IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Path Designing of Known Complex Environment by Using Hybrid of GA & PSO

Sushma Bhatt[1], Mr. Anil Kumar Mishra[2]

M.Tech Scholar, Department of ET&T, Bhilai Institute of Technology, Durg, India [1]

Associate Professor, Department of ET&T, Bhilai Institute of Technology, Durg, India [2]

Abstract: A hybrid of Genetic Algorithm (GA) & PSO for associate optimized path designing for known atmosphere containing obstacles is conferred. The goal of our project is to seek out a best answer with best price & minimum distance to achieve destination. There are two sorts of atmosphere that are known and unknown atmosphere. Our project is based on known atmosphere. Earlier projects were supported single objective however when it involves real world,several parameters affects like temperature, humidity etc. We apply hybrid of GA with PSO. We have implemented the algorithms described in this paper and finally our approach can perform in various known environment with minimum energy and minimum distance. For the PSO parameter we have taken 500 of maximum iteration, 150 No of population (SWARM size), 1 inertia weight, 0.98 Inertia Weight Damping Ratio, c1 =1.5 Personal Learning Coefficient, c2=1.5 Global Learning Coefficient and with best cost we get result after the simulation in matlab using hybridization of GA & PSO.

Keywords: PSO, GA , Hybrid, Known Atmosphere, Single Objective, Damping Ratio, Global Learning Coefficient.

I. INTRODUCTION

For path coming up with there square measure varied soft computing technique for best-known and unknown complicated atmosphere. Path coming up with are often divided into 2 broad classes as world and native path coming up with. and also the path coming up with algorithms are often classified as single-solution algorithms and population-based algorithms. world path coming up with is that the one that needs a very best-known atmosphere and a static piece of ground. during this variety of path coming up with, the algorithmic rule develops a whole path from the supply purpose to the target purpose before the golem starts its motion. Basically path coming up with is that the term wont to describe the method of determinant a collision free path which means that a golem ought to arrange a reliable path between the supply and therefore the target while not colliding with the dynamic and static obstacles found in its atmosphere either complicated or unsure. This was drained real time by developing formula victimization numerous techniques. lots of approaches were projected for the design algorithms. Finding shortest path is that the problems with finding the best path is with minimum path length, with minimum reaching time and minimum management effort.

The major aim of those coming up with algorithms was to form the automaton optimize the shortest path. In most existing add path coming up with for obstacles dodging varied soft computing techniques has been used. In previous paper , Jean Bosco Mbede [1] given strong sturdy} Neuro-Fuzzy sensing element based mostly motion management among dynamic obstacles for automaton manipulators within which a brand new robust Neuro-Fuzzy controller for autonomous and intelligent automaton manipulators in dynamic and part far-famed environments containing moving obstacles is employed.

Now, Avneesh Sud, Erik writer, Sean botanist, Ming C.Lin [2] bestowed a time period path coming up with in dynamic virtual environments victimisation multiagent navigation graphs within which the MaNG is employed to perform route coming up with and proximity computations for every agent in real time. For Path coming up with and execution in quick dynamical environments with celebrated and unknown obstacles Thorsten Gecks and Dominik Henrich [3] determines a path planner capable of economical and time period handling of celebrated and unknown obstacles in extremely dynamic workspaces and generate a pseudo code to search out shortest distance. Allan R.Willlms and Simon X. Yang[4] presents period of time mechanism path coming up with via a distance-propagating dynamic system with obstacle clearance during which associate economical grid-based distance-propagating dynamic system is planned for period of time mechanism path coming up with in dynamic environments and compares with D* algorithmic rule.Durgesh Kumar Gupta &Anant Kumar Jaiswal [5] presents Path coming up with with real time obstacle shunning. during this associate integrated approach for each native furthermore as international path coming up with of a mechanism has been planned. the first algorithmic rule that has been used for path coming up with is that the artificial Potential field approach and A* search algorithmic rule has been used for locating the foremost optimum path for the mechanism.

Pranab K Dan [6] presents Obstacle turning away and travel path determination in Facility location designing within which analysis finds an answer methodology for deciding best travel path to and from existing facilities and corresponding location of a replacement facility having physical flow interaction between them in numerous degrees translated into associated weights, in presence of barriers preventative the shortest flow-path involving straight-line distance metric.

