Indian Journal of Science and Technology, Vol 8(35), DOI: 10.17485/ijst/2015/v8i35/86639, December 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Layered Approach for Three Dimensional Collision Free Robot Path Planning using Genetic Algorithm

K. S. Suresh[1*], V. Vaithiyanathan[1] and S. Venugopal[2]

[1]School of Computing, SASTRA University, Thanjavur - 613401, Tamil Nadu, India; kssuresh@cse.sastra.edu, vvn@it.sastra.edu
[2]Indira Gandhi Centre for Atomic Research, Kalpakkam - 603102, Tamil Nadu, India; venu@igcar.gov.in

Abstract

Robot Path Planning (RPP) is solved using Genetic Algorithm (GA) principle for collision-free navigation for the three dimensional static space to find the optimal path. In this paper, Layered Approach is employed where the whole three dimensional space is considered as layers of two dimensional spaces to accomplish the RPP to reach the target by avoiding obstacles and find the shortest path. The quality of the path is ensured by nearest neighbourhood approach. Implementation of the principle of GA to solve RPP is effective where the environment contains huge number of solution paths compared to the classical methods to obtain the shortest path from source to target. This approach is tested for different number of layers and the results are tabled. The path generated and optimal path obtained by the implementation of this approach has been compared with the cost of the optimal path obtained manually.

Keywords: Collision-Free, Genetic Algorithm, Robot Path Planning, Three Dimensional Spaces

1. Introduction

In the research domain of robotics, path planning is an area which creates lot of room for different dimensions of research and solution models. It includes almost all the fields of science and technology, since Robot Path Planning (RPP) involves mechanical design and movements, electrically and electronically controlled components and artificial intelligence to take dynamic decisions to meet the demand of the environment. While taking dynamic decisions, competence of the robot are to be taken into account to analyse and devise a model to find the solutions.

The finding solution for a robot starting from a source and to reach the target, RPP is categorized into NP-hard problem by[1]. The solution to the RPP not only depends on the physical design and type of the robot, but also the environment it is going to navigate as well. The space in which the robot has to move may be static or dynamic. The static environment is a predictable environment or predefined navigational space to the robot which is easier to simulate and find the solutions.

The research in the domain of RPP was instigated in 1970s, by[2]. The classical methodologies were employed for RPP such as Roadmap by[3,4], Cell Decomposition by[5], Potential Fields by[6] and Mathematical Programming etc., by[7] but the time complexity of the system is a drawback for these methodologies. On the other side, Methodologies with probabilistic approach are implemented to accomplish the task of RPP such as Probabilistic Road Maps

*Author for correspondence

proposed by[8], Rapidly Exploring Random Trees used by[9], Level Set methodology implemented by[10] and Linguistic Geometry by[11] which are all affected by the problem of trapping into the local minima. On the other hand, the emergence of Meta heuristic algorithms was surveyed by[12], to elaborate the paradigm shift of the approach to solve the RPP. The popular heuristic algorithms are interleaved to get better results as implemented by[13]. Similarly various methodologies were combined to create hybrid technologies such as Neural Network, Genetic Algorithm, Simulated Annealing, Ant Colony Optimization, Particle Swarm Algorithm, Stigmergy, Wavelets, Tabu Search, and Fuzzy Logic. In[14] discussed all pros and cons of population based algorithms. But the Genetic Algorithm principles are employed for RPP with variations by[15,16] and pooled with other techniques proposed by[17].

In this paper, the section 3 describes the general methodology to implement Genetic Algorithm (GA) principle. Section 4 elaborates the layered model and in section 5, the implementation of the model is explained. The results are tabled and discussed the effectiveness of the approach in section 6.

