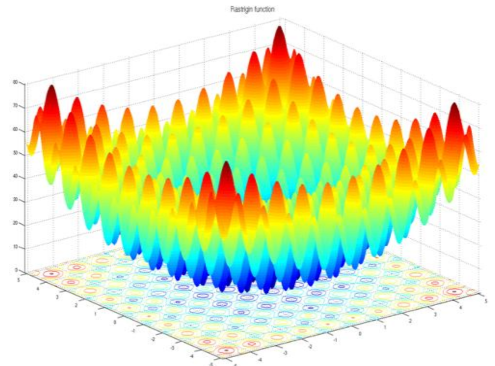


Fig. 1 Function Graph for F1 for n=2

In *Schwefel's function* (F2), the global minimum is geometrically distant over parameter spaces, from next best local optima [18].Therefore search algorithms are prone to converge in wrong direction.

Function definition:

$$f2(x) = \sum_{i=1}^{n} [-x .\sin(\sqrt{|x_i|})]$$

$-500<=x_i<=500$

global minimum: $x_i=420.9678$ $fn(x)=-418.9829n$

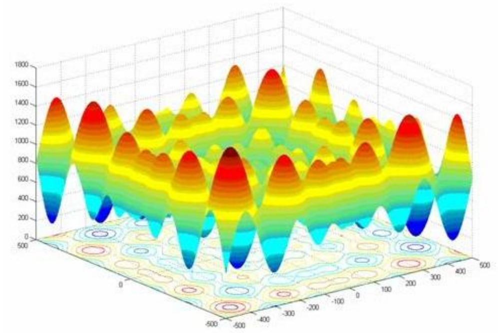


Fig. 2 Function Graph for F2 for n=2

Ackley's Function (F3) is continuous multimodal function obtained by modulating an exponential function with a cosine wave of moderate amplitude [17]

Function definition:

$$f3(x) = -a.\ e(-b.\sqrt{1/n}\sum_{i=1}^{n} x_i^2) - e(1/n \sum_{i=1}^{n} \cos(c.x_i)) + a + e$$

$-32.768<=x_i<=32.768$ a=20, b=0.2, c=2*p

global minimum: $fn(x)=0,\ x_i=0$

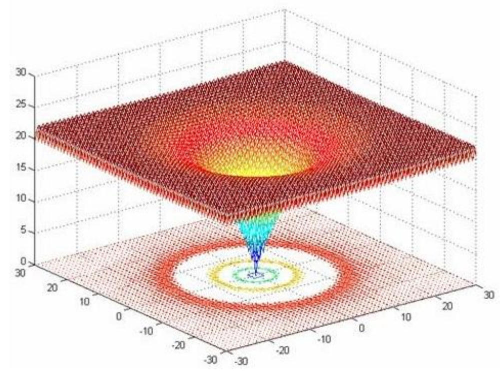


Fig. 3 Function Graph for F3 for n=2

Griewangk's Function (F4) is similar to Rastrigin and has many wide spread regularly distributed local minima[18].

Function definition:

$$f4(x) = 1/4000 \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(x_i/\sqrt{i}) + 1$$

$-600<=x_i<=600$

global minimum: $fn(x)=0, x_i=0$

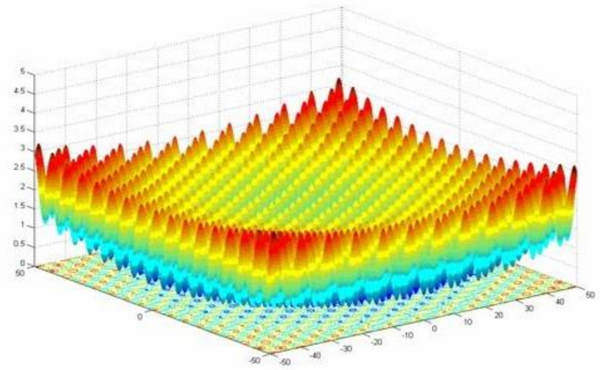


Fig. 4 Function Graph for F1 for n=2

V. IMPLEMENTATION & OBSERVATIONS

In this paper, MATLAB code has been developed to find the performance of proposed memetic algorithm and genetic algorithm. The code considers the group of four benchmark multimodal functions and uses the same initial population, same crossover and mutation probability in both selection cases to compare the performance of genetic algorithm (GA) with the proposed memetic algorithm (MA). Min and Average value of functions is computed for 100 & 50 generations and plotted to compare the result of two approach.

