JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

Hybridizing Genetic Algorithm and Beam Search for Solving Optimization Problems

Ankita Chhikara[1], Rakesh Kumar[2]
Research Scholar[1], Professor[2]
Department of Computer Science & Applications,
Kurukshetra University, Kurukshetra (India)
ankita.30@kuk.ac.in[1], rakeshkumar@kuk.ac.in[2]

Received 2022 April 02**; Revised** 2022 May 20**; Accepted** 2022 June 18

Abstract: Genetic Algorithm (GA) is the search-based approach that mimics the idea of natural selection and the genetics of living beings. Due to the finite population size, GA suffers from the problem of genetic drift and premature convergence. To solve these problems, hybridization between the Genetic Algorithm and local search, i.e. Memetic Algorithm (MA) is used. MA minimizes the search time by reducing the computation and preventing premature convergence. In this paper, a proposal of new memetic algorithm has been given in which Beam search is applied in the selection process of a Genetic Algorithm to enhance the performance of a simple Genetic Algorithm. Experiments have been conducted using four benchmark functions, and the whole implementation is carried out using Python. Results demonstrate that the proposed hybrid algorithm provides better output than a simple Genetic Algorithm and maintains a state of equilibrium between exploration and exploitation within the search space.
Keywords: Beam Search, Benchmark function, Genetic Algorithm, Memetic Algorithm, Optimization

1. Introduction

The process of optimization is much more essential in this era and plays a very significant role in today's life. Generally, optimization problems have an objective function to be optimized based on given constraints which depend on the problem's availability [24]. There have been many heuristics and meta-heuristics algorithms available to solve an optimization problem. In the present technical era, one meta-heuristic approach fails to provide adequate results. So, it becomes important to combine one or more heuristic algorithmic approaches to get better results. Hybridization may be treated as a process that merges two or more algorithms to improve their performance [13]. Nowadays, the hybridization of evolutionary algorithms has become widespread due to their better and more significant capabilities for handling real-world problems like travelling salesman problem, knap-sack problem, bin packing, etc. Practically, the size of the population is limited, which may increase the genetic drift problem and influence the overall performance of the genetic algorithm. To speed up the search towards global optima and to reduce the problem of genetic drift, the local search approach is used within the operators of the genetic algorithms.

Merging a local search technique with a genetic algorithm is known as a hybrid genetic algorithm or memetic algorithm (MA). Different types of heuristic search techniques like simulated annealing, hill climbing, beam search, tabu search, etc. can be merged together with a genetic algorithm to form a hybrid genetic algorithm. In MA, the concept of local search and knowledge can be included in different phases of genetic algorithms such as initialization, selection, crossover, and mutation.

This paper integrates beam search after the selection step, where genes of the population are improved using beam search before passing to the crossover phase. Later, the proposed hybrid approach is compared with a simple Genetic algorithm (GA) on standard benchmark functions i.e. sphere function, rosenbrock function, rastrigin function, and schwefel function.

Section-wise structure of the paper is as follows: In section II, the Genetic Algorithm is explained and discussed in detail. In section III, the preliminary study gives a broader view of different researchers in the field of hybridization and memetic approach. Section IV describes the methodology including the memetic algorithm approach and beam search along with the algorithm of the proposed approach. In section V, different benchmark functions are explained in the detail. The experimental results and their analysis are explained in section VI. In the last section, the paper is briefly concluded.

2. Genetic Algorithm

GA is a search heuristic approach that depends upon evolutionary consideration of genetics and natural selection [23]. GA is a meta-heuristic search algorithm that gets inspiration from the biological evolution of living beings. The idea of "survival of the fittest" is imitated here. John Holland presented and introduced GA in the 1970s [1]. GA is a search technique that generates tremendously better solutions for hard optimization problems. The basic terms of GA have been inherited from genetics which resembles the hereditary process in biology. GA is a meta-heuristic method that can efficiently solve complex optimization problems. It outlines the search space as a population of chromosomes that explore the better individuals by generating the off-springs for the next generations iteratively [7]. It is a well-organized method that solves different types of problems to find the global optimal solution.

2.1 Phases of Genetic Algorithm:

Initialization: In this phase, strings are initialized randomly and then the initial population is generated using those strings. The population consists of chromosomes where each chromosome includes various genes to specify their attributes.