II. PROBLEM DEFINITION

A. Single objective verses multiobjective:

Earlier comes square measure single objective however once we come back to world there's uncountable parameter to contemplate. like temperature, humidness etc.

- Many real-world deciding issues have to be compelled to accomplish many objectives: minimize risks, maximize dependability, minimize deviations from desired levels, minimize price, etc [7] .
- The main goal of single-objective (SO) optimisation is to seek out the "best" resolution, that corresponds to the minimum or most worth of one objective perform that lumps all totally different objectives into one.

This type of improvement is helpful as a tool that got to provide decision makers with insights into the character of the matter, but generally cannot provides a cluster whole totally different solutions.On the other, in associate degree extremely multi objective improvement with conflicting objectives, there is not any single best answer. The interaction among utterly totally different objectives offers rise to a bunch of compromised solutions, largely referred to as the trade-off, non dominated, non inferior or Pareto-optimal solutions.

B. Known environment verses unknown environment:

Known environment being a "known deterministic or probabilistic environment" – which means that we may have an environment that is probabilistic, i.e. we can't say with certainty exactly what will happen, but we at least know the probabilities of various options at various junctures. For that reason, we suggested that all games were "known environments" other than "Hide-and-go-seek" and "Robot Soccer", based on the ideas that:

1) In poker, for instance, you don't know what card we'll get dealt next, or what card an opponent holds, but we can say with what probability they hold a given card, based on whatever knowledge we have been collecting (whether that's none, or some by, for instance, counting cards);

2) In robot soccer, for instance, we cannot say in advance with what probability the opposing team will apply certain tactics.

C. An Optimization Problem:

The main components of an optimization problem are:

Objective Function:-

An objective operate expresses one or additional quantities that are to be reduced or maximized. The optimisation issues could have one objective operate or additional objective functions. sometimes the various objectives don't seem to be compatible. The variables that optimize one objective could also be removed from best for the others. the matter with multi-objectives may be reformulated as single objective issues by either forming a weighted combination of the various objectives or by treating a number of the objectives as constraints.

Variables:-

A set of unknowns, which are essential are called variables. The variables are used to define the objective function and constraints. One cannot choose design variable arbitrarily, they have to satisfy certain specified functional and other requirements. The design variables can be continuous, discrete or Boolean.

Constraints:-

A set of constraints are those which allow the unknowns to take on certain values but exclude others. They are conditions that must be satisfied to render the design to be feasible. Once the design variables, constraints, objectives and the relationship between them have been chosen, the optimization problem can be defined.

Statement of an optimization problem:-

An optimization problem can be stated as follows: To find $x = [x_1\, x_2 \,........ x_n\,]^T$, which minimizes or maximizes f(x) ; Subject to the constraints

$G_i(x) \leq 0$; i = 1, 2, 3......m

$H_j(x) = 0$; j = 1,2,3.........p

Where x is an n-dimensional vector called design variable, f(x) is called the objective function, $G_i(x)$ and $H_i(x)$ are known as inequality and equality constraints respectively. This type of problem is called constrained optimization problem.

D. Classification of Optimization Problems:

Optimization problems can be classified based on the type of constraints, nature of design variables, nature of the equations involved and type & number of objective functions. This classifications are briefly discussed below.

Based on existence of constraints:-

A problem is called constrained optimization problem if it is subject to one or more constraints otherwise it is called unconstrained.

Based on the nature of the equations involved:-

Based on the nature of equations for the objective function and the constraints, optimization problems can be classified as linear and nonlinear programming problems. The classification is very useful from acomputational point of view since many predefined special methods are available for effective solution of a particular type of problem.

1. Linear Programming problem

If the target operate and every one the constraints square measure 'linear' functions of the planning variables, the optimisation downside is named a applied math downside (LPP).

2. Quadratic programming problem

If the objective function is a quadratic function and all constraint functions are linear functions of optimization variables, the problem is called a quadratic programming problem. It is possible to solve Quadratic Programming problems using extensions of the methods for LPP.

3. Nonlinear programming problem

If any of the functions among the objectives and constraint functions is nonlinear, the problem is called a nonlinear programming (NLP) problem. This is the most general form of a programming problem and all other problems can be considered as special cases of the NLP problem.