Table 2 : Minimum and Average value of fn(x) in F1

N			5	10	15	20
Gen=50	**SA**	**Min**	15.570	5.513	3.056	0.501
		Avg	28.46	14.12	24.41	24.71
	MA	**Min**	1.038	0.032	0.228	0.181
		Avg	14.24	10.64	17.33	15.50
Gen=100	**SA**	**Min**	21.64	2.192	2.495	3.235
		Avg	17.94	16.70	22.30	14.64
	MA	**Min**	0.164	0.296	0.099	0.224
		Avg	10.98	2.76	15.79	10.15

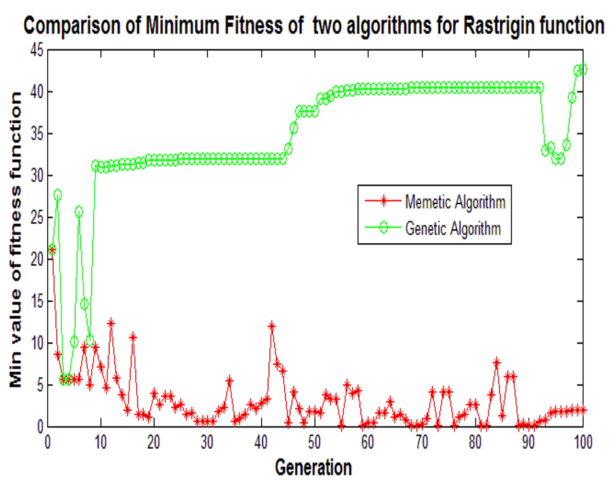


Fig. 5 Comparison of Minimum fitness value of fn(x) in F1

This section contains the result of code runs. Comparison of two algorithms is based on their respective function values.

Parameters used for implementation are-

- Population size (N): 5, 10, 15, 20
- Number of generations (Gen): 50, 100
- Encoding: Value encoding
- Selection: Roulette wheel selection
- Crossover operator: Arithmetic crossover operator
- Mutation: Creep Mutation
- Crossover probability (p_c=0.7)
- Mutation probability (p_m=0.01)

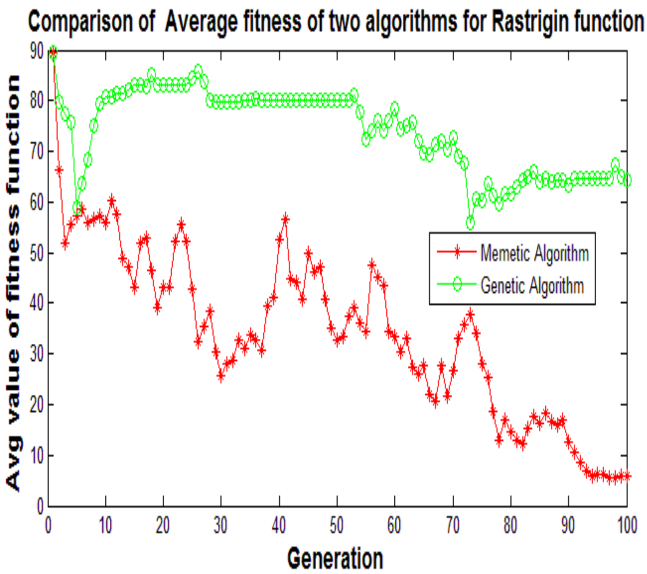


Fig. 6 Comparison of Average fitness value of fn(x) in F1

Table 3 : Minimum and Average value of fn(x) in F2

N			5	10	15	20
Gen=50	SA	Min	-396.6	-299	-678	-597.5
		Avg	-299.2	-359.3	-9.6	-180.7
	MA	Min	-766.2	-821.1	-718.8	-837.2
		Avg	-388	-575.3	-282.1	-824.8
Gen=100	SA	Min	-454.7	-365.5	-468.6	-589.7
		Avg	-235.5	-257.3	-306.4	-284
	MA	Min	-677.1	-836.1	-781.4	-835.7
		Avg	-371.2	-824.6	-459.9	-829.9

