COSC 6368 Artificial Intelligence

Final Project Summer 2021

Genetic Algorithms: with Species Consideration and Multi-Selection Policies using Clustering methods

Presented by

Mohan Reddy P (1993053)

Ram Narayan Reddy Tadi (2047211)

Vamshi Krishna Manne (2007919)

Abstract:

Genetic Algorithms is one area in AI which is Innovative concept and has many variation. However, most of the variations of Genetic Algorithm which are present today do not consider the grouping of various species in the population by using similarities and differences of the chromosomes. Also, presence of so many selection strategies has led to a difficult decision of choosing most optimal strategy for each application. In this Project we propose and implement an ensemble or Multi-Selection Strategy which uses many selection strategies in the same generation to make the GA more robust and scalable. Also, we group the various species in the population with the help of clustering methods like K-means Clustering and select each parent from a cluster to avoid converging of population on same individual . We also compare the results of the proposed methods by evaluating 4 selection strategies (Random Pick, Roulette pick, Tournament pick, Rank Pick) with and without clustering and also compare the same with Multi-Selection Genetic Algorithm

I. Introduction:

Artificial Intelligence (AI) is a broad field of computer science. It was coined by John McCarthy in 1956, however due to recent advances in technology and computing capacity of various systems, it has recently regained prominence. The goal of this field is to create smarter machines that can perform activities that require human intelligence. AI enables robots to learn from their mistakes, adapt to new inputs, and execute human-like jobs. The majority of AI instances and concepts we hear today involve using the AI's decision-making or analytical ability to solve problems. The Genetic Algorithm is one method for meeting AI's decision-making and analytical capability needs. We will examine the various existing research in Genetic Algorithms and suggest a technique to improve it utilizing machine learning clustering approaches in this report. The following is how the report is organized: In Section II we will discuss the various concepts and Literature review of Genetic Algorithms. In Section III, we will discuss our proposed methodology and Implementation along with the evaluation and comparison of results in section IV, we will discuss the challenges and Improvements based on our results and conclude the report in section V using a short summary.

II. Literature Review and Survey:

A. Genetic Algorithms:

Genetic Algorithms apply genetics and natural selection approaches to tackle programming issues and applications, as the name implies. It is a programming approach that is based on biological evolution. The Genetic Algorithm is a problem-solving technique that seeks out the best and most efficient answer. They're the most efficient way of dealing with a scenario in which little is understood. They will work well in any search field because they are a highly generic algorithm. The only thing to keep in mind is the specific situation. Mutation, Fitness Function, Crossover, and Selection are the main phases involved. The advantage of a generic method is that it can be readily parallelized and offers multi-objective optimization. It is, nevertheless, computationally costly and time intensive.

B. Existing Research and Strategies in Genetic Algorithms

Crossover and mutation rates have been proven to be important factors in GA search performance [1]. According to DeJong [2], the best population size range is [50–100], the mutation parameter rate is (0.001), and large mutation rates cause the search to be random. The crossover employed was based on one single point crossover at a rate of about 0. (0.6). A dynamic crossover and mutation rates were proposed by Dong and Wu [3]. Calculating the Euclidean distance between two chromosomes was used to compute the crossover rates dynamically [4].They also computed the Euclidean distance between the population's highest and lowest fitness values for each individual.

C. Fitness Functions

Simply explained, the fitness function is a function that takes a candidate solution to a problem as input and outputs how "fit" or "excellent" the answer is with respect to the problem at hand. The fitness function of N-queens of problem we are trying to solve in report is number of non attacking queens which takes a maximum value of Nc2(In case of 8 queens problem it will be 8c2=28)

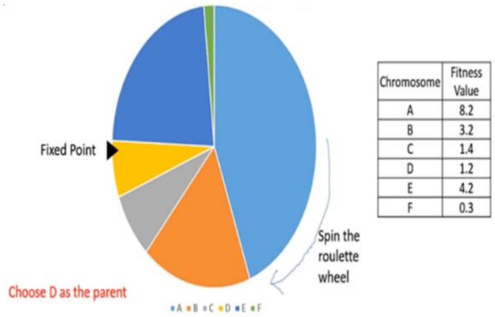
D. Various Selection Methods

Selection in genetic Algorithm:

Selection is the stage of a genetic algorithm in which individual chromosomes are chosen from a population for later breeding. The following are the various selection methods.