Fitness Function: The fitness function determines how close the given solutions are to the most favorable solution to the specific problem. It evaluates the generated solution by taking the input and producing the output solution. Basically, it explains the optimality of a solution. Also, it performs an assessment of the produced solution by providing fitness scores to each individual.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

Selection: It is a process of choosing parents who further combine together to produce new off-springs for the next generation. The parent's selection has a very significant task in the convergence of GA as the best parents will lead to a fitter and better output [8]. On the basis of fitness scores given to each individual, the best individuals are selected as parents. Further, the chosen parents will combine together to generate children for the next level. The objective behind the selection operator is to get rid of the worst solution and consider the best solution keeping in mind that the population size should be constant. It is motivated by the basic idea of "Survival of the fittest".

Crossover: It is a genetic algorithm phase that merges the attributes of two parents to form a new offspring. It is also known as recombination, which is very crucial for the reproduction process of GA. It is motivated by inter-course among individuals in living beings. In this process, randomly some sites are chosen to exchange the genes of parents to generate new children. Then, mating between the individuals is done to produce next generation individuals which may be better than the individuals of prior generation.

Algorithm 1: Genetic Algorithm

Procedure GA(n, ff, ngen)
// n – population size
// ff – fitness function
// ngen –maximum number of generations
Generate initial population P gen=1
while gen <= *ngen* **do**
Calculate fitness of all individuals
for *i= 1 to n* **do**
Select best parents based on fitness values
Apply Crossover on selected parents
Apply mutation on offspring
Replace old population with a newly generated population
end
gen = gen+1
end
Return best individual

Mutation: It is an operator which performs local modifications to new generations by altering some random genes in the population. Basically, it is a method to include new genetic codes into the population. Various ways can be there to mutate the individuals like inserting new genes in children or turn over the existing live genes. The main aim behind the mutation is to preserve the diversity of the population and to prevent the chromosomes from premature convergence.

Replacement: It is the replacement of the old population by the new population according to some replacement strategies. Various replacement strategies are there, the most common among them is Generational Replacement, in which the complete population of the previous generation is replaced by a new population.

A Genetic algorithm is an approach that follows the basic principle of genetics and natural selection [1]. This algorithm helps to produce the best possible solution if adequate time is provided to solve a particular problem. Various complex optimization problems can be solved using a Genetic Algorithm such as Image Processing, Network Routing Protocol, Travelling Salesman Problem, Gaming, Vehicle Routing Problem, Path Planning, etc.

3. Related Work

GA is an adaptive and perfect heuristic approach that belongs to the evolution of living beings. The GA and its basic workflow are discussed in detail. Process of GA includes initialization of population, fitness evaluation, selection of parents, crossover to obtain offspring, mutation to maintain diversity, and replacement of the old population by a new population. Various operators of GA are discussed with their types. The discussion has been done on the issues and challenges faced by GA. Various applications of GA in the area of Data mining, image processing, object-oriented approach, and some real-world problems are discussed[1][9][17][26][20].

Beam Search is a heuristic search method that expands the most optimistic node in a limited set. Basically, it is a variation of the branch and bound algorithm. In these papers, beam search-based scheduling algorithms are developed to solve job shop problems. A variation of best-first Beam Search is proposed which consumes less memory as compared to simple Beam search and at last, its applications in various scoring functions are discussed.[12][19][18]

An introduction to the memetic algorithm is given which reflects the futuristic research in the development and application of memetic algorithms to solve specific problems and relevant topics. The memetic algorithm has proved to be a successful approach to finding the imprecise solution to complex optimization problems. It depends upon the population of agents which combine various heuristic and local search techniques to form a hybrid approach. [3][29][16]

Genetic Algorithm (GA) and Ant Colony Optimization (ACO) are merged together to create a new hybrid approach that solves the Travelling Salesman Problem. The evolution process of a Genetic algorithm is usually combined with the instincts of an ant colony to calculate the shortest route to explore food. The building blocks of GA are found by the pheromones value of ACO. Sometimes, the factors affecting ACO are improved by GA in correspondence to the final outcome of ACO. After performing some experiments, results revealed that ACO produces considerably better results than GA to solve TSP. [14][15][27][30]

A New Memetic Algorithm is proposed in which a hill-climbing algorithm is applied to different phases of the Genetic Algorithm. Sometimes, it has been applied after the selection operator, after crossover, or after the mutation operator. Also, it may be applied in the replacement phase of the Genetic Algorithm. Different benchmark functions have been taken under consideration to perform the experiments and the whole implementation is done in MATLAB. Results showed that the proposed hybrid approach performs considerably far superior to the Genetic Algorithm in generating favorable output. A state of balance is maintained between exploration and exploitation within the search space. [6][10][22][23]