2. Ideology of Genetic Algorithm

Principally GA simulates the process of evolution and is based on the concept of survival of the fittest. John Holland and David Goldberg proposed the GA for implementing optimization problems. Fundamentally GA consists of three significant phases as selection, crossover and mutation. Other tasks required to implement the GA are formulating the objective function with minimization or maximization of objective function and fitness function formulation demanded by the application. The individual chromosome of the population can be represented by so many ways, the choice depends on the processing methodology for the taken population and the interpretation of the output required. All the possible chromosomes generated are consumed by the phases of GA. For the selection phase, various methods are possible to select the better parents likely to produce the better offspring for the next generation based on the fitness value. Also the number of parents selected to produce the next generation is also playing the vital role. In the following phases of GA, the appropriate crossover rate and mutation rate assumed are the influencing factor for the rate of convergence to the optimal result. The high crossover rate and low mutation rate is recommended for earlier convergence of optimal result experimented by Suresh, et.al[18].

3. Proposed Layered model

For any problem to be solved, the approach to attack the problem to get the solution is determined by the available environment and the given input. As for as the RPP is concerned, the environment may be defined as the navigational space, may be static or dynamic space, two dimensional or three dimensional space, the capacity of the robot in terms of physical capability as degree of freedom, control resolution of the robot in linear and angular dimensions to turn and to reach the target from source, by avoiding obstacle in the path.

In a two dimensional scenario, the cell decomposing approach is popular, if the navigation space contain the objects to avoid is not so complex. According to this principle, the entire space will be divided into array of cells and cells are denoted to represent either the obstacle or the path which depends on the algorithm used to solve RPP.

In the proposed model, the moving space for robot is considered as the three dimensional space and mapping of the space done by splitting the space into cells as in two dimensional space, but the third dimension is considered as the layers of cells. The cell in each layer is either path or obstacle.

The number of layers and cells is influenced by

- Control resolution of the robot.
- Minimum space between the obstacles.
- Size of the robot arm.

The number of layers and cell in each layer can be increased in turn to attain the shortest path to reach the target, but it depends on the resolution of the robot. But the computational complexity will be increased manifold by increasing number of layers and cells which obviously increases the time complexity. Therefore for the better performance, optimal number of cells and layers are to be found.

The proposed model can be principally categorized into 4 stages and 9 phases as shown in Figure 1. At the end of 9 phases, it will be checking for exit condition as either accomplishment of determined number of generations or size of population becomes less than 2.

With the assumed layers and cells, generate all the possible paths. The generated paths are considered as

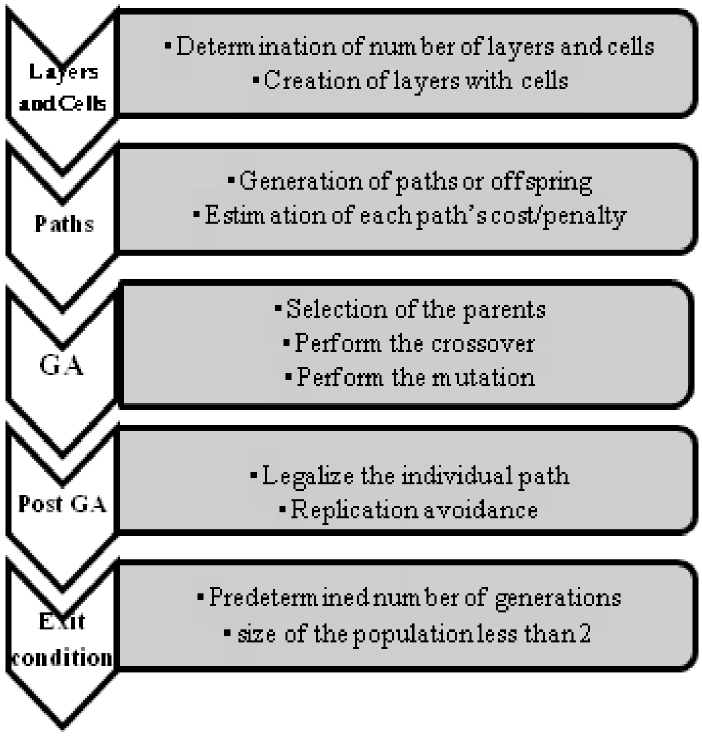

Figure 1. Stages and phases of the proposed model.