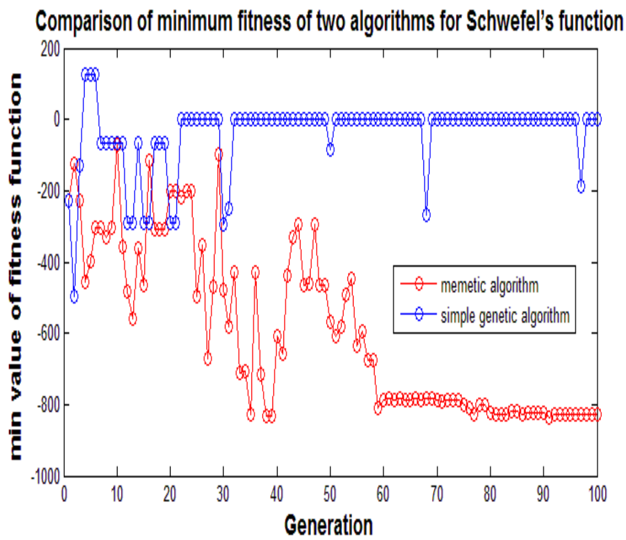


Fig. 7 Comparison of Minimum fitness value of fn(x) in F2

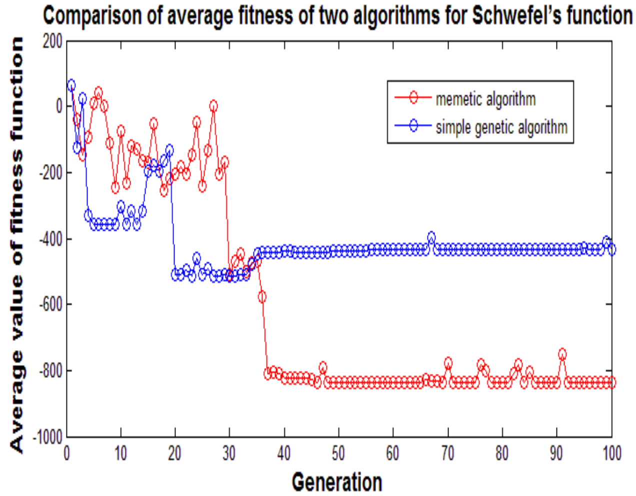


Fig. 8 Comparison of Average fitness value of fn(x) in F2

Table 4 : Minimum and Average value of fn(x) in F3

N			5	10	15	20
Gen=50	SA	Min	5.710	1.180	2.423	1.331
		Avg	5.803	5.985	7.226	5.986
	MA	Min	2.168	1.083	1.183	1.165
		Avg	4.269	2.991	6.194	5.168
Gen=100	SA	Min	3.659	5.633	1.848	1.742
		Avg	13.38	17.23	9.322	4.609
	MA	Min	1.157	1.369	1.128	1.139
		Avg	7.250	3.112	7.617	2.233