4. Methodology

4.1 Hybridization

Two or more components approximately perform the same function are merged together to form a hybrid. Hybridization may be treated as a process that merges two or more algorithms to improve their performance [13]. Two or more algorithms are combined together that can execute concurrently to generate a fruitful synergy from their combination [28]. In today's era, where optimization has become a necessary task, various hard optimization problems are solved using hybrid approaches. To solve complex real-world problems, one particular meta-heuristic approach is inadequate to generate optimal results. Therefore, in this context, several heuristics and meta-heuristics approaches are merged together to form a hybrid approach. The main objective behind hybridization of meta-heuristics is to take the benefit of co- relative aspects of various strategies [11]. The performance of such strategies is often better than the individual approach. Nowadays, hybridization of evolutionary algorithms has taken a step forward and has become popular to solve complex optimization problems. This is all due to their efficiency to handle complexity, ambiguity, variability, imprecision and noisy environment occur in real world problems.

4.2 Beam Search

In this method, a graph is traversed and the most optimistic node is expanded in a finite set [18]. It is a search method to choose the most effective options to achieve a particular goal. It is a restricted version of best-first searches i.e. the memory requirements for storing alternative nodes are limited. In this approach, a predetermined number of best candidates is chosen as a solution. Beam Search constructs its search tree using a breadth-first search strategy in which the W number of best nodes is expanded at each level. Firstly, all the descendants of the current level of the tree are generated. At each level, it only evaluates the best W nodes and prunes the remaining nodes, and then moves to the next level. To select the best nodes, heuristic costs associated with the nodes are used. At each level, only W * B nodes will be considered, but only W will be selected, where B is the branching factor and W is the width of the beam search. If W = 1, the beam search is treated as a hill-climbing search where the best node is selected from the successive nodes. If W is infinity, beam search behaves the same as breadth-first search.

Beam search is a fast heuristic method whose performance depends on the "evaluation function" i.e. the heuristic value used to estimate the quality of a solution [18]. Beam search minimizes the time of a search by reducing the computation. Meanwhile, the risk of convergence to local optima is also reduced [31]. It was developed to achieve the optimal or sub-optimal solution without the consumption of too much memory. Only limited nodes are expanded, thus resources used are also limited. It is an improvement over the greedy technique by considering more than one option at a time. Sometimes, a search may not reach a goal or the search may not result in the optimal solution. Hence, Beam's search is likely to be incomplete. It terminates for 2 cases:

i) When the search reaches the

appropriate goal.

ii) When no nodes remained in the search space and the target node is not achieved.

In spite of having such disadvantages, Beam search has gained widespread success in the practical areas of job scheduling, speech recognition, vision, planning and machine learning, integrated circuit design, etc. [25]

4.3 Hybrid GA and Beam Search

Hybridization between the Genetic Algorithm and local-search approach is known as memetic algorithm (MA). MA is significantly faster and more accurate than a simple Genetic Algorithm to solve a different kinds of problems. To maintain a state of balance between global and local search in regards to computational effort and time has become an important challenge for MA [21]. MA can form a hybrid framework with different heuristic search methods like tabu search, hill climbing, simulated annealing, Beam Search, etc.

Algorithm 2: Proposed Memetic Algorithm

Procedure MA(ff, n, P_c, P_m, ngen)
// ff – fitness function
// n – population size
// P_c – crossover probability
// P_m – mutation probability
// ngen – maximum number of generations

Generate population P gen=1
while *gen <= ngen* **do**
calculate ff(n) evaluate min(ff) **for** *i= 1 to n* **do**
par1, par2 = Selection(n)
//apply beam search on selected individual
Bpar1 = BS(par1, *B*)
Bpar2 = BS(par2, *B*)
Child = Crossover(Bpar1, Bpar2)
Mchild = Mutation(Child)
//Replacement
P := Mchild
end
gen=gen+1
end
Return best individual

In order to increase exploitation, this proposed memetic algorithm includes beam search after selection phase of Genetic algorithm. Using this approach, improvement in genes of each individual is done before passing the individuals to crossover phase. Some standard benchmark functions i.e. sphere, rosenbrock, rastrigin and schwefel are considered to evaluate the performance of proposed MA. To produce global optimal solution, Genetic Algorithm is provided a well-defined problem with adequate time [4]. It provides wide and large solution space search ability. It provides parallelism and is easily modifiable and adaptable to different problems. Due to these advantages, Genetic Algorithm is selected to optimize different test functions. Beam Search maintains efficiency and reduce the risk of convergence to locally optima. [31]. Using this approach, computation is reduced, thus consumption of memory is also less.

