International Journal For Technological Research In Engineering
Volume 2, Issue 7, March-2015 *ISSN (Online): 2347 - 4718*

COMPLETE REVIEW ON ARTIFICIAL NAVIGATION SYSTEM USING A-STAR ALGORITHM

Sajjan Chahl[1], Mr. Parvinder Bangar[2]
[2]Guided, [1,2]CBS Group of Institutions, Jhajjar

ABSTRACT: To find the shortest distance always suffers struggle and more complexity rather than simple and easy path. Here we are finding the position & shortest path to an object from image taken from camera or any visual sensing device by making use of fuzzy logic & Artificial Intelligence. First we take image of the desired area then take out the desired place where we want to reach as quick and by proper way. By Artificial Intelligence we can find the best way out of all possible good ways to reach. Shortest path not always depends upon the shortest path it may lead the rush, large number of traffic signals, worst condition of road which may cause for delay. So we take all into account to make our modify model for shortest path. Here our model provides the user with more than one way/path to go to the destination.

Key Word: Shortest Path, AI, Fuzzy Logic, Grid Algorithm

I. INTRODUCTION

Position tracking system is to find the shortest path by means to save time. Earlier model give the shortest path on the basis of shortest distance, it does not include the time factor where our model include the factors that affect the time to reach the destination earlier than the previous shortest path. Here, we are making use of AI & Fuzzy-Logic to track the position to the object & then finding the shortest path from source to destination. Shortest path is the best path to reach from source to destination & it will make use of AI that helps in decision making power.

A. FUZZY LOGIC:

Fuzzy logic expert systems often improve performance by allowing knowledge to generalize without requiring the knowledge engineer to anticipate all possible situations. Thus, for many types of applications, “soft computing” such as Fuzzy logic can incur lower overhead in terms of representing and engineering task knowledge. My project uses the application of fuzzy expert systems to position tracking.

B. ARTIFICIAL INTELLIGENCE :

Artificial intelligence (AI) is the intelligence exhibited by machines or software, and the branch of computer science that develops machines and software with human-like intelligence. The use of AI research includes reasoning, knowledge, planning, learning, communication, perception and the ability to move and manipulate objects. General intelligence is still among the field's long term goals. There are an enormous number of tools used in AI, including versions of search and mathematical optimization, logic, methods based on probability and economics, and many others.

II. BACKGROUND

Previous research showed that there are various other algorithms being used to find the shortest distance such as Bellman Ford in 1957 gave the Bellman Ford algorithm that solves the single source problem if edges weights may be negative. Dijkstra’s in 1959 gave the Dijkstra’s algorithm that solves the single source shortest path problems. Roy–Warshall and Robert Floyd in 1962 gave the Floyd-Warshall algorithms that solve all pair shortest path. Andrew Viterbi in 1967 gave the Viterbi algorithm that solves the shortest stochastic problem with an additional probabilistic weight on each node. Peter Hart , Nils Nilssons and Bertram Raphael in 1968 gave the A* search algorithm that solves for single pair shortest path using heuristic to try to speed up the search. Donald B. Johnson in 1977 gave the Johnson’s algorithm that solves all pair shortest path and may be faster than Floyd-Warshall on sparse graphs. Dijkstra’s algorithm is one of the best algorithms that is being used. All the above previous researches discussed on the shortest path algorithms are purely based on the distance factor. Previous research showed that Fuzzy Logic can be used to track the motion of a brightly colored object against a dark background, with relatively low development and run-time costs.

III. CORE CONCEPT

Position Tracking System to find shortest path to object focuses to track the position to object and then find the shortest path to objects from source to destination where, the shortest path problem is the problem of finding the best path between two objects.

We are using grid value algorithms which is as follow :
Inputs: (#) Size of the NxN Grid. (N); (#) No. of Paths: Z; (#) Source and Destination Coordinates of each of the individual Z paths.
Given: (#) Each path comprises of the cells which are ADJACENT to each other and NOT diagonal. (#) Time taken to cross each cell is EQUAL and is an unit time. (#) Travelling starts from the respective source coordinates of each path in the grid SIMULTANEOUSLY. (#) Paths may intersect, BUT after say 'x'th unit time no two paths must intersect in one particular cell at a time. If such case arises, then one of the paths need to bypass into a new path taking alternate adjacent cell just before the clashing cell.

Output: Z no. of paths which are SHORTEST possible without clash.
[Say there are two paths (Z=2), say X to Y and W to Z. While travelling, both X and W start simultaneously from their respective source coordinates, going to their respective destinations Y and Z such that they should take the SHORTEST possible path, AND after 'x'th unit time, they must not CLASH (when both the paths have traveled exactly (x-1) no. of cells and clashing into the 'x'th cell) into one single cell. If that happens, one of the path needs to be bypassed towards its destination keeping in mind without clash, shortest possible path. And this has to be implemented for Z no. of lines starting simultaneously in the grid.For position tracking we provide Grid value of both source and destination and finding it coordinate and match its referential points. We are adding various factors such as jamming, damage of road ,no. of traffic signals and highway dimension in real time to enhance our above algorithm so that the user can find the destination in the short time.