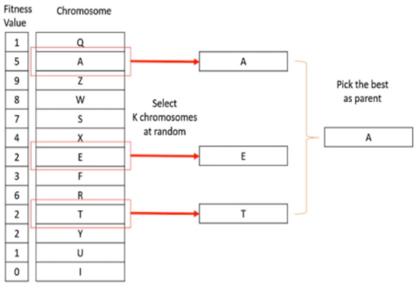
Random Selection: We choose parents at random from the existing population in this technique. Because there is little selection pressure for fitter people, this method is frequently ignored.

Roulette wheel : Fitness proportionate selection, often known as roulette wheel selection, is a genetic operator for picking potentially helpful crossover solutions in genetic algorithms. The fitness function assigns a fitness to feasible solutions or chromosomes in fitness proportionate selection, as in all selection procedures. This fitness score is used to link each chromosome to a chance of selection[5]. If f_i is the fitness of the individual, then its chance of being selected is given by $P_i = \frac{fi}{\sum_{j=1}^{N} fj}$ where N is size of the population. In a roulette wheel selection, the circular wheel is into n pies, where n is the number of individuals in the population. Each individual gets a portion of the circle which is proportional to its fitness value. A fixed point is chosen on the wheel circumference as shown and the wheel is rotated.



Chromosome	Fitness Value
A	8.2
B	3.2
C	1.4
D	1.2
E	4.2
F	0.3

Fig1: Roulette Selection

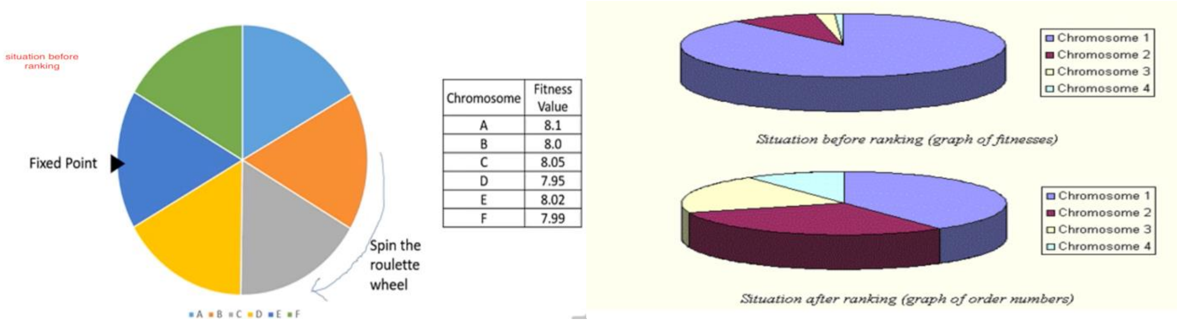


Fitness Value	Chromosome
1	Q
5	A
9	Z
8	W
7	S
4	X
2	E
3	F
6	R
2	T
2	Y
1	U
0	I

Fig2: Tournament Selection

Tournament Selection : Tournament selection is a selection strategy used to select the most suitable candidate from the current generation in the genetic algorithm. These selected candidates are then passed on to the next generation. In K-way tournament selection, we select individuals and run tournaments in them. Only the most suitable candidate among the selected candidates will be selected and passed on to the next generation. In this way, many of these contests were held and we ultimately selected candidates to enter the next generation. It also has a parameter called selection pressure, which is a measure of the probability that a candidate will participate in a competition. If the tournament is bigger, the weaker candidate has less chance of being selected because he must compete with stronger candidates. The selection of the pressure parameters determines the convergence speed of the genetic algorithm. The higher the selection pressure, the higher the convergence rate. GA can determine the best or near optimal solution within a wide range of selection pressures. Tournament selection also applies to negative fitness values.

Rank selection : Rank Selection sorts and ranks the population first based on fitness value. Then, based on its rank, each chromosome is assigned a selection probability[6]. Individuals are chosen based on their likelihood of being chosen. Rank selection is a method of selection that is used to discover new possibilities. In terms of selection pressure, rank selection differs from roulette wheel selection in that it inhibits too rapid convergence. Ranking uses a standard way of scaling across the population to control selective pressure. Rank selection has a more stable behavior than other methods. When fitness values are too near to one other or vary enormously, ranking algorithm can be utilized.