So, to overcome the limitations of individual method, the concept of Memetic Algorithm is used which combine the advantages of both approaches. The performance of such memetic frameworks is better than their counterparts.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

Algorithm 3 Beam Search Algorithm

Procedure BS(Par, *B*)
// Par = currently best parent
// *B* = Beam width
while *terminate condition not specified* **do**
Generate neighbours of parent of *B* width
New sol := best neighbours(Par)
if *New sol is better than Par* **then**
Par := New sol
end end

5. Test Functions

Different benchmark test functions are available to evaluate the performance of genetic algorithm approach. Here, four test functions i.e. sphere, rosenbrock, rastrigin and schwefel function are considered to access the perfor mance of proposed MA.

Function	**Name**
F1	Sphere Function
F2	Rosenbrock Function
F3	Rastrigin Function
F4	Schwefel Function

Table 1: Benchmark Test Functions

5.1 Sphere Function

The Sphere function is appropriate for single-objective optimization, which means that it presents a single objective function. It is continuous, convex and uni-modal. It has d local minima besides the global minima. Its two- dimensional form is represented in Figure 1. Generally, hypercube is accessed using this function.

Mathematical definition:

$F\ 1(x) = \sum[x_i^2]$

$-5.12 <= x_i <= 5.12$

global minimum: fn(x) = 0, xi = 0

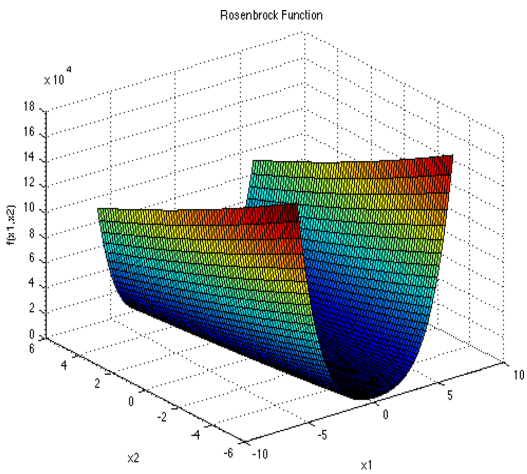

Figure 1: Sphere function

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

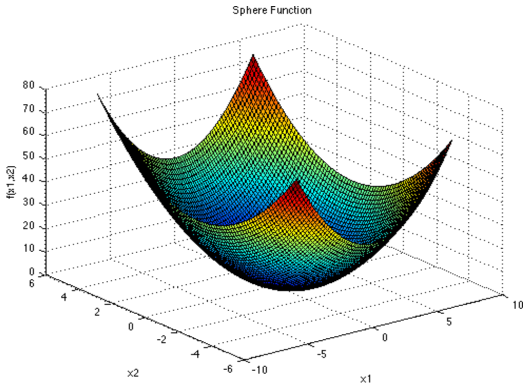

Figure 2: Rosenbrock function

5.2 Rosenbrock Function

Rosenbrock function is one of the famous benchmark function which is also known as the Banana function. The analytic result of gradient-based optimization algorithms can be analyzed using this function [2]. The global minima of this function occur in a valley which is narrow and parabola in shape. The paradigm for Rosenbrock function has some problem to recognize the significance of the parabolic valley & problem conjunction is required to resolve this technique [2]. Since, convergence of a problem becomes a difficult task to attain global optimum solution. Hence, the performance of optimization problems can be evaluated by using Rosenbrock function.

Mathematical definition:

$F\,2(x) = \sum[b(x_{i+1} - x_i^2)^2 + (a - x_i)^2]$

$-2.048 <= x_i <= 2.048$

where 'a' and 'b' are constant parameters and are normally may be assigned with values as: a=1 and b=100.

5.3 Rastrigin Function

Rastrigin function is highly multi modal function in which local minima is usually dispersed among different localities. It consists of more than two local minima. Figure 3 shows its two-dimensional form. Its complexity is O (n.log(n)), where n represents parameters of this function. Mathematical definition:

$F\,3(x) = 10 * n + \sum [x_i^2 - 10 * cos(2 * 3.14 * x_i)]$

$-5.12 <= x_i <= 5.12$

global minimum: fn(x) = 0 , xi = 0

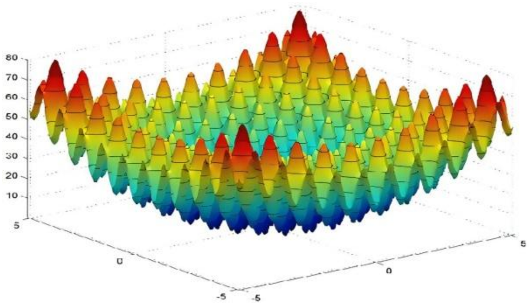

Figure 3: Rastrigin function

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

5.4 Schwefel Function

Schwefel function is a composite function with more than one local optima. In this function, the global minima are structurally located far away over parameter spaces from next best local optima. Consequently, search algorithms are vulnerable to converge in wrong aspect. Its two dimensional form is represented in Figure 4.

