Path Planning Optimization Using Genetic Algorithm – A Literature Review

[1,]Er. Waghoo Parvez , [2,]Er. Sonal Dhar
[1,] *(Department of Mechanical Engg , Mumbai University, MHSSCOE , Mumbai - 400008.*
[2,] *(Department of Production Engg, Mumbai University, DJSCOE, Mumbai – 410206.*

Abstract

This paper presents a review to the path planning optimization problem using genetic algorithm as a tool. Path planning is a term used in robotics for the process of detailing a task into discrete motions. It is aimed at enabling robots with capabilities of automatically deciding and executing a sequence motion in order to achieve a task without collision with other objects in a given environment. Genetic algorithms are considered as a search process used in computing to find exact or an approximate solution for optimization and search problems. There are also termed as global search heuristics. These techniques are inspired by evolutionary biology such as inheritance mutation, selection and cross over.

Keywords *- Chromosome, Genetic Algorithm (GA) , Mutation, Optimization, Path Planning.*

I. INTRODUCTION

Motion planning is a term used in robotics for the process of detailing a task into discrete motions. It is a process to compute a collision-free path between the initial and final configuration for a rigid or articulated object (the "robot") among obstacles. It is aimed at enabling robots with capabilities of automatically deciding and executing a sequence motion in order to achieve a task without collision with other objects in a given environment. Typically the obstacles and the mobile objects are modeled. Given a source position & orientation for mobile object and goal position & orientation, a search is made for a path from source to goal that is collision free and perhaps satisfied additional criteria such as a short path, a path which can be found quickly or a path which does not wander too close to any one of the obstacles. The general path planning problem requires a search in six dimensional spaces since the mobile object can have three translational and three rotational degrees of freedom. But still there are three dimensional search problems which have two translational and one rotational degrees of freedom. [10]The Genetic algorithm is an adaptive heuristic search method based on population genetics. Genetic algorithm were introduced by John Holland in the early 1970s [1].Genetic algorithm is a probabilistic search algorithm based on the mechanics of natural selection and natural genetics. Genetic algorithm is started with a set of solutions called population. A solution is represented by a chromosome. The population size is preserved throughout each generation. At each generation, fitness of each chromosome is evaluated, and then chromosomes for the next generation are probabilistically selected according to their fitness values. Some of the selected chromosomes randomly mate and produce offspring. When producing offspring, crossover and mutation randomly occurs. Because chromosomes with high fitness values have high probability of being selected, chromosomes of the new generation may have higher average fitness value than those of the old generation. The process of evolution is repeated until the end condition is satisfied. The solutions in genetic algorithms are called chromosomes or strings [2].

A genetic algorithm is a search technique used in computing to find exact or approximate solutions to optimization and search problems. Genetic algorithms are categorized as global search heuristics. Genetic algorithms are a particular class of evolutionary algorithms (EA) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover [7]. Genetic algorithms have been used to find optimal solutions to complex problems in various domains such as biology, engineering, computer science, and social science. Genetic algorithms fall under the heading of evolutionary algorithm. Evolutionary algorithms are used to solve problems that do not already have a well defined efficient solution. Genetic algorithm have been used to solve optimization problems (scheduling, shortest path, etc), and in modeling systems where randomness is involved (e.g., the stock market).

II. GENETIC ALGORITHM

2.1. Initialization

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions (the search space).

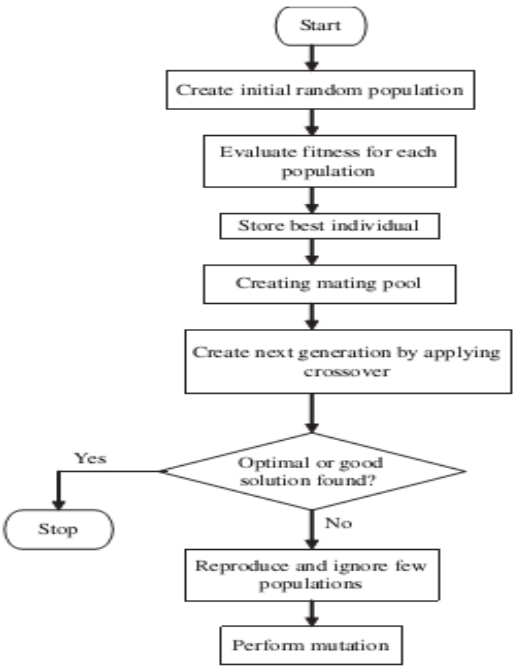


Figure No.1 Flowchart of GA [9]

2.2. Selection

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions.

Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Popular and well-studied selection methods include roulette wheel selection and tournament selection.

2.3. Reproduction

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation. For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously.

By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each new child, and the process continues until a new population of solutions of appropriate size is generated.