Chromosome	Fitness Value
A	8.1
B	8.0
C	8.05
D	7.95
E	8.02
F	7.99

Fig3: Rank Selection

Table1: Advantages and disadvantages of some Selection Strategies

Selection Strategy	**Advantages**	**Disadvantages**
Roulette wheel	A fitter individual has a greater pie on the wheel and therefore a greater chance of landing in front of the fixed point when the wheel is rotated. Therefore, the probability of choosing an	When the finesses differ much, this selection will fail. Outstanding individuals will add bias at the start of the search, which could lead to an early convergence and loss of diversity. Also, It doesn't work with negative fitness values

	individual depends directly on its fitness.	
Rank selection	Avoids a premature convergence. Works with negative fitness values	Every cycle, populations must be sorted.
Tournament Selection	When implementing in parallel it has efficient time complexity. And Fitness scaling or sorting not required.	This type of selection does not give guarantee to reproduction of best solution. [4]

E. Various Crossover methods

- **Single point Crossover:** The single point crossover is the most recognized and widely used crossover. The crossover point is randomly selected along the matched string and bits very close to the crossover point are swapped. If you choose a suitable location, you can obtain better offspring by combining good quality parents; Otherwise, it will seriously affect the quality of the string. At a single point crossing, the head and tail of a chromosome break off, if the head and tail have good genetic material, the offspring will not directly acquire these two good characteristics.
- **N point crossover:** In N-point cross over expect for crossover point, the rules used are the same as the we use in the single point cross. In 2 points crossover crossovers points are 2. Adding more and more crossover points will affect the interruption of the building block, which sometimes reduces the performance of the genetic algorithm, but allow head chromosomes to be accepted together in the offspring.

- **Uniform Crossover:** Uniform crossover does not fragment the chromosomes for recombination. Each gene in the offspring is copied from the parent according to the matching bit in the binary crossover mask, which is the same length as the parent chromosomes. The resultant gene is copied from the first parent if the bit in the crossover mask is 1, and the resultant gene is copied from the second parent if the bit in the crossover mask is 0. For each pair of parent chromosomes, a new crossover mask is created at random. The quantity of crossover points is not determined at first. As a result, the offspring have a combination of genes from both parents.
- **Three parent crossover:** Three parents are chosen at random in this crossover. Each gene from the first parent is compared to the gene from the second parent. If two genes are similar, the gene is occupied for offspring; otherwise, the corresponding gene from the third parent is used. It's usually employed when dealing with chromosomes that are binary encoded.
- **Arithmetic Crossover :** In the case of real-value encoding, arithmetic crossover is utilized. The arithmetic crossover operator joins the two parent chromosomes in a linear fashion. Two chromosomes are chosen at random for crossover, resulting in two offspring which are a linear mix of their parents.
- **Partially mapped Crossover:** The most commonly used crossover operator is Partially Matched or Mapped Crossover (PMX). Goldberg and Lingle offered it as a solution to the Traveling Salesman Problem. Two chromosomes are connected in Partially Matched Crossing, and two crossover sites are chosen at random. A comparable selection is formed by the proportion of chromosomes between the two crossover points, which undergoes the crossover process through position-by-position exchange procedures. PMX is a firm believer in absolute positions.

F. Mutation

The strings are subjected to mutation after crossover. The algorithm is prevented from becoming stuck in a local minimum by mutation. Mutation is responsible for both restoring lost genetic materials and randomly disrupting genetic information. It's a form of insurance from irreversible genetic material loss. Traditionally, mutation has been thought of as a basic search operator. If crossover is meant to take advantage of the existing answer in order to locate a better one, mutation is meant to aid in the exploration of the entire search space. Mutation is considered as a background operator that keeps the population's genetic diversity. It modifies some of the population's building pieces at random to introduce new genetic structures. Mutation aids in escaping the trap of local minima and sustains population variety. It also ensures ergodicity by keeping the gene pool well filled. Some of the well known mutation strategies are

- **Twors Mutation**: Twors mutation allows the exchange of position of two genes randomly chosen.
- **Centre inverse mutation (CIM**): There are two portions to the chromosome. Each part's genes are copied and then inversely placed in the same section of a child.
- **Reverse Sequence Mutation** : In the reverse sequence mutation operator, we take a sequence S and limit it to two randomly chosen locations i and j, such that i<j. In this sequence, the gene order will be reversed in the same way as it was in the preceding operation. This mutation operator is implemented in the below algorithm.
- **Partial Transfer Shuffle Mutation**: The Partial Transfer Shuffle (PSM) alters the order of some genes in the genotype, as the name implies. The steps of transformation are described in detail in the algorithm below.