Mathematical definition:

$$F4(x) = \sum[(-x_i * sin(\sqrt{|x_i|})]$$

global minimum: xi = 420.9678, fn(x) = -418.9829n

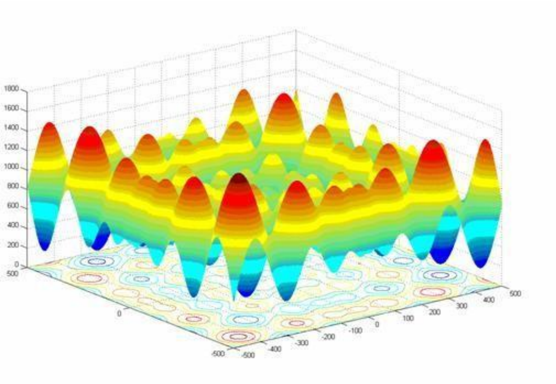

Figure 4: Schwefel function

$$-500 <= x_i <= 500$$

6. Results and Discussion

6.1 Experimental Setup

Here, the implementation is carried out using four benchmark test functions i.e. sphere function, rosenbrock function, rastrigin function and schwefel function. To access the performance of proposed hybrid approach, the coding is done using python. The work starts by choosing an initial population arbitrarily. Then, the fitness values of the chromosomes of population are calculated based on different test functions. Selection of best parents is done using Roulette wheel selection. The selected individual is then passed to the Beam Search algorithm. Then, the selected chromosomes pass through crossover and mutation phase to construct a new and better population. Then, the old population gets replaced by the new population. This entire process continues till the required number of generations or termination condition is fulfilled.

Parameter	Value
Population size	10, 20, 50
Number of Generation	100, 200
Encoding	Real Encoding
Selection	Roulette Wheel Selection
Crossover Probability	0.8
Mutation Probability	0.01

Table 2: Parameter Settings

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

6.2 Experimental Results

This section consists of the implemented python code. The executable code considers four benchmark functions to evaluate the performance of suggested approach of hybridization of Genetic Algorithm and Beam Search. Initial population, selection method, crossover and mutation probability have been kept same to compare the performance of both approaches in all functions. Minimum fitness values for various functions are computed for 100 and 200 generations and the results of two approaches are compared using graph.

N		10	20	50
Gen=100	SGA MA	0.0400 0.00136	0.0169 0.0024	0.0016 4e-06
Gen=200	SGA MA	0.1680 0.0144	0.3844 0.0001	0.0289 6.4e-05

Table 3: Minimum Fitness values for F1 function

N		10	20	50
Gen=100	SGA MA	12.508 1.502	7.291 4.975	4.093 0.372
Gen=200	SGA MA	9.455 4.623	0.419 0.396	1.802 0.349

Table 4: Minimum Fitness values for F2 function

N		10	20	50
Gen=100	SGA MA	-3.29 -9.097	-9.920 -9.998	-8.962 -10.0
Gen=200	SGA MA	-9.044 -9.508	-9.9208 -9.98	-9.822 -10.0

Table 5: Minimum Fitness values for F3 function

N		10	20	50
Gen=100	SGA MA	-401.24 -417.67	-388.145 -417.24	-417.67 -418.98
Gen=200	SGA MA	-193.33 -267.44	-411.95 -416.16	-395.75 -418.64

Table 6: Minimum Fitness values for F4 function

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

Table 2 shows the parameter settings used for implementation. Results for minimum fitness values of F1 function for 100 and 200 generations are given in Table 3. Table 4 shows minimum fitness values of F2 function for 100 and 200 generations. Results for minimum fitness values of F3 function for 100 and 200 generations are given in Table 5. Table 6 shows minimum fitness values of F4 function for 100 and 200 generations.