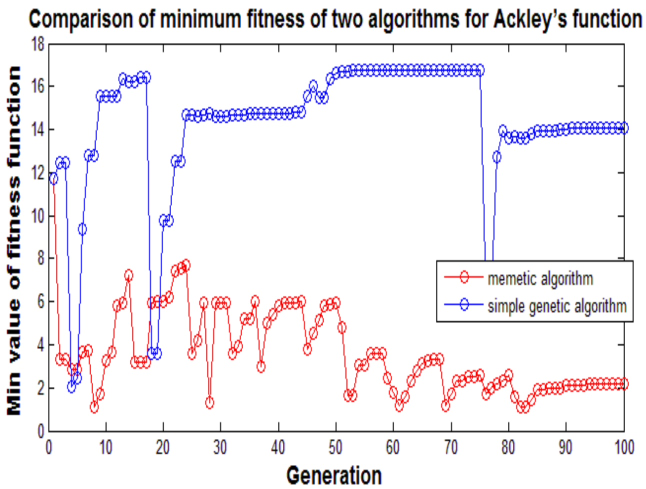


Fig. 9 Comparison of Minimum fitness value of fn(x) in F3

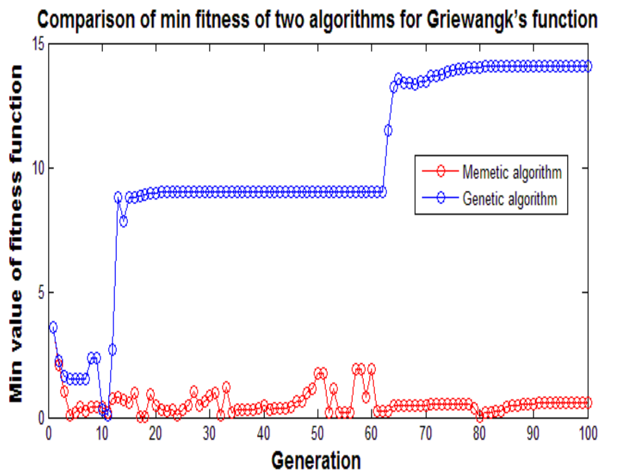


Fig. 11 Comparison of Minimum fitness value of fn(x) in F4

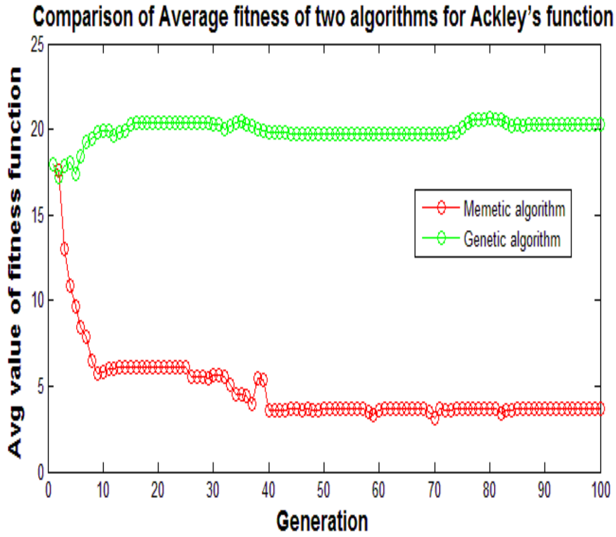


Fig. 10 Comparison of Average fitness value of fn(x) in F3

Table 5 : Minimum and Average value of fn(x) in F4

N			5	10	15	20
Gen=50	SA	Min	0.143	1.603	0.347	0.279
		Avg	1.410	1.502	1.732	1.177
	MA	Min	0.071	0.019	0.021	0.095
		Avg	1.341	0.743	0.623	1.170
Gen=100	SA	Min	1.519	0.078	0.045	0.447
		Avg	0.630	1.521	4.039	1.008
	MA	Min	0.079	0.024	0.014	0.052
		Avg	0.499	0.691	1.592	0.642

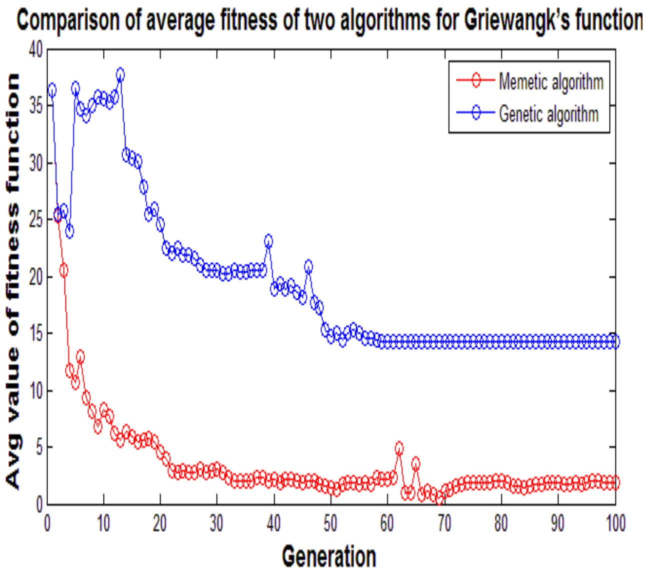


Fig. 12 Comparison of Average fitness value of fn(x) in F4

Results for Minimum as well as Average fitness value using different population size or generation in four benchmark multimodal functions are given in Table 2, Table 3, Table4 and Table 5. Figure 5, Figure 6, Figure 7, Figure 8, Figure 9, Figure 10, Figure 11 and Figure 12 shows the performance curve of two algorithms for 100 generations.

It has been observed that the proposed memetic algorithm has outperformed genetic algorithm in terms of convergence and optimal solution. The proposed MA maintains more diversity in population and prevent algorithm to stick in local optima and genetic drift problem. By applying the local search operation after roulette wheel selection, a good balance between exploitation and exploration is maintained. The local search operator generates more meaningful building blocks that help the genetic algorithms in making small moves within a defined search area. It has been found that with increasing problem size, problem did not converge prematurely and the same trend of results is observed in each problem size.

VI. CONCLUSION

The paper compares two algorithms namely genetic algorithm and proposed memetic algorithm on the standard multimodal benchmark test functions. It was found that the proposed memetic algorithm provides better results than the genetic algorithm. The proposed algorithm uses the concept of hill climbing local search after selection operation so as to allow the genetic algorithm to improve the exploiting ability of search without limiting its exploring ability. The proposed algorithm improves the performance in terms of convergence and optimal solution as well as maintains diversity in the population & solves the problem of premature convergence and genetic drift. The Proposed algorithm can prove to be better for different NP Hard problems also. This algorithm can be tested and implemented in different combination of crossover and initialization in future to substantiate its performance. Hybridization of selection and knowledge of incorporates hill climbing, has increased the existing genetic algorithm technique and amplified its search performance.