- Table 2: Examples of some Mutation Strategies

Mutation Type	Parent	Child
Reverse Sequence Mutation	1 2 3 4 5 6	1 5 4 3 2 6
Centre inverse mutation	1 2 3 4 5 6	4 3 2 1 6 5
Twors Mutation	1 2 3 4 5 6	1 2 3 6 5 4

G. Introduction to Clustering

Clustering is the process of separating a population or set of data points into many groups so that data points in the same group are more similar than data points in other groups. In layman's terms, the goal is to separate groups with similar characteristics and assign them to clusters. Clustering may be classified into two categories in general:

- **Hard Clustering** - In hard clustering, each data point is either entirely or partially associated with a cluster.
- **Soft Clustering**: Instead of placing each data point in its own cluster, soft clustering assigns a chance or likelihood of that data point being in those clusters.

H. Use of ML in Genetic Algorithms

There have been many applications in ML, where generic algorithms are used to improve the machine learning techniques. In paper [8], the author presents an effective genetic algorithm that combines the capacity of genetic operators to conglomerate different solutions of the search space with the exploitation of the hill-climber. Here with the intervention of fast hill-climbing cycles of K-Means and obtain an algorithm that is faster than its predecessor and achieves clustering results of higher quality.

To tackle the express delivery path optimization problem, a novel set of immune genetic algorithms has been developed, which incorporates the static propagation concept and machine learning theory. Enhance individual immunity and raise the average fitness value of stocks by using adaptive vaccinations, in order to successfully avoid the loss of the best solution to narrow the search space. Increasing the rate of evolution allows the system to find the best answer in a short period of time. The method has a considerably greater accuracy than a basic genetic algorithm after verification, and the number of iterations required to get a stable solution is significantly decreased.

I. Use of Genetic Algorithms in ML

Paper[9]'s study offers a machine-learning-based technique for Android Malware Detection that use an evolving Genetic algorithm for discriminating feature selection. Machine learning classifiers are trained using selected features from the Genetic algorithm, and their ability to identify malware before and after feature selection is compared. The results of the experiments show that the Genetic algorithm provides the best optimal feature subset, reducing the feature dimension to less than half of the original feature set. After feature selection, machine learning based classifiers retain a classification accuracy of more than 94 percent while operating on a considerably smaller feature dimension, resulting in a positive impact on computational complexity of learning classifiers.

[10]. This paper presents a novel methodology for optimizing the traffic signal timings in signalized urban intersections, in the event of a one-time traffic incident they integrated fast running Machine Learning (ML) algorithms and trustworthy Genetic Algorithms (GA) into a single optimization framework with the goal of delivering quick and reliable decisions. Experimental results show that the combined GA-ML algorithm reduced total travel time by almost half when used under incident conditions.

III. Experimentation and Implementation:

A. Problem:

As seen in Section II, Table 1 each of the well-known Selection strategies that are currently in use are having various advantages and also drawbacks So, every time a new application is being used all the selection policies need to be studied in detail for optimal result. Also, one aspect being ignored by current genetic methods is the non-consideration of species of the population, i.e., each member of the current population can resemble another member with slight variations thus giving the entire population an possibility to be grouped into Species based in similarities and drastic differences

B. Solution and Methodology

In this Project, we suggest using Clustering algorithm to Differentiate various species choose a single parent from each cluster as a solution to consider Species in Genetic Algorithm. Also, to solve the issue of presence of multiple Selection policies each with their own advantages and disadvantages, we propose the use of multiple selection policies in each generation of the same Genetic Algorithm. Also, by combining these two , we implement the concept that 'Different Species use different selection policies for crossover'.

