An Approach for Defining the Optimal Path for a Mobile Robot Navigating in an Area with Fixed Obstructions Using Genetic Algorithm Applying Travelling Salesman Problem

Suman Kumar Guha[†,a], Anannya Mukherjee[b], Soumak Roy[c]

[a]Accelerator Physics Group, Variable Energy Cyclotron Centre, Department Of Atomic Energy, Govt. Of India 1/AF Bidhannagar, Kolkata 700064.

[b]Department of Electronics and Telecommunication Engineering, Institute of Engineering and Management, Kolkata

[c]Department of Electronics and Telecommunication Engineering, Bengal College of Engineering and Technology, Durgapur.

Date of Submission: 3rd June, 2014 Date of Acceptance: 18th June, 2014

Abstract

Variable Energy Cyclotron Centre, Kolkata has positive ion accelerating facilities such as K-130 room Temperature Cyclotron, K-500 Super Conducting Cyclotron for various nuclear physics experiments. In order to reduce accidental radiation exposure the cyclotron vault, pit and ECR high bay areas are in accessible during the operation of the cyclotron due to high intensity of mixed radiation fields and neutron flux. This paper deals with an approach to define the unique least weighted path using TSP (applying GA) for a path tracking mobile robot within those in accessible areas with fixed obstructions through several intermediate nodes.

Key words: K-130 room Temperature Cyclotron, K-500 Super Conducting Cyclotron, (ECR) Electron Cyclotron Resonance Ion Source, Travelling Sales Man Problem, Genetic Algorithm.

[†] Corresponding author, Email: suman@vecc.gov.in

1. Introduction

Initially the job has been carried out for the ECR high bay area, which is near about 60 sq mt consisting of ECR beam line structure, Low Conductive Water lines, Electrical cable trays, control panel, distributed vacuum pumping modules scattered in different locations. The physical positions of these structures are fixed and hence considered as fixed obstacles. The main aim is to find out the optimal solution for a least weighted path of a path tracking mobile robot with near about 30 nodes.

2. Travelling Salesman Problem

Though there are rapid advancement of current technology in all areas, still there are some real world NP problems (non-deterministic method to generate possible solutions, deterministic method to verify in polynomial time that the solution is correct) like the travelling salesman problem[1], vehicle routing problem which are gaining interest of many researchers since the last few decades. The Travelling salesman problem (TSP) has been proved to be the base for comparison of different existing algorithm and the new algorithm . The TSP was formulated in 1930 and extensively studied for the optimization of the shortest possible path when a network with several nodes or cities with a starting node is given and the travelling cost between the nodes is known, provided each city should be exactly visited once and the salesman should return to the origin city (Refer figure 1). TSP can be solved easily but as the number of visiting cities increases the calculation rate increases geometrically and hence it is classified as a NP hard problem. TSP has several applications such as movement of the robotic arm in the production line , logistics, manufacturing of the PCBs and VLSI circuits ,DNA sequencing etc .

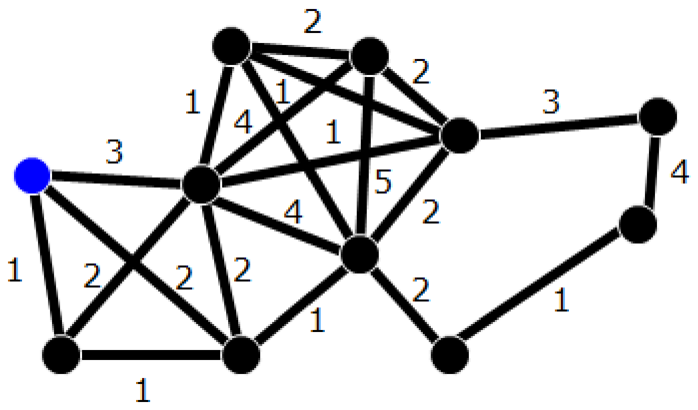

Figure 1. Typical city network with edge widths describing TSP.

Several techniques can be used to reduce the running time of the Travelling Salesman Problem such as Tabu search, heuristic search, Bio Inspired Algorithm (such as Ant colony optimization, Bee colony optimization algorithms) , Linear Programming and Genetic algorithms.