REFERENCES

[1] D. E. Goldberg, *Genetic algorithm in search and optimization and machine learning*, Addison Wesley Longman, Inc., ISBN 0-201-15767-5, 1989.

[2] D. Fogel, *Evolutionary computation*, IEEE Press, 1995.

[3] J. Holland, *Adaptation in natural and artificial systems*, University of Michigam Press, Ann Arbor, 1975.

[4] W. E. Hart, *Adaptive global optimization with local search*, Doctoral diss., San Diego, University of California, 1994.

[5] D. E. Goldberg and P. Segrest, "Finite Markov chain analysis of genetic algorithms", Proceedings of 2nd International Conf. on Genetic Algorithms, Lawrence Erlbaum Associates, 1987, pp 1-8.

[6] L .Booker, *Improving search in genetic algorithm, genetic algorithm and simulated annealing*, Pitman, vol 5, 1987, pp 61-73.

[7] Rakesh kumar and Jyotishree, "Novel knowledge based tabu crossover in genetic algorithms", International Journal of Advanced research in Computer science and software Engineering, vol 2, No. 8, Aug 2012, pp 78-82.

[8] H. A. Sansi, A. Zubair and R. O. Oladele, "Comparative assessment of genetic and memetic algorithms", Journal of Emerging Trends in Computing and Information Science, vol 2, No. 10, Oct 2011, pp 498-508.

[9] Poonam Garg, " A comparison between memetic algorithm and genetic algorithm for the cryptanalysis of simplified data encryption standard algorithm", International Journal of Network Security and its Applications,vol 1, No. 1, April 2009, pp 34-42.

[10] Antariksha Bhaduri, "A mobile robot path planning using genetic artificial immune network algorithm", Proceedings of World Congress on Nature and biologically Inspired Computing, NaBIC, IEEE, 2009, pp 1536-1539.

[11] E. K. Burke and A. J. Smith, "A memetic algorithm for the maintenance scheduling problem", Proceedings of International Conf on Neural Information Processing and Intelligent Information, Springer 2010, pp 469-473.

[12] Malin Bjornsdotter and Johan Wessberg, "A memetic algorithm for selection of 3D clustered featured with applications in neuroscience", Proceedings of International Conference on Pattern Recognition, IEEE, 2010, pp 1076-1079.

[13] P. Mascato and P. C. Cotta, "A gentle introduction to memetic algorithms", *Handbook of Metaheuristics*, 2003, pp 105-144.

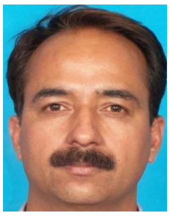
[14] R. Dawkins, *The selfish gene* , Oxford University Press, Oxford, 1976.

[15] K. Ku, M. Mak, "Empherical analysis of the factors that affect the Baldwin effect: Parallel problem solving from nature", Proceedings of 5th International Conference on lecture notes in computer science, Berlin, Springer, Heidelberg, 1998, pp 481-90.

[16] G. M. Morris, D. S. Goodsell, R. S. Halleday, W. E. Hartand and R. K. Belew, "Automated docking using a Lamarckian genetic algorithm and an empherical bending free energy function", Journal of Computational Chemistry, vol 19, 1998, pp 1639-62.

[17] J. G. Digalakis and K. G. Margaritis, "An experimental study of Benchmarking Functions for genetic algorithms", International Journal of Computer Mathematics, vol 79, No. 4, 2002,pp 403-416.

[18] Marcin Molgo and Czeslaw Smatnicki, *Test functions for optimization needs*, kwietnia, vol 3, 2005.



Dr. Rakesh Kumar obtained his B.Sc. Degree, Master's degree – Gold Medalist (Master of Computer Applications) and PhD (Computer Science & Applications) from Kurukshetra University, Kurukshetra. Currently, he is Professor in the Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India. His research interests are in Genetic Algorithm, Software Testing, Artificial Intelligence and Networking. He is a senior member of International Association of Computer Science and Information Technology (IACSIT).

Dr. Sanjay Tyagi obtained his Master's degree (Master of Computer Applications) and PhD (Computer Science & Applications) from Kurukshetra University, Kurukshetra. Currently, he is Assistant Professor in the Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India. His research interests are in Genetic Algorithm and Software Testing. He has presented 26 papers in National & International Conferences.

Ms Manju Sharma obtained her B. Tech (Computer Science and Engg.) degree from Kurukshetra University. Currently, she is pursuing M.Tech (Computer Science & Applications) from the Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India. She has presented papers in 3 National Conferences. Her research interests are Genetic Algorithms, Soft Computing methods and Software Engineering.