Based on the permissible values of the decision variables:-

1. Integer programming problem
If some or all of the design variables of an optimization problem are restricted to take only integer (or discrete) values, the problem is called an integer programming problem. Other restrictions on minimum and maximum number of usable resources may be imposed.

2. Real-valued programming problem
A real-valued problem is that in which it is sought to minimize or maximize a real function by systematically choosing the values of real variables from within an allowed set. When the allowed set contains only real values, it is called a real-valued programming problem.

Based on the number of objective functions:-
Under this classification, objective functions can be classified as single-objective and multi-objective programming problems.

1. Single-objective problem:
Problem in which there is only a single objective function.

2. Multi-objective problem:
A multi-objective programming problem can be stated as follows:
Find x which minimizes $f_1(x)$, $f_2(x)$..........$f_k(x)$;
Subject to $g_j(x) \leq 0$; j = 1,2,3......m
Where f_1 , f_2 ....f_k denote the objective functions to be minimized simultaneously. There are m number of constraints also. For multi-objective optimization problems one tries to find good trade-offs rather than a single solution as in single objective problems.

III. METHODOLOGY

In this paper we use of Hybrid of GA & PSO for path planning and for getting best solution to reach from source to destination and also use of Euclidian distance formula for minima distance.

Euclidian distance:
In mathematics, the Euclidean distance or Euclidean metric is the "ordinary" (i.e. straight line) distance between two points in Euclidean space. With this distance, Euclidean space becomes a metric space . The Euclidean distance between point's p and q is the length of the line segment connecting them (pq). In Cartesian coordinates, if p = (p1, p2,..., pn) and q = (q1, q2,..., qn) are two points in Euclidean n space, then the distance (d) from p to q, or from q to p is given by the **Pythagorean formula:**

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2+(q_2-p_2)^2+\cdots+(q_n-p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}. \quad (1)$$

The position of a point in a Euclidean n space is a Euclidean vector.
So, p and q are Euclidean vectors, starting from the origin of the space, and their tips indicate two points.

Now optimization technique to optimize the path to reach from source to destination.
Optimization means maximization or minimization of one or more functions with any possible constraints. In this chapter different types of optimization techniques are described briefly with emphasis on those that are used in the present dissertation.

A. Optimization using GA:
In this work, a GA technique is used due to itsgenerality and capability to heuristically overcomesituations where an exhaustive solution would be toocomputationally demanding. The main goal is to optimize and find a best solution for path planning.

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).

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.

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.

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;
- Manual inspection.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results;