3. Genetic Algorithm (GA)

GA[2] was being proposed by Dr. J. H. Holland of Michigan State University in 1975 as a search and optimization algorithm. GA is based on the principle of natural selection process of evolution such as selection of best chromosome, crossover between the selected pairs and mutation of specific chromosomes. It is a parallel search procedure applicable to both continuous and discrete problem. GA approaches towards the optimization by encoding each probable solutions of a given problem into a binary bits strings which is known as chromosomes. Here the set of probable solutions is known as the initial population pool and each chromosome of this population pool carries a fitness value. In each iteration (in this case generations), GA generates a new population pool using the crossover and mutation operators and the chromosomes which are having higher fitness values will be survived for the next generation. After a several number of such iterations the convergence of the algorithm is being assured .The process is executed for a fixed number of iterations and the chromosome having the best fitness function which has obtained so far is considered as the solution. The process can be terminated when at least 50% of the total population carries almost same fitness value. Although there are some problems which do not guarantee any convergence issue after a number of iterations but in our case the program is executed for a fixed number of iterations and the chromosome having the best fitness function which has obtained so far is considered as the solution .

In this application the concept of genetic algorithm has been utilized to solve the travelling salesman problem.

4. GA Cycle

Begin

Generation of initial population randomly

Generation = 1

While (the point of convergence is not reached)

{

Fitness evolution of each chromosome in the population pool

Select parents for the mating pool by using roulette wheel selection process

Shuffle the mating pool

For each consecutive pair apply crossover with probability p_c , otherwise copy parents

For each offspring apply mutation (bit-flip with probability p_m independently for each bit)

Replace the longest tour within the population with the resulting offspring with better fitness

Generation = generation + 1

}

Final population is the output.

End

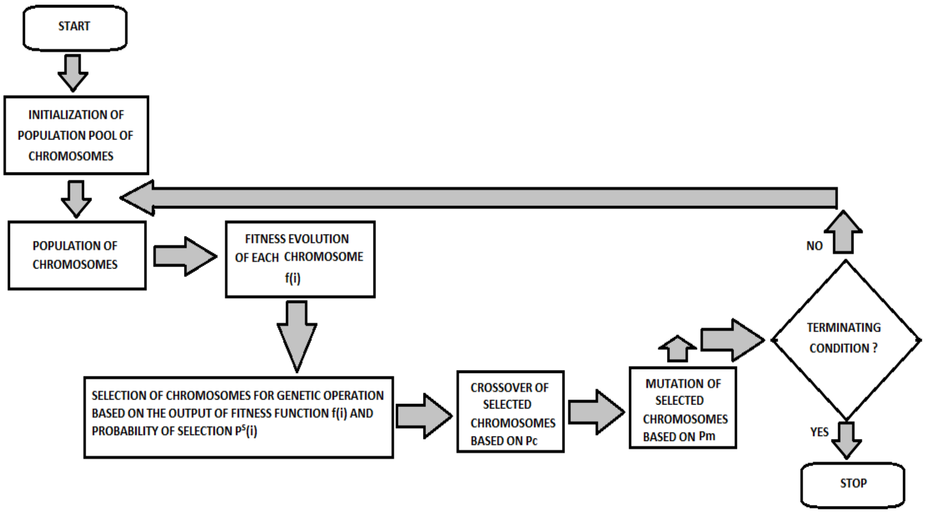

Figure 2. Basic block diagram of GA

4a. Generation of Initial Population

Here total 28 nodes (cities) have been selected within the specified zone as shown in the figure 3. A unique number [1-256] has been assigned to each nodes. Several tour paths starting from node 1 is randomly generated. Each tour path represents one probable solution or one chromosome.