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best or genetic algorithm from the first generation are selected for breeding, along with a small proportion of less fit solutions.

2.4. Termination

This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria;
- Fixed number of generations reached;
- Allocated budget (computation time/money) reached;
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results;
- Manual inspection.

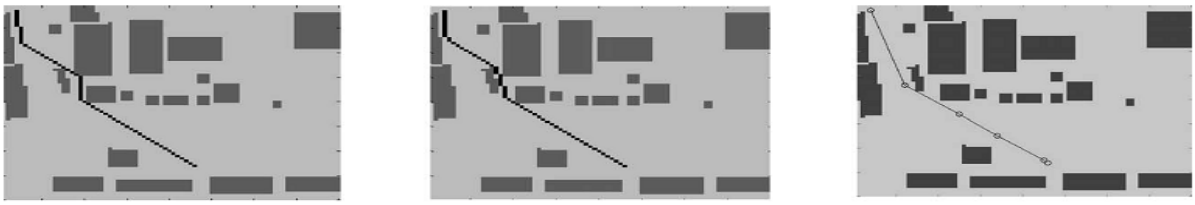
2.5 APPLICATIONS OF GENETIC ALGORITHM [8]

- Nonlinear dynamical systems–predicting, data analysis
- Robot trajectory planning
- Evolving LISP programs (genetic programming)
- Strategy planning
- Finding shape of protein molecules
- TSP and sequence scheduling
- Functions for creating images
- Control–gas pipeline, pole balancing, missile evasion, pursuit
- Design–semiconductor layout, aircraft design, keyboard configuration, communication networks
- Scheduling–manufacturing, facility scheduling, resource allocation
- Machine Learning–Designing neural networks, both architecture and weights.
- Signal Processing–filter design

III. LITERATURE REVIEW

3.1 Path planning in construction sites: performance evaluation of the Dijkstra, A*, and GA search algorithms A.R. Soltani, H. Tawfik, J.Y. Goulermas, T. Fernando [2] says:

The study illustrated the potential of deterministic and probabilistic search algorithms in addressing the site path planning issues with multiple objectives. The application generate the shortest path, low risk path, most visible path, and finally the path that reflects a combination of low risks, short distance, and high visibility between two site locations. Dijkstra algorithm can find optimal solutions to problems by systematically generating path nodes and testing them against a goal, but becoming inefficient for large-scale problems. A* can find optimal and near to optimal solutions more efficiently by directing search towards the goal by means of heuristic functions, reducing the time complexity substantially. These algorithms suffer from the curse of dimensionality effect, which limits the Dijkstra and A* operation to small and medium problems.



(a) (b) (c)

Comparison of the algorithms path cost solution

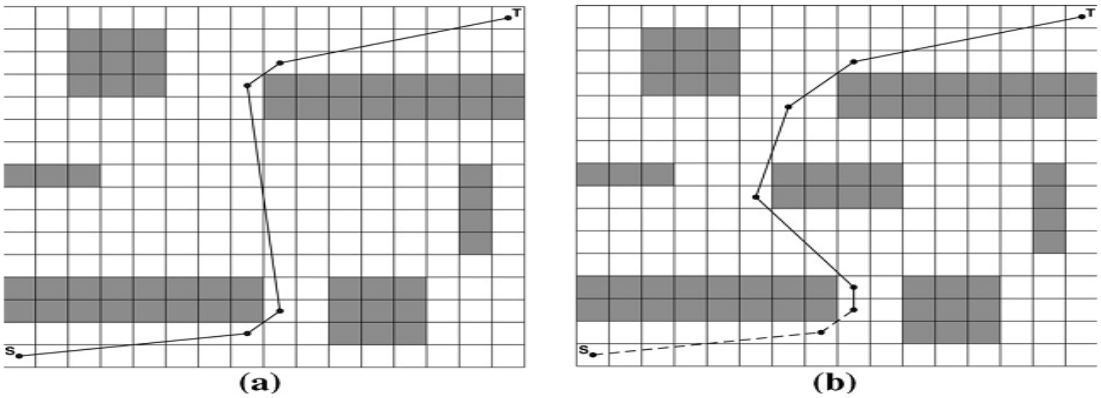
Path evaluation criteria	Dijkstra path cost solution	A* path cost solution	GA path cost solution
Distance minimisation	80.4	80.4	78.6
Safety minimisation	41.5	41.6	43.3
Visibility maximisation	43.5	43.7	42.5
Combined optimisation via G_1	35.3	43.2	39.5
Combined optimisation via G_2	29.7	38.9	30.2

Figure No.2 Courtesy [2] environment for research

Probabilistic optimization approach based on GA generates a set of feasible, optimal, and close-to-optimal solutions that captures globally optimal solutions. GA operators exploit the similarities in string structures to make an effective search. Good regions of the search space get exponentially more copies and get combined with each other by the action of GA operators and finally form the optimum or a near-optimum solution in substantially less time. The GA's performance limitations are mainly related to obtaining less accurate solutions and the time-consuming fine-tuning process to guide the search. The future avenues for this work include investigation of the applicability of fuzzy based multi-criteria evaluation, and hybrid optimization search algorithms.