C. Implementation:

Implementation Code can be found at
https://github.com/MohanPutluru/AI_Genetic_Algorithm_Species_Consideration_and_MultiSelection_Policies_with-Kmeans

Using above solution , we try to solve the 8 queen problem by implementing the same in Python using the libraries pandas, numpy, random, sklearn and matplot lib and compare the results by analyzing the number of generations taken, Maximum fitness in each generation and average fitness in each generation. Also, for representation, we take the index of the gene as the Column position of the queen and value of the gene as the Row Position of the queen. For example, if we have chromosome as [1,5,7,8,6,7,3,2] then the position if first queen on chess board is (1,1) and the position of second queen is (2,5) and so on .We have the below Functions which help with the implementation.

- **Random_Being_Generator** (gene_length): Generates each individual of the population with random Genes
- **Fitness_Function** (x): Generates Fitness of queen arrangement by calculating the total number of collisions
- **Selection_Probability**(x): Probability of each member being selected based on their fitness
- **random_pick**(population, probabilities):Implementation of Random Pick Selection Method
- **roulette_wheel**(population): Implementation of Roulette Wheel Selection Method

- **tournament_selection**(population): Implementation of Tournament Selection Method
- **rank_selection**(population): Implementation of Rank Selection Method
- **Cross_Over**(x,y): Cross over 2 parents and Provide a new offspring. Since our focus is on selection method, most simple Single point Crossover is used here
- **Mutate_Child**(x): Mutates child to produce tiny gene changes
- **genetic_queen_selection**(population, fitness, sel_type) : implements basic Genetic Algorithm where parents are chosen based on sel_type
- **genetic_queen_kmeans_selection**(population, fitness, sel_type): implements basic Genetic Algorithm with K-means clustering where parents are chosen from each cluster based on sel_type
- **genetic_queen_kmeans_multiselection**(population, fitness, sel_type): implements basic Genetic Algorithm with K-means clustering where parents are chosen from each cluster based on different selection policies. The selection policy for each cluster is different. we have used 4 Clusters and 4 selection policies

D. Evaluation and Comparison:

We implement the code for 8 cases and the result can be found in Table3 and below Fig 4 ,5

Table 3: Results of each Case

Selection Policy	Clustering	Number of Generations	Solution
Random	No	5167	[3, 5, 8, 4, 1, 7, 2, 6]
Roulette	No	2292	[6, 1, 5, 2, 8, 3, 7, 4]
Rank	No	6369	[4, 2, 8, 5, 7, 1, 3, 6]
Random	Yes(Kmeans)	714	[6, 4, 2, 8, 5, 7, 1, 3]
Roulette	Yes(Kmeans)	4558	[3, 6, 4, 2, 8, 5, 7, 1]
Rank	Yes(Kmeans)	178	[4, 2, 8, 5, 7, 1, 3, 6]