the population are to be fed to the GA implementation. From the population the parents are selected using tournament selection algorithm. Based on the fitness value as in this case, the costs of the individuals are accounted to select the fittest parents and apply the genetic operators such as crossover and mutation. The selection of crossover rate and mutation rate also influence the convergence speed to attain the optimal result. As far as this problem is concerned, the population has to undergo legalization phase to convert individuals as feasible. These phases are to be iterated for the predefined number of generations or until the number of individuals in the population reaches the two, since to perform crossover and mutation operation minimum of two parents required.

3.1 Phases of Models

The phases of models are depicted in the Figure 1. The each phase of the model may consist of two or more steps to be done.

3.1.1 Layers and Cells

The first step of this model is determining the number of layers and cells to be assumed to virtualize the navigational space of the robot. The number of layers may be determined by the computational capacity and control resolution of the robot. For the programming convenience, the number of cells is also taken as equal to the number of layers. The layers will give the three dimensional view of the navigation space. In each layer, if the number of cells is different from number of layers then that has to be maintained as a separate array. Hence to reduce the overhead, the number of layers is taken as equal to the number of cells in each layer. After determining the number of layers, the cells are filled with appropriate elements depends on whether the particular cell occupied by the obstacles or not. The cell elements of each layer are filled with ones if obstruction present in the cell, otherwise filled with zeros.

3.1.2 Path Population Generation

Picking up one zero elements from each layer and reach the target will constitute a path. Similarly all combination of the zero cells from each layer will form the pool of possible paths which are considered as the population to be consumed by GA. To derive all the possible paths, recursive calling of function can be used effectively in the programming level implementation.

3.1.3 Cost Realization with Penalty Quotient

The cost function estimates the cost for each path. While navigating from one layer to another layer, if it is going to be in the same row and same column, only then the cost will be added. In the case of different rows and columns, depends on the number of rows and columns it has to travel, the penalty will be added. The penalty quotients determine the path quality while selecting the best path through the cost function.

3.1.4 Implementation of GA Operator

Since the huge size of the population space involved, the crossover and mutation operators of GA determines the rate convergence as well as the quality of the result. 0.9 and 0.1 are the crossover mutation rate assumed respectively for the implementation[18]. Probability distribution function is used for selection of an individual path to perform the crossover. Though different crossover techniques are available, the single point crossover is preferred in this implementation for simplicity. The high crossover and low mutation rate will fetch speedy convergence to the better result.

3.1.5 Legalization of Paths

Implementation of the crossover and mutation phases of the GA may lead to the population with the paths which are not feasible for the given scenario. Therefore it is necessary to add the legalization phase to convert the infeasible paths to feasible paths. During this conversion phase, nearest neighbourhood concept is employed.

3.1.6. Avoidance of Replication

After these phases of implementation, there is a chance for replication of same paths. This duplication of paths must be scanned and removed to get the unique paths to avoid unnecessary overhead due to processing the same paths in turn it leads to the more number of generations to reach the optimal result.

3.1.7 Creation of Pool of Population for Mating

From the population generated, the binary tournament selection methodology is applied to select the parents from the population to produce the offspring for the next generation. According to the binary tournament selection, two randomly selected parents are compared with the estimated cost and an individual with lower costs will be qualified as the parent to participate in the process of producing the offspring for the next generation.

3.1.8 Exit Condition

One generation comprises of above phases and at the end of each generation it is verifying the occurrence of exit condition.

Two exit conditions are possible to break the generation as in the Figure 1.

- Completion of predetermined number of generations.
- Size of the population is less than 2.