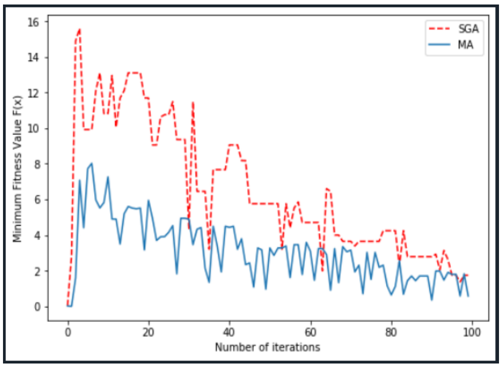

Figure 5: Comparison of minimum fitness
Value of F1 for 100 generations (N=10)

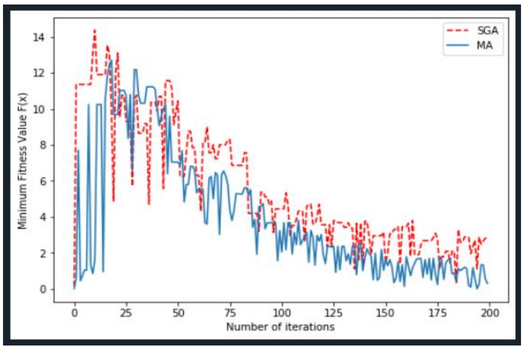

Figure 6: Comparison of minimum fitness
Value of F1 for 200 generations (N=10)

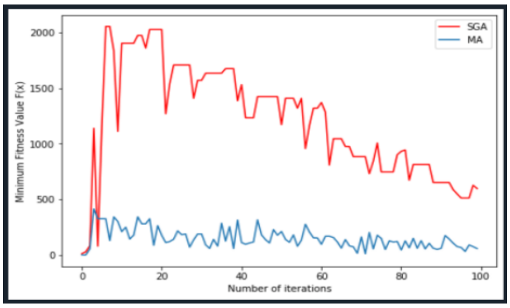

Figure 7: Comparison of minimum fitness
Value of F2 for 100 generations (N=10)

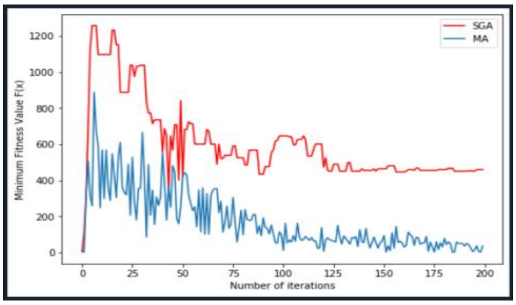

Figure 8: Comparison of minimum fitness
Value of F2 for 200 generations (N=10)

The performance of both algorithms of F1 function with initial population size as 10 for 100 and 200 generations are shown in Figure 5, Figure 6 respectively.
The performance of both algorithms of F2 function with initial population size as 10 for 100 and 200 generations are shown in Figure 7, Figure 8 respectively.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.1637-1651
https://publishoa.com
ISSN: 1309-3452

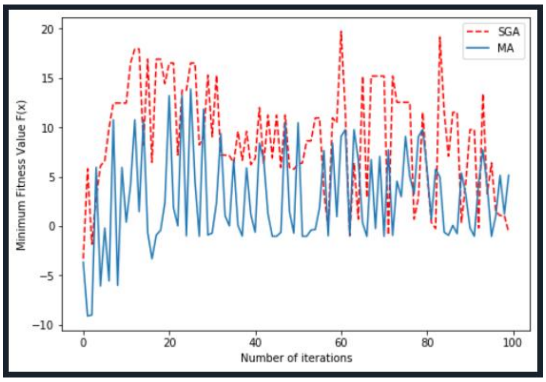

Figure 9: Comparison of Minimum fitness values of F3 for 100 generations(N=10)

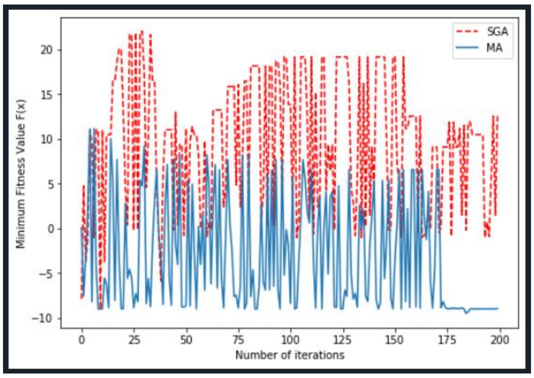

Figure 10: Comparison of Minimum fitness values of F3 for 200 generations(N=10)