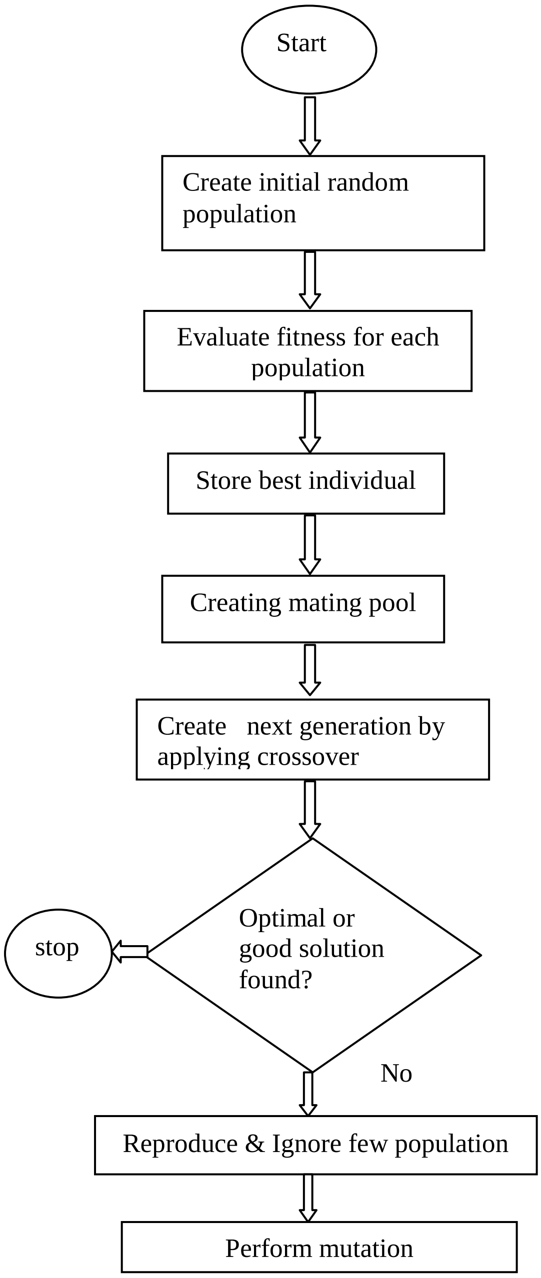

Figure No.1 Flowchart of GA

B. Optimization using PSO:

PSO could be a Swarm intelligence meta-heuristic impressed by the cluster behaviour of animals, as an example bird flocks or fish faculties. equally to genetic algorithms (GAs), a population-based methodology, that is, it represents the state of the rule by a population, that is iteratively changed till a termination criterion is glad.

In PSO algorithms, the population $P=\{P_1,\ldots,P_n\}$ of the feasible solutions is often called a **swarm**.

- The feasible solutions $P_1,\ldots,P_n$ are called **particles**.
- The PSO method views the set R^n of feasible solutions as a "space" where the particles "move".

Basic Variants of PSO

The lacks of PSO are reduced with a variation of PSO. several variations are developed to enhance speed of convergence and quality of answer found by the PSO.

The variation is influenced by variety of management parameters, particularly the dimension of the matter, the quantity of particles (swarm size), acceleration constants (The acceleration coefficient, c1 & c2 in conjunction with random vector r1 and r2, management the random influence), inertia weight, neighbourhood size, variety of iteration, and also the random values that scale the contribution of the psychological feature and socialcomponent. Below area unit the essential variations of particle swarm optimization:

A. Velocity clamping:

Velocity clamping can management the world exploration of the particle. If the speed v of a particle i exceeds the utmost allowed regulation, it'll set a most worth of rate (vmax(j)). so , v_{max}, j indicates the utmost allowable speed for a particle within the jth dimension. Speed (velocity) of the particle is adjusted exploitation the equation :

$$v_{ij} = \begin{cases} v'_{ij}(t+1), if\ v_{ij}(t+1) < v_{max}(J) \\ v_{max}(j)\ \ otherwise \end{cases} \tag{2}$$

High value of $v_{max}(j)$ will cause global exploration, whereas lower values result in local exploration. $v_{max}(j)$ will control the movement of the particle and aspect of exploration and exploitation. The following equation is used to initialize the max and min velocity to the solution:

$$\begin{aligned} v_{max,j} &= \delta\ (x_{max,j} - x_{min,j}) \\ v_{min,j} &= \delta\ (x_{min,j} - x_{max,j}) \end{aligned} \tag{3}$$

Where as $x_{max,j}$ and $x_{min,j}$ are the minimum and maximum positions of the particle in the j^{th} dimension. δ is a constant factor and is taken from 0 until 1. The problem is if all the velocity becomes equal to v_{max} the particle will continue to conduct searches within a hypercube and will probably remain in the optima but will not converge in the local area.

B. Inertia weight:

It is a mechanism to manage a probe and exploitation talents of the swarm, and as mechanism to eliminate the necessity of rate clamping. The inertia weight, controls the momentum of the particle by advisement the contribution of the previous rate essentially dominant what quantity memory of the previous flight direction can influence the new rate. For the gbest PSO, the rate equation changes from equation.

$$V_{ij}(t+1) = wv_{ij}(t) + c_1r_{1j}(t)\{y_{ij}(t) - x_{ij}(t)\} + c_2r_{2j}(t)\{y_j(t) - x_{ij}(t)\} \tag{4}$$

A similar change is made from the-ibest PSO. Inertia weight presenting how much the amount of memory from the previous flight direction will affect the new velocity.

- If w > 1, then the velocity will decrease with time, the particle will accelerate to maximum velocity the swarm will be divergent.
- If w < 1, then the velocity of particle will decrease until it reaches zero.

DOI 10.17148/IJARCCE.2016.5356

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

The larger value of will facilitate an exploration, rather small values will promote the exploitation.