These two conditions are to be treated differently. If the condition 1 occurs, best out of remaining population is to be identified and that will be considered as the optimal path. It may be achieved through sorting out the remaining population and selecting the first or last as the optimal path depends on the ascending or descending order sorting is performed respectively on the population.

In the case of condition 2, remaining path is going to be the optimal path.

If the above exit conditions are not satisfied, the next generation will be starting with the population through offspring produced by the previous generation.

4. Implementation of Layered Model

The layered model is implemented and tested for 3, 4, 5 and 6 layers and each layer consists equal number of rows and columns of cells (i.e., 3x3, 4x4, 5x5 and 6x6 arrays respectively) for computational convenience. It need not be the same always as the number of layers equal to number of rows and columns of cells. The rows and columns may be increased or decreased at the cost of increased complexity.

For implementation purpose, the layers and cells are generated at random manner with zeros and ones. The zero elements represent availability of path and the element one denotes the obstacle. Then the individual path is generated by taking the index of the first occurrence of the zero elements from the first layer and this is progressed to the consecutive layers which constitute a path. In the same way all other combinations are taken and formed pool of paths, observed as the population space to be processed.

Then the cost of each path is estimated based on the previous row and column index and current row and column index of the path. If it is away from the current row and column, to quantify the deviation, the penalty will be imposed on the path cost. This penalty quotient is excised to get the competent quality path from the huge size of population.

The GA operator is employed on the population. The 90% of the population are involved in the process of crossover to achieve the better optimal result. The 10% of the population are participated in the mutation operation for effective solution space.

During the crossover and mutation, the population may be modified, which may contain infeasible points of navigation path. Therefore each path of population space has to be scanned and to be converted into feasible path.

The populated paths from previous phases may not be the feasible path, because of implementation of genetic operators. To stabilize the paths, these are to be legalized. Instead of legalizing the paths with arbitrary indices, the implementation find the nearest neighbourhood of the index the path with least cost in turn it qualified as better path for the next generation.

After the conversion, there is a possibility of duplication of paths which are to be eliminated. Then the remaining will be the population to be fed to the selection phase. During this phase, the tournament selection methodology has been adopted where two randomly selected parents are compared with their costs and the path with minimum cost is selected for the next generation. The exit condition is verified and either go to the initial phase or exit from the iteration.

6. Discussion

The result of implementation for various numbers of layers and cells are listed. Analyzing the results from the implementation, it is observed that, if the layers are increased the number of possible paths would be increased exponentially. But the paths will have more resolution. Comparing the manually achieved better path with system generated optimal path, it is need not be always same. However the cost estimated for these paths may be very closer though they are not the same. By increasing number of generations, equivalent result may be accomplished but at the cost of increasing time factor tested by Suresh, K.S.et al[18].

The Table 1 shows the number of layers and cells assumed, the results obtained from the implementation of the proposed model which compared with manually calculated results. Consider the number of layers taken is 3 and cells in each layer also 3x3 = 9. Since the size of matrix is less in size, it is easier to find the better path manually. Specifically in this case, the following is the randomly generated matrix where 0s pave the way for the target, on the other side the 1s in the matrix represents the obstacles in the space.

$$\begin{matrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix} \qquad \begin{matrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{matrix} \qquad \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$$

From the possible paths populated by the represented matrix are 224 paths. The manually calculated best path and model generated optimal path and their corresponding costs are enumerated for observation.

Manually generated best path: 1 1 1 1 1 2 => cost: 6
Optimal path arrived : 1 1 2 1 2 1 => cost: 6

7. Conclusion

The layered approach is implemented and discussed the results achieved for various numbers of layers and cells for RPP problem to derive the shortest path by avoiding obstacles. Through layered model population is generated and GA principle is effectively implemented with appropriate methodologies for each phase. The efficient rate of crossover and mutation are implemented to accomplish the effective cost depends on the requirement

Table 1. Comparison of manually calculated and system implemented results for different layers and cells

Number of layers tested	3+2	4+2	5+2	6+2
Number of cells in each layer	3	4	5	6
Number of paths generated	224	5120	191664	21848400
Cost of best path manually calculated	6	8	9	10
Cost of optimal path	6	9	10	10

by determining the size of the layers, cells and number of generations. The nearest neighbourhood concept followed to maximize the quality of the paths.