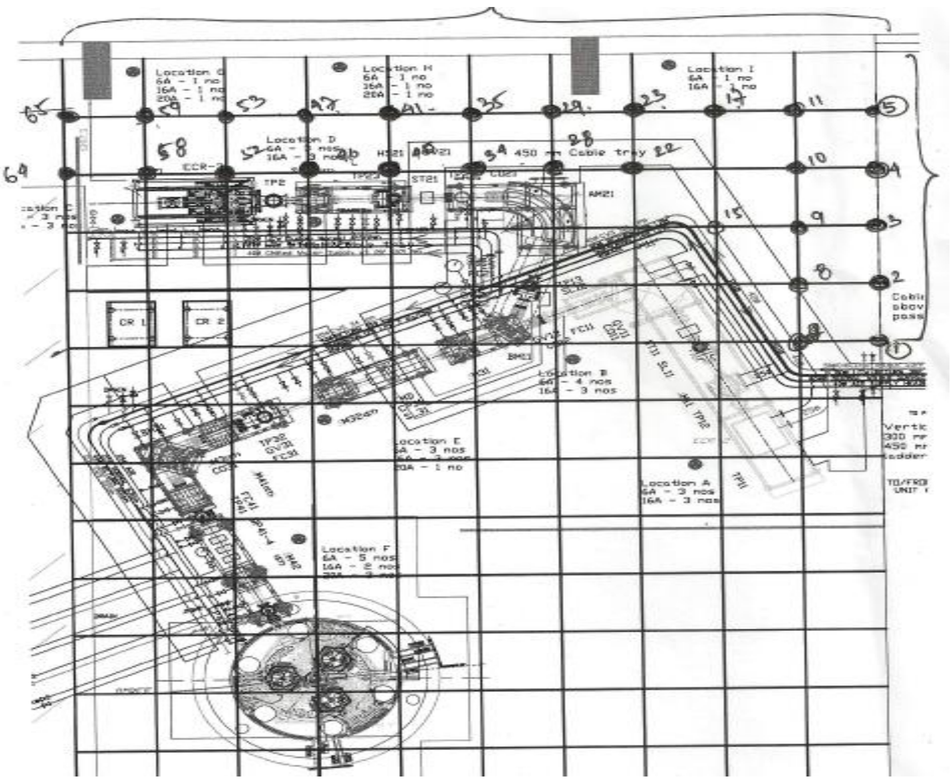

Figure 3. ECR Highbay area showing the Ion source, Beam line, LCW lines and Electrical cable tray which has been treated as obstructions and the 1m X 1m grid showing the positions of the nodes.

In our case the node identities (refer figure 3) are

1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 15, 16, 17, 22, 23, 28, 29, 34, 35, 40, 41, 46, 47, 52, 53, 58, 64, 65

The entire region is divided by a 1m x 1m grid structure and the crossing points which are not coinciding with any permanent physical obstruction not being considered as tour nodes. The distances between the nodes (edge width) have been noted in a tabular form. The distance between any two nodes where no traversal can take place because of physical obstruction have been considered as ∞.

The list of 20 chromosomes from the diagram is given below:-

1.(v1) = 1 2 4 6 ……… 41 65

2.(v2) = 1 3 7 9…………53 58

3.(v3) =

4.(v4) =

5.(v5) =

20.(v20) = 1 7 15 22 …….29 35

4b. Evolution of Fitness of Each Chromosome

Here the fitness of each chromosome has taken as the value of total edge weights. We now calculate the fitness function of above chromosome sequences $f(i)$.

4c. Selection

Fitness-Proportionate Selection or **Roulette Wheel Selection Method**:

Main idea: Better individuals get higher chance. Chances proportional to fitness

Implementation: roulette wheel technique

Assign to each individual a part of the roulette wheel

Spin the wheel n times to select n individuals (n population size).

Individual fitness of each chromosome *f(i)*,

Probability of selection of each chromosome $P_i^s = \frac{f(i)}{\sum_1^n f(i)}$

Cumulative probability $q(j) = q(j-1) + P_j^s$

In order to select a chromosome we now first generate n random numbers arbitrarily in the interval r [0,1] by spinning the **ROULETTE WHEEL SELECTION METHOD** n times.

Each time if q(i) < r then select v(i).

In this way the mating pool is generated and the number of chromosomes in the mating pool is m.

4d. Crossover

Ordinary crossover operator will create problem when being operated on valid mating pool. This problem will arise because there will be a great chance of generating offspring with some cities missing and some may be repeated. Refer figure 4.

P1	1	4	2	8	5	7	3	6	9
P2	7	5	3	1	9	8	6	4	2
C1	1	4	2	8	5	8	6	4	2
C2	7	5	3	1	9	7	3	6	9

Figure 4. Example of the ordinary cross over operator, where P1, P2 represent the parent and C1, C2 represent the offspring.

Although Disqualification or Repairing technique can be used to handle this problem, in our case we have used the special operator which is order- crossover[3]. Refer figure 5.