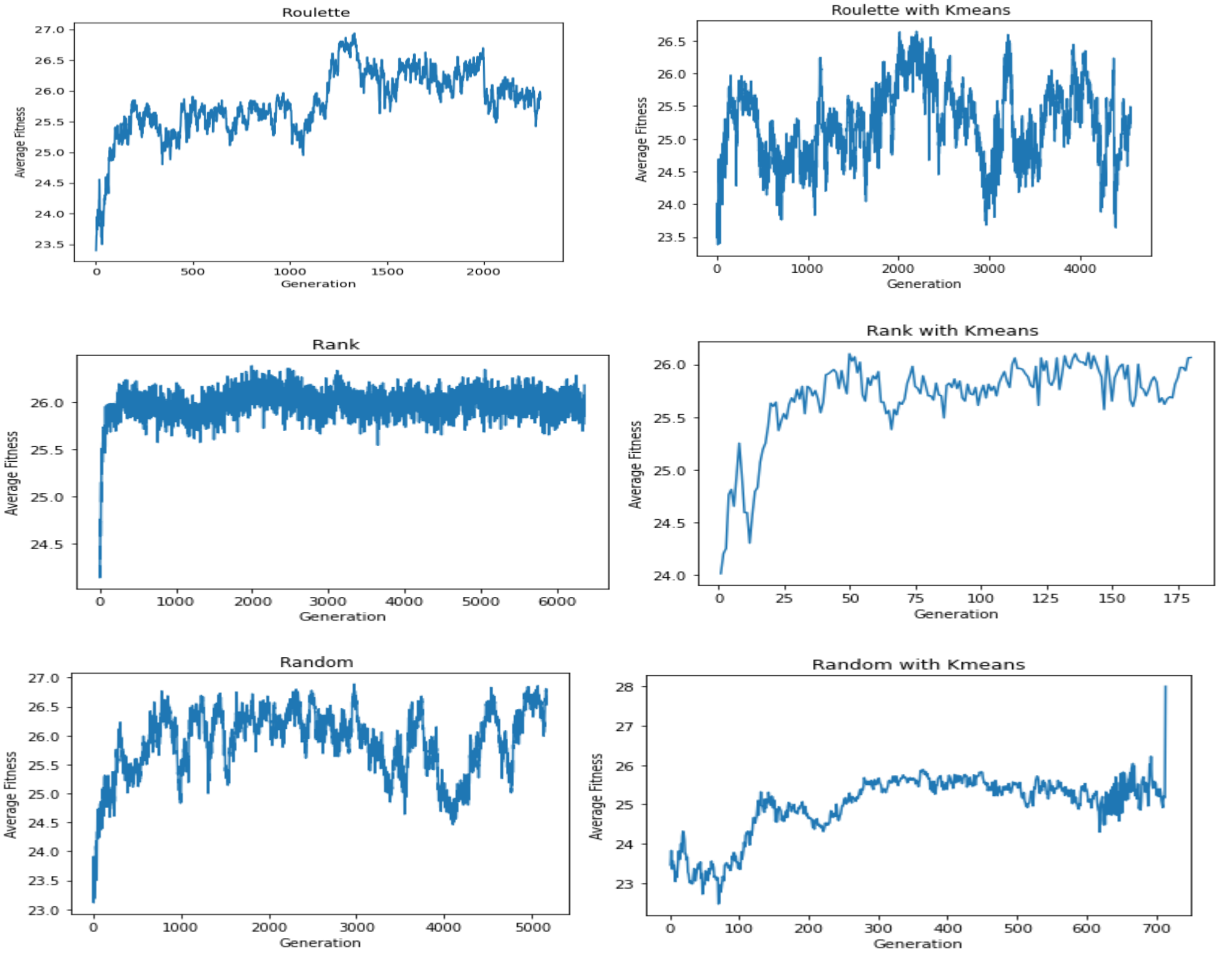


Fig3: Average Fitness over generation for each Strategy.