3.2 Dynamic path planning of mobile robots with improved genetic algorithm Adem Tuncer , Mehmet Yildirim [3] says:

They have improved a new mutation operator for the GA and applied it to the path planning problem of mobile robots. The improved mutation method simultaneously checks all the free nodes close to mutation node instead of randomly selecting a node one by one. The method accepts the node according to the fitness value of total path instead of the direction of movement through the mutated node. It is clearly seen from the results that the GA with the proposed mutation operator can find the optimal path far too many times than the other methods do. The average fitness values and the average generation numbers of the proposed method are better than the other methods.



Experimental results for the initial environment in Fig. 5a.

	# of optimal solution	# of near optimal solution	# of infeasible solution	Fitness value	Generation number	Solution time (s)
Random mutation	1	86	13	31.78	21	0.28
Mutation in Ref. [14]	3	69	28	29.25	23	0.31
Mutation in Ref. [20]	2	69	29	29.91	22	0.46
Mutation in this study	54	44	2	27.82	11	0.89

Experimental results for the modified environment in Fig. 5b.

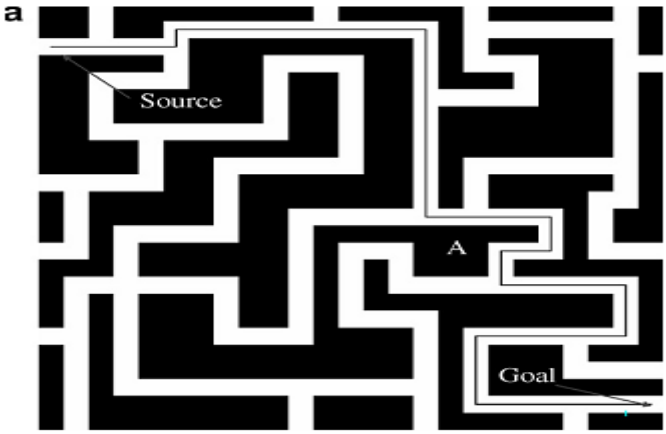
	# of optimal solution	# of near optimal solution	# of infeasible solution	Fitness value	Generation number	Solution time (s)
Random mutation	0	95	5	35.37	23	0.20
Mutation in Ref. [14]	0	95	5	31.21	22	0.26
Mutation in Ref. [20]	0	89	11	30.87	23	0.41
Mutation in this study	44	56	0	29.08	11	0.86

Figure No.3 Courtesy [3] environment for research

3.3 Multi-robot path planning using co-evolutionary genetic programming Rahul Kala [4] says:

Motion planning for multiple mobile robots must ensure the optimality of the path of each and every robot, as well as overall path optimality, which requires cooperation amongst robots. The paper proposes a solution to the problem, considering different source and goal of each robot. Each robot uses genetic programming for figuring the optimal path in a maze-like map, while a master evolutionary algorithm caters to the needs of overall path optimality. Co-operation amongst the individual robots, evolutionary algorithms ensures generation of overall optimal paths. Experiments are carried out with a number of maps, scenarios, and different speeds. Experimental results confirm the usefulness of the algorithm in a variety of scenarios.The modeling scenario has a maze like map where the different robots are initially located at distinct places and are given their own goals that they are supposed to reach. It further assumes that each robot moves with its own

speed. The algorithm makes use of co evolutionary genetic programming. At the first level a linear representation of genetic programming is used. The individual in this case consists of instructions for movement whenever a cross is encountered. The other level consists of a genetic algorithm instance. This algorithm selects the individuals from the genetic programming and tries to generate a combination such that the overall path of all the robots combined is optimal.



Summary of situation and results for simulation with 2 robots.

S. No.	Factor	Robot 1	Robot 2
1	Source	(0,2)	(24,23)
2	Goal	(24,23)	(0,2)
3	Speed	0.5	0.3
4	Reached Goal	Yes	Yes
5	Collision	No	No
6	Time	142	383
7	Path Length	66	115

Figure No.4 Courtesy [4] environment for research

3.4 A new vibrational genetic algorithm enhanced with a Voronoi diagram for path Planning of autonomous UAV Y. Volkan Pehlivanoglu [5] says:

The algorithm emphasizes a new mutation application strategy and diversity variety such as the global random and the local random diversity. Clustering method and Voronoi diagram concepts are used within the initial population phase of mVGA process. The new algorithm and three additional GA's in the paper are applied to the path planning problem in two different three-dimensional (3D) environments such as sinusoidal and city type terrain models and their results are compared. The first mutation operator is applied to all genes of the whole population and this application provided global but random diversity in the population. The global diversity afforded a chance for the population to escape from all local optima. The second mutation operator was specifically applied to the genes of an elite individual in the population. This application provided local diversity leading to a fast convergence. From the results obtained, it is concluded that a Voronoi supported multi frequency vibrational genetic algorithm is an efficient and fast algorithm since it avoided all local optima within relatively short optimization cycles.