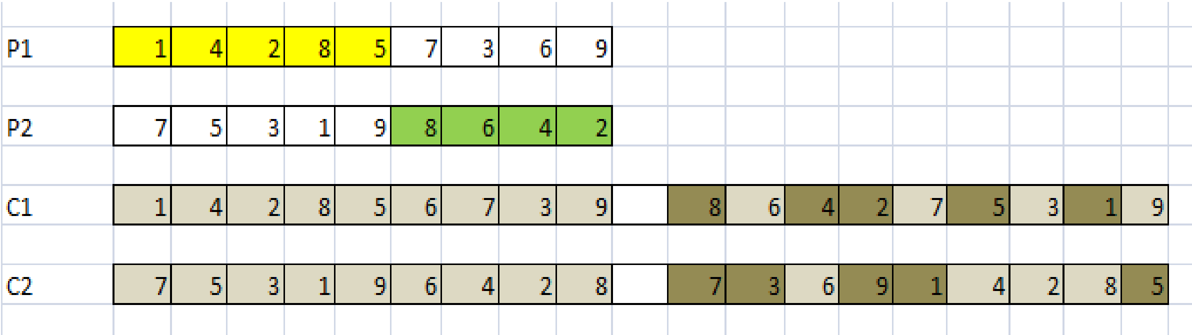

Figure 5. Example of the order cross over operator, where P1, P2 represent the parent and C1, C2 represent the offspring.

With a crossover probability (Pc) = 0.2, we generate the random numbers r [0,1] and allow a chromosome to crossover if the random number generated for Parent $_i$ is less than Pc. Let, the random no generated for the crossover feasibility testing of the chromosomes in order are

0.21890, 0.1527, 0.2690, 0.1890, 0.2690, 0.1422, 0.1628, 0.2567, 0.2678, 0.25855, 0.5830, 0.6532

So the chromosomes v2, v4, v6, v7 are now selected for participation in the crossover process. We have assigned the partners arbitrarily. Let us allow v2 and v4 to form a pair and v6 and v7 to form a second pair.

To identify the cross sites we again generate a random number in [0, n-1] where n denotes the word length of the chromosome string. If the random number generated is m, the m th bit is considered as the crossover point for the selected pair. The process is repeated for each pair of a selected strings .As an example suppose the crossover pair point of v2 and v4 is 7.

4e. Mutation

After crossover the chromosomes are subjected to undergo mutation process. Mutation is used to maintain a genetic diversity in the population pool. It helps the search algorithm to escape from converging in local minima. Among several mutation operators, we have used Twors Mutation operator (**Reciprocal exchange mutation)**[4] where exchange two node positions or index have done. Refer figure 6.

Let Pm=0.01,

We generate a random number in r[0,1], for each chromosome in the population pool (i=1 to n)

If, r<0.01, we select the chromosome for mutation. Thus for each chromosome we test the feasibility of the chromosome to be mutated.

To identify the first index position of mutation we generate a random number in r[0,n-1] where n is the word length of the chromosome .

To identify the second index position of mutation we generate another random number in r[0,n-1] where n is the word length of the chromosome .

After that the two nodes are exchanged.

Probable route	1	4	2	8	5	7	3	6	9
After mutation	1	4	3	8	5	7	2	6	9

Figure 6. Reciprocal exchange mutation with index 6 and 2 selected randomly for mutation.

4f. Next Generation Population Pool

We thus completed the first pass of the algorithm. We now redefine the above chromosomes by (v1) through (v20) and repeat the steps of genetic evolution for next generation. The algorithm terminates if the number of passes crosses a given threshold. The chromosome with best fitness value is then selected as the optimum path.

5. Conclusion and Future Work

The optimization technique adopted for finding out the least weighted path is done only for ECR Highbay Area. The same process has to be carried out for the Superconducting Vault area also. Application of other Crossover and Mutation Operators with different Pc and Pm has to be compared to find out optimum solution. Finally the permanent mark has to be put for the path tracking of mobile robot. The mobile robot development is also in progress which will collect several field information from the designated nodes falling within its path and will send it to a remote end.

References:

1. David L. Applegate, Robert E. Bixby, V. Chvátal, William J. Cook, *The Traveling Salesman Problem*: *A Computational Study*, Princeton University Press, 2007 Edition.

2. A. Konar, *Artificial Intelligence and Soft Computing*, CRC Press, Special Indian Edition 2008.

3. Zakir H. Ahmed, *International Journal of Biometrics and Bioinformatics,***3**, 96 (2010)

4. O. Abdoun, J. Abouchabaka, C. Tajani, *International Journal of Emerging Sciences,* **2**, 61(2012)