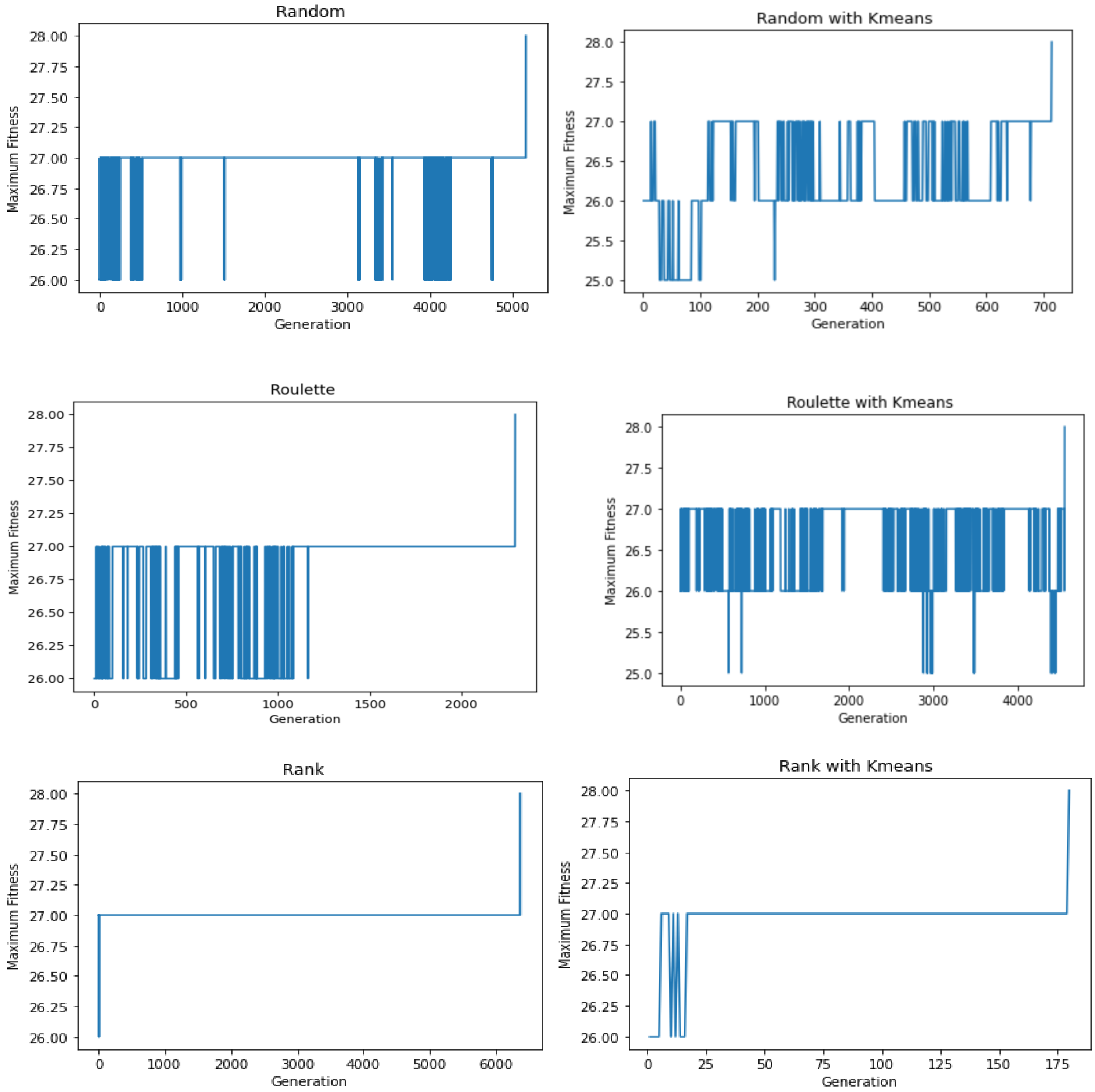


Fig3: Maximum Fitness over generation for each Strategy.

IV. Conclusion

As seen in the evaluation results in Fig 3, 4 and Table 3, we can conclude that Random and Rank Selection methods perform better when implemented with clustering. Also since there is randomness involved in the Genetic Algorithm, the results may vary for each set of initial population. However, it can be assumed that while using clustering, we can improve the effectiveness . Also inclusion of more multi selection policies will keep improving as we use more different startegies

V. References

[1]. Beasley, D.; Martin, R.R.; Bull, D.R. An overview of genetic algorithms: Part 1 Fundamentals. Univ. Comput. 1993, 15, 56–69

[2]. DeJong, K. Analysis of the Behavior of a Class of Genetic Adaptive. Ph.D. Thesis, University of Michigan, Ann Arbor, MI, USA, 1975.

[3]. Dong, M.; Wu, Y. Dynamic crossover and mutation genetic algorithm based on expansion sampling. In International Conference on Artificial Intelligence and Computational Intelligence; Springer: Berlin/Heidelberg, Germany, 2009; pp. 141–149.

[4] Firas Alabsi, Reyadh Naoum, 2012. " Comparison of Selection Methods and Crossover Operations using Steady State Genetic Based Intrusion Detection System", Journal of Emerging Trends in Computing and Information Sciences, Volume-03, No-07, July 2012 ,1053-1058.

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

[6] J. E. Baker, "Adaptive selection methods for genetic algorithms", Proceedings of an International Conference on Genetic Algorithms and their applications, pp 101-111, 1985.

[7] Firas Alabsi, Reyadh Naoum, 2012. " Comparison of Selection Methods and Crossover Operations using Steady State Genetic Based Intrusion Detection System", Journal of Emerging Trends in Computing and Information Sciences, Volume-03, No-07, July 2012 ,1053-1058.

[8]. Combining K-Means and a genetic algorithm through a novel arrangement of genetic operators for high quality clustering.

[9]. Android Malware Detection Using Genetic Algorithm based Optimized Feature Selection and Machine Learning

[10]. Boosted Genetic Algorithm Using Machine Learning for Traffic Control Optimization