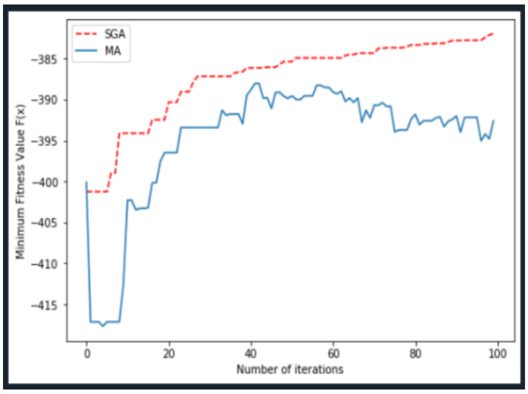

Figure 11: Comparison of Minimum fitness values of F4 for 100 generations(N=10)

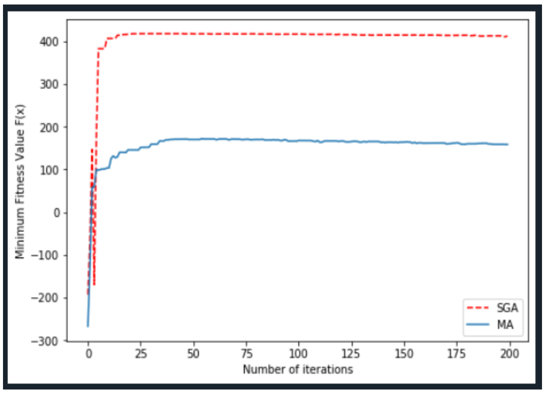

Figure 12: Comparison of Minimum fitness values of F4 for 200 generations(N=10)

Figure 9, Figure 10 represents the performance of both algorithms of F3 function with initial population size as 10 for 100 and 200 generations respectively. The performance of F4 function is shown in Figure 11, Figure 12.

It has been observed that the performance of proposed hybrid algorithm is considerably better than simple genetic algorithm approach with reference to optimality and convergence of solution. The proposed approach solves the problem of genetic drift and local optima and allows more diversity in the population. A state of balance can be maintained between exploration and exploitation within the search space if local search is combined with Genetic Algorithm.

7. Conclusion

Hybridization of local search with genetic algorithm improves the performance of hard optimization problems and also prevents premature convergence. This paper compares simple genetic algorithm and proposed hybrid approach. Four different benchmark functions are considered to perform experiments and the execution is done using python. This hybrid approach uses beam search algorithm in selection operator of Genetic Algorithm to enhance the performance of simple genetic algorithm. Using this approach, improvement in genes of each individual is done before passing the individuals to crossover phase. In doing so, diversity is maintained and the problem of genetic drift and local optima is also solved. A state of balance can be maintained between exploration and exploitation if local search is combined with Genetic Algorithm. Results reveal that the suggested memetic approach is considerably better than simple genetic algorithm approach. In future, this algorithm can be implemented in different genetic operators to validate its performance. Further, this proposed algorithm may be applied to various real world problems.

References

[1] Manoj Kumar, Mohammad Husain, Naveen Upreti and Deepti Gupta. Genetic algorithm: Review and application. *International Journal of Information Technology and Knowledge Management*, 2:451–454, 2010.

[2] Nasib Singh Gill, Kapil Juneja. Optimization of dejong function using ga under different selection algorithms. *International Journal of Computer Applications*, 64:28–33, 2013.

[3] Pablo Moscato, Carlos Cotta. A modern introduction to memetic algorithms. *International Series in Operations Research Management Science*, pages 141–183, 2010.

[4] Pratibha Bajpai, Manoj Kumar. Genetic algorithm – an approach to solve global optimization problems. *Indian Journal of Computer Science and Engineering*, 1:199–206, 2010.

[5] Sourabh Katoch, Sumit Singh Chauhan and Vijay Kumar. A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80:8091–8126, 2021.

[6] Amita Sharma, Kiran Jain. Hybridization in genetic algorithm with hill climbing to optimize the knapsack problem. *International Journal for Scientific Research Development*, 3:1112–1115, 2015.

[7] Ankita. Optimization of rosenbrock function using genetic algorithm. *Turkish Journal of Computer and Mathematics Foundation*, 12:3364–3368, 2021.

[8] Ankita Chhikara. Different selection strategies of genetic algorithm using rosenbrock function. *GIS Science Journal*, 8:789–799, 2021.

[9] Annu Lambora, Kunal Gupta and Kriti Chopra. Genetic algorithm- a literature review. In *International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (Com-IT-Con)*, pages 380– 384, 2019.