3.5 Path planning on a cuboid using genetic algorithms Aybars UG˘UR [6] says:

Optimization on a cuboid has potential applications for areas like path planning on the faces of buildings, rooms, furniture, books, and products or simulating the behaviors of insects. This paper, addresses a variant of the TSP in which all points (cities) and paths (solution) are on the faces of a cuboid. They developed an effective hybrid method based on genetic algorithms and 2-opt to adapt the Euclidean TSP to the surface of a cuboid. The aim is to develop a simple and efficient method to find the optimum route visiting all items on a cuboid, one of the most common man-made object shapes. They selected a good TSP solving hybrid method based on GA and 2-opt that has been used for many years. They integrated it with an algorithm developed to calculate the distances of any two points on a cuboid; this implementation allows the hybrid method to be replaced by faster TSP solvers. In accordance with the main goal of this study, the first TSP optimization results were obtained and presented for different point densities in the cuboid environment. A second contribution is the presentation approach of the solution and environment. The user can rotate or scale the cuboid and trace the

optimum path easily.

GA Parameters

GA parameter	Default	Min	Max
Generation Size	100	1	65,535
Population Size	250	3	65,535
Crossover Rate	0.80	0.0	1.0
Mutation Rate	0.05	0.0	1.0

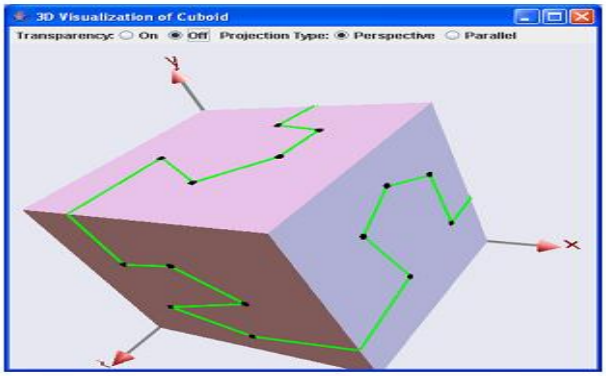


Figure No.5 Courtesy [6] environment for research

IV. CONCLUSION

Based on the various papers studied and textbooks referred it can be concluded that for multi objective optimisation problems genetic algorithms find variety of applications in different fields. Optimal path planning is one of the important factors in scheduling, moving, transporting, etc.Various factors considered for optimal path planning includes avoiding obstacles, minimum path finding, multi objectives constraining, safe distance travelling, etc. For finding the optimal path various optimization techniques are used out of which genetic algorithm finds extreme applications as it gives optimum results considering global population.If proper mathematical model is formed it can give the optimum results in optimum time for path planning/motion planning in the field of robotics.

REFERENCES

[1] Manoj Kumar, Mohammad Husian, Naveen Upreti, & Deepti Gupta "Genetic Algorithm: Review & Application" International Journal of Information Technology and Knowledge Management July-December 2010, Volume 2, No. 2, pp. 451-454.

[2] A. R Soltani, H Tawfik, et al" Path planning in construction sites: performance evaluation of the Dijkstra, A* and GA search Algorithm", Advanced Engineering Informatics pg 291-303.

[3] Adem Tuncer, Mehmet Yildirim "Dynamic path planning of mobile robots with improved genetic algorithm" Computers & Electrical Engineering submitted for publication.

[4] Rahul Kala "Multi-robot path planning using co-evolutionary genetic programming" Expert Systems with Applications 39 (2012) 3817–3831.

[5] Y. Volkan Pehlivanoglu "A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV" Aerospace Science and Technology 16 (2012) 47–55.

[6] Aybars UG˘UR "Path planning on a cuboid using genetic algorithms" Information Sciences 178 (2008) 3275–3287.

Books:

[7] David E. Goldberg , Genetic Algorithms in search, optimization & Machine Learning" (Pearson Education Twelfth Impression 2013).

[8] Melanie Mitchell, An Introduction to Genetic Algorithms (Prentice Hall of India Edition 2005).

[9] S.N.Sivanandam S.N.Deepa Introduction to Genetic Algorithms (Springer Edition 2008)

Website:

[10] http://en.wikipedia.org/wiki/Motion_