8. References

1. Raja P, Pugazhenthi S. Optimal path planning of mobile robots: A review. International Journal of Physical Sciences. 2012; 7(9):1314–20.
2. Munson JH. Robot planning, execution and monitoring in an uncertain environment. Proc Int Jt Conf Art Intel; London. 1971. p. 338–49.
3. Lozano-Perez T. Spatial planning: A configuration approach. IEEE Transactions on Computers. 1983; 32(3):108–20.
4. Dunlaing OC, Yap CK. A retraction method for planning the motion of a disc. Journal of Algorithms. 1985; 6(1):104–11.
5. Robotin R, Lazea G, Dobra P. Mobile robot navigation using graph search techniques over an approximate cell decomposition of the free space. Advances in Intelligent Systems and Computing. Berlin Heidelberg: Springer-Verlag. 2013; 187:129–42.
6. Zhang PY, Lu TS, Song LB. Soccer robot path planning based on the artificial potential field approach with simulated annealing. Robotica. 2004; 22(5):563–6.
7. Habibi G, Masehian E, Beheshti MTH. Binary integer programming model, of point robot path planning. IEEE 33rd Annual Conference of the Industrial Electronics Society, IECON' 2007; Taipei. 2007. p. 2841–5.
8. Salzman O, Shaharabani D, Agarwal PK, Halperin D. Sparsification of motion-planning roadmaps by edge contraction. International Journal of Robotics Research, 2014; 33(14):1711–25.
9. Ju T, Liu S, Yang J, Sun D. Rapidly exploring random tree algorithm-based path planning for robot-aided optical manipulation of biological cells. IEEE Transactions on Automation Science and Engineering. 2014; 11(3):649–57.
10. Yu C, Dai M, Chen X. Hierarchical robot path planning base on voronoi gragh, generalized level-set algorithm and local FMM algorithm. Journal of Computational Information Systems. 2012; 8(14):5765–83.
11. Stilman B. Network languages for complex systems. Int J Computers and Mathematics with Applications. 1993; 26(8):51–79.
12. Suresh KS, Vaithiyanathan V, Venugopal S. Emergence of meta heuristic algorithm principles in the field of robot motion planning-A survey of paradigm shift. International Journal of Applied Engineering Research. 2013; 8(20):2717–22.
13. Koubaa CI, Trigui A, Bennaceur S, Ammar H, Al-Shalfan A. Smart PATH: An efficient hybrid ACO-GA algorithm for solving the global path planning problem of mobile robots. International Journal of Advanced Robotic Systems. 2014; 11(1):1–15.
14. Kasilingam G, Pasupuleti J. Coordination of PSS and PID controller for power system stability enhancement-Overview. Indian Journal of Science and Technology. 2015; 8(2):142–51.
15. Jiang M, Fan X, Pei Z, Jiang J, Hu Y, Wang Q. Robot path planning method based on improved genetic algorithm, Sensors and Transducers. 2014; 166(3):255–60.
16. Liu F, Liang S, Xian X. Optimal robot path planning for multiple goals visiting based on tailored genetic algorithm. International Journal of Computational Intelligence Systems. 2014; 7(6):1109–22.
17. Azimirad V, Shorakaei H. Dual hierarchical genetic-optimal control: A new global optimal path planning method for robots. Journal of Manufacturing Systems. 2014; 33(1):139–48.
18. Suresh KS, Rajarajan S, Prabhu M, Vaithiyanathan V. Study of the influence of genetic algorithm parameters on travelling salesman problem. International Journal of Applied Engineering Research. 2013; 8(20):2667–74.