C. Constriction Coefficient:
Velocity update equation that using constriction coefficient changes to:

$$V_{ij}(t+1) = x[V_{ij}(t) + \Phi_1\{y_{ij}(t) - x_{ij}(t)\} + \Phi_2\{y_j(t) - x_{ij}(t)\}] \quad (5)$$

Where,

$$x = \frac{2k}{|2-\Phi-\sqrt{\Phi(\Phi-4)}|} \quad (6)$$

With

$$\Phi = \Phi_1 + \Phi_2$$
$$\Phi_1 = c_1 r_1$$
$$\Phi_2 = c_2 r_2 \quad (7)$$

- Equation above is used under the constraints that $\Phi \geq 4$ and k € [0,1].
- The constriction approach was developed as a natural, dynamic way to ensure convergence to a stable point, without the need for velocity clamping.
- Condition $\Phi \geq 4$ and k € [0,1] and of the swarm is guaranteed to convergence

D. Synchronous Versus Asynchronous Updates:
Flow chart for PSO as shown below:

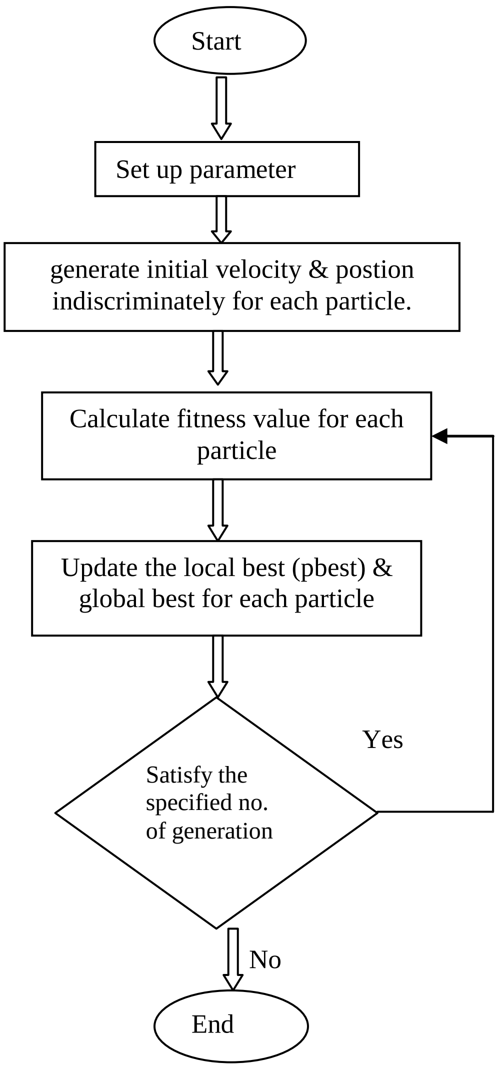

Fig 2 flow chart of PSO

Synchronous Updates area unit done severally from the particle (personal best and neighbourhood bests) position updates, solely given one feedback per iteration update, slower feedback and higher for gbest. whereas asynchronous is best for, updates calculate the new best positions when every particle position update and have the advantage that immediate feedback is given regarding the most effective region of search house.

IV. RESULTS

We apply GA algorithm for optimization of path planning in which we have a known environment and for this we take of population size 50 , No of generation 10 & No of position solution 3. we get a graph as shown in figure 3 between generation-fitness & generation & average distance between individual.from this we can see that best fitness is 938 and mean fitness is 941926 and we compare of best, worst & mean scores.after this we get path from source to destination as shown in fig 4.