[10] Anu. Hybridizing genetic algorithm with hill climbing in replacement operator. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3:270–278, 2013.

[11] Christian Blum, Jakob Puchinger, Gunther Raidl and Andrea Roli. Hybrid meta-heuristics in combinatorial optimization: A survey. *Applied Soft Computing, Elsevier*, 11:4135–4151, 2011.

[12] Clara Meister, Tim Vieira, Ryan Cotterell. Best-first beam search. *Transactions of the Association for Computational Linguistics*, 8:795– 809, 2020.

[13] Crina Grosan, Ajith Abraham.

Hybrid evolutionary algorithms: Methodologies, architectures, and reviews. *Hybrid Evolutionary Algorithms Studies in Computational Intelligence*, 75:1–17, 2007.

[14] Daoxiong Gong, Xiaogang Ruan . A hybrid approach of ga and aco for tsp. In *Proceedings of 5th World Congress on Intelligent Control and Automation*, volume 3, pages 2068–2072, 2004.

[15] Emel Soylu, Ali Uysal. A hybrid genetic-ant colony algorithm for travelling salesman problem. *International Journal of Engineering Science and Application*, 1:86–90, 2017.

[16] Carlos Cotta, Ferrante Neri. Memetic algorithms and memetic computing optimization: A literature review. *Swarm and Evolutionary Computation*, 2:1–14, 2012.

[17] Lingaraj Haldurai. A study on genetic algorithm and its applications. *International Journal of Computer Sciences and Engineering*, 10:139– 143, 2016.

[18] I. Sabuncuoglu, M. Bayiz. Job shop scheduling with beam search. *European Journal of Operational Research*, pages 390–412, 1999.

[19] Jorge M.S.Valente, Rui A.F.S. Alves. Beam search algorithms for the early/tardy scheduling problem with release dates. *Journal of Manufacturing Systems*, 24:35–46, 2005.

[20] K.Jayavani, G.M.Kadhar Nawaz. Study of genetic algorithm, an evolutionary approach. *International Journal on Recent and Innovation Trends in Computing and Communication*, 2:2331–2334, 2014.

[21] M. Lozano, F. Herrera, N. Krasnogor, and D. Molina. Real-coded memetic algorithms with crossover hill-climbing. *Evolutionary Computation*, 12:273–302, 2004.

[22] Meenu Vaid, Amit Verma. Memetic algorithm: Hybridization of hill climbing with replacement operator. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4:737–743, 2014.

[23] Rakesh Kumar, Sanjay Tyagi and Manju Sharma. Memetic algorithm: Hybridization of hill climbing with selection operator. *International Journal of Soft Computing and Engineering (IJSCE)*, pages 140–145, 2013.

[24] S. Muthuraman,V. P. Venkatesan. A comprehensive study on hybrid meta-heuristic approaches used for solving combinatorial optimization problems. In *2017 World Congress on Computing and Communication Technologies (WCCCT)*, pages 185–190, 2017.

[25] Sandip Aine, Partha Pratim Chakrabarti. An analysis of breadth-first beam search using uniform cost trees. In *International Symposium on Artificial Intelligence and Mathematics*, 2010.

[26] Santosh Kumar Suman, Vinod Kumar Giri. Genetic algorithms: Basic concepts and real world applications. *International Journal of Electrical, Electronics and Computer Systems (IJEECS)*, 3:116–123, 2015.

[27] Sheng Li, Huiqin Chen and Zheng Tang. Study of pseudo-parallel genetic algorithm with ant colony optimization to solve the tsp. *International Journal of Computer Science and Network Security*, 11, 2011.

[28] T O Ting, Xin-She Yang, Shi Cheng and Kaizhu Huang. Hybrid meta-heuristic algorithms: Past, present, and

future. *Recent Advances in Swarm Intelligence and Evolutionary Computation*, 585:71–83, 2015.

[29] Yew Soon Ong, Natalio Krasnogor and Hisao Ishibuchi . Special issue on memetic algorithms. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics: a publication of the IEEE Systems, Man, and Cybernetics Society*, 37:2–5, 2007.

[30] Zainuddin Zukhri, Irving Vitra Paputungan. A hybrid optimization algorithm based on genetic algorithm and ant colony optimization. *International Journal of Artificial Intelligence and Applications*, 4:63–75, 2013.

[31] Zhong-wei Xu, Feng Liu and Ying-xin Li. The research on accuracy optimization of beam search algorithm. In *7th International Conference on Computer-Aided Industrial Design and Conceptual Design*, pages 1– 4, 2006.