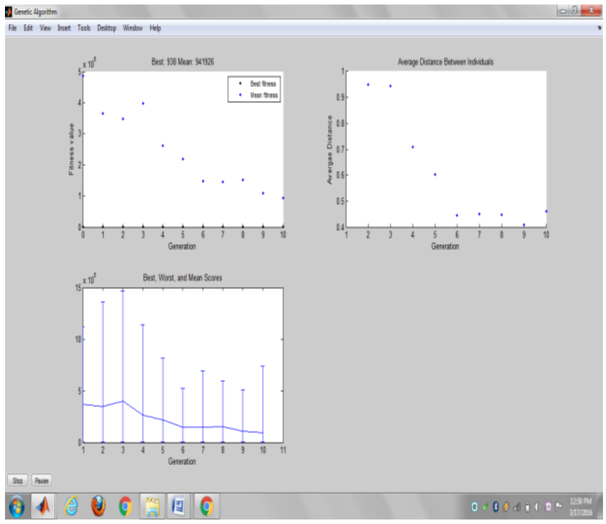

Fig No 3 Graph of Best, Worst & Mean Score

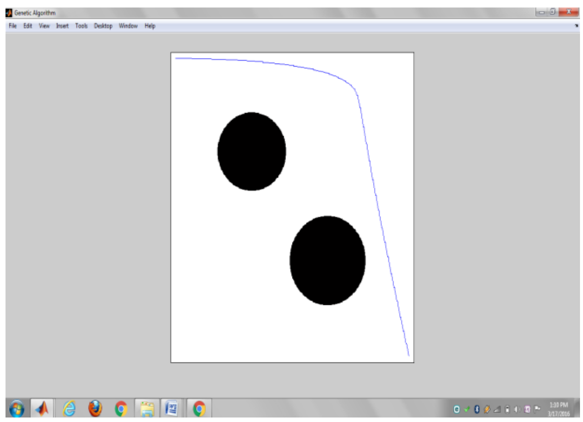

Fig No 4 Applying GA algorithm in Known Evironment

When we apply PSO algorithm in path planning it is little bit easier to reach from source to destination as compare to GA algorithm.in this PSO algorithm we have taken 500 Maximum Number of Iterations, 150 no of Population i.e. Population Size (Swarm Size), 1 Inertia Weight,0.98

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Inertia Weight Damping Ratio,1.5 Personal Learning Coefficient & 1.5 Global Learning Coefficient.we get as shown in fig no 5 & graph between iteration and best cost in fig no 6.

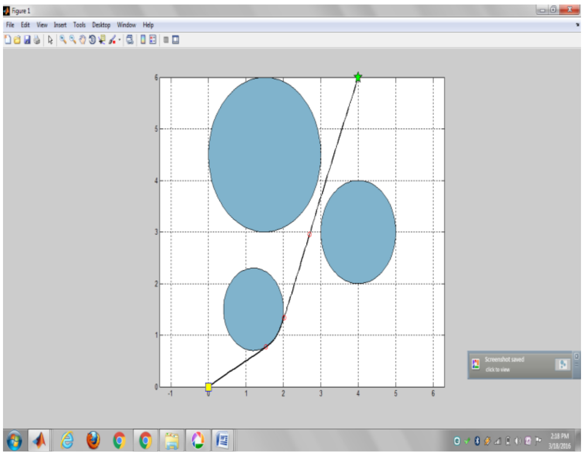

Fig No 5: Applying PSO Algorithm

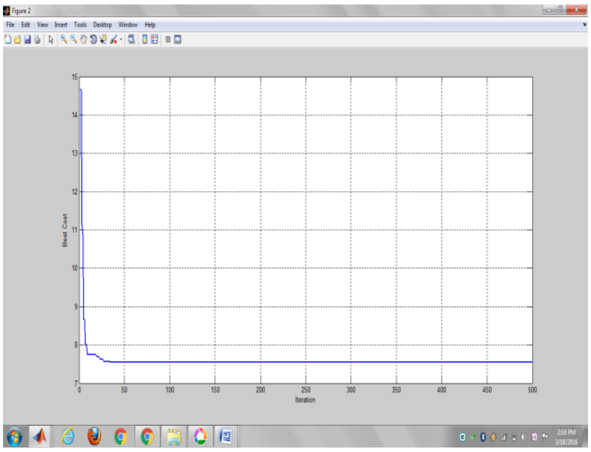

Fig No 6 : Result Of Pathplanning by Using PSO

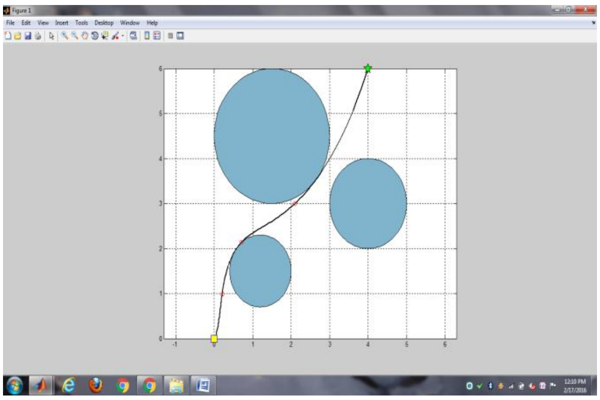

Fig No 5: Hybridization of GA &PSO In Known Environment

Now, we apply hybrid of GA with PSO. We have implemented the algorithms described in this paper and finally our approach can perform in various known environment with minimum energy and minimum distance. For the PSO parameter we have taken 500 of maximum iteration, 150 No of population(SWARM size), 1 inertia weight, 0.98 Inertia Weight Damping Ratio, c1 =1.5 Personal Learning Coefficient, c2=1.5 Global Learning Coefficient and with best cost we get result after the simulation in matlab using hybridization of GA + PSO, we use of no of iteration to optimize the path and we get as shown below:

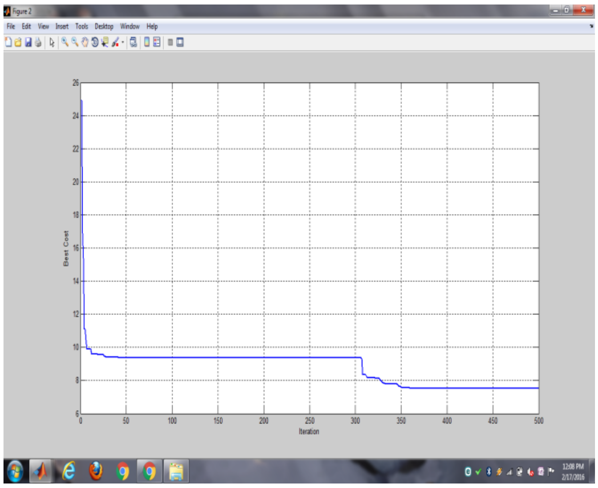

Fig No 6: Result of PathPlanning by Using Hybridization

V. CONCLUSION

We presented a path planning approach for a category of path planning problems with known complex environment for optimizing the path to reach from source to destination we use of hybridization. No of iteration has been used for the value of best cost. GA and PSO algorithm for robot path planning in known complex environment has been extended to the situation where the optimal paths are not simply the shortest path from a robot to a target but also paths that minimizes a cost function based on both distance to a target and proximity to obstacles.

REFERENCES

[1] Jean Bosco Mbede, Xinhan Huang, and Min Wang, "Robust Neuro-Fuzzy sensors-based motion control among dynamic obstacles for robot manipulators," IEEE transactions on fuzzy systems, vol. 11, no. 2, April 2003.

[2] Avneesh Sud, Erik Andersen, Sean Curties, Ming C.Lin,, and Dinesh Manocha, " Real-Time Path Planning in Dynamic Virtual Environments Using Multiagent Navigation Graphs," IEEE Transactions on Visualization and Computer Graphics, Vol. 14, No. 3, May/June 2008.

[3] Thorsten Gecks and Dominik, "Path Planning and Execution in Fast-Changing Environments with Known and Unknown Obstacles,"International Conference on Intelligent Robots and Systems in San Diego, USA, 29 October – 2 November 2007

[4] Allan R. Willms and Simon X. Yang, "Real-Time Robot Path Planning via a Distance-Propagating Dynamic System with Obstacle Clearance," IEEE Transactions On Systems, Man, and Cybernetics—Part B: Cybernetics, Vol. 38, No. 3, June 2008.

[5] Durgesh Kumar Gupta,Anant Kumar Jaiswal, "Path Planning with Real Time Obstacle Avoidance,"International Journal of Computer Applications (0975 – 8887) Volume 70– No.5, May 2013.

[6] Pranab K, "Obstacle Avoidance and Travel Path Determination in Facility Location Planning," JJMIE Volume 3, Number 1, ISSN 1995-6665 Pages 37- 46 March. 2009

[7] Dragan Savic, "Single-objective vs. Multiobjective Optimisation for Integrated,"Centre for Water Systems, Department of Engineering School of Engineering and Computer Science, University of Exeter, United Kingdom, E-mail: D.Savic@ex.ac.uk