IV. PROPOSED WORK

To find the shortest distance always suffers struggle and more complexity rather than simple and easy path. Here we are finding the position & shortest path to an object from image taken from camera or any visual sensing device by making use of fuzzy logic & Artificial Intelligence [8]. First we take image of the desired area then take out the desired place where we want to reach as quick and by proper way [6]. By Artificial Intelligence we can find the best way out of all possible good ways to reach. Shortest path not always depends upon the shortest path it may lead the rush, large number of traffic signals, worst condition of road which may cause for delay [7]. So we take all into account to make our modify model for shortest path. Here our model provides the user with more than one way/path to go to the destination.

A. REAL TIME INFORMATION MODEL FOR SHORTEST PATH

- Real Time information model with grid algorithm is the proposed Algorithm for shortest path problem.
- In this algorithm, we provide real time information to shortest path algorithm.
- In this we include various factors like jamming, damage of roads, no. of traffic signals, rush which affects the shortest path in time domain

B. A Algorithm for Real time system*

Construct grid, where a * = obstacle and you can move up, down, left and right, and you start from S and must go to D, and 0 = free position:

```
S 0 0 0
* * 0 *
* 0 0 *
0 0 * *
* 0 0 D
```

You put S in your queue, then "expand" it:

```
S 1 0 0
* * 0 *
* 0 0 *
0 0 * *
* 0 0 D
```

Then expand all of its neighbours:

```
S 1 2 0
* * 0 *
* 0 0 *
0 0 * *
* 0 0 D
```

And all of those neighbours' neighbours:

```
S 1 2 3
* * 3 *
* 0 0 *
0 0 * *
* 0 0 D
```

And so on, in the end you'll get:

```
S 1 2 3
* * 3 *
* 5 4 *
7 6 * *
* 7 8 9
```

So the distance from S to D is 9. The running time is O(NM), where N = number of lines and M = number of columns. I think this is the easiest algorithm to implement on grids, and it's also very efficient in practice. It should be faster than a classical dijkstra.. The following program calculates the minimum point of a multi-variable function using the grid search method. This method performs a multi-dimensional grid search. The grid is defined by a multiple dimensions. Each dimension has a range of values. Each range is divided into a set of equal-value intervals. The multi-dimensional grid has a centroid which locates the optimum point. The search involves multiple passes. In each pass, the method local a node (point of intersection) with the least function value. This node becomes the new centroid and builds a smaller grid around it. Successive passes end up shrinking the multidimensional grid around the optimum.

The function Grid_Search has the following input parameters:

N - Number of variables
XLo - array of lower values
XHi - array of higher values
NumDiv - array of number of divisions for each range
MinDeltaX - array of minimum ranges
Eps_Fx - tolerance for difference in successive function values
MaxIter - maximum number of iterations
myFx - name of the optimized function
The function generates the following output:
X - array of optimized variables
BestF - Function value at optimum
Iters - number of iterations
Here is a sample session to find the optimum for the following function:

y = 10 + (X(1) - 2)^2 + (X(2) + 5)^2

The above function resides in file fx1.m. The search for the optimum 2 variables has the search range of [-10 -10] and [10 10] with a divisions vector of [4 5] and a minimum range vector of [1e-5 1e-5].
The search employs a maximum of 10000 iterations and a function tolerance of 1e->> [XBest,BestF,Iters]= Grid_Search(2, [-10 -10], [10 10], [4 4], [1e-5 1e-5], 1e-7, 10000, 'fx1')
XBest =2.0001 -5.0000
BestF =10.0000
Iters = 200
Notice how close the located optimum is to the actual one [-2 5].
Here is the MATLAB listing:

```
Function y=fx1(X, N)
y = 10 + (X(1) - 2)^2 + (X(2) + 5)^2;
end
function [XBest,BestF,Iters]=Grid_Search(N, XLo, XHi, NumDiv, MinDeltaX, Eps_Fx, MaxIter, myFx)
% Function performs multivariate optimization using the
% grid search.
% Input
% N - number of variables
% XLo - array of lower values
% XHi - array of higher values
% NumDiv - array of number of divisions along each dimension
% MinDeltaX - array of minimum search values for each variable
% Eps_Fx - tolerance for difference in successive function values
% MaxIter - maximum number of iterations
% myFx - name of the optimized function
% Output
% XBest - array of optimized variables
% BestF - function value at optimum
% Iters - number of iterations
Xcenter = (XHi + XLo) / 2;
XBest = Xcenter;
DeltaX = (XHi - XLo) ./ NumDiv;
BestF = feval(myFx, XBest, N);;
if BestF >= 0
  LastBestF = BestF + 100;
else
  LastBestF = 100 - BestF;
end
X = XLo; % initial search value
Iters = 0;
bGoOn = 1;
while (bGoOn > 0) && (abs(BestF - LastBestF) > Eps_Fx) && (Iters <= MaxIter)
bGoOn2 = 1;
while bGoOn2 > 0
Iters = Iters + 1;
F = feval(myFx, X, N);
  if F < BestF
    LastBestF = BestF;
    BestF = F;
    XBest = X;
  End
```

The next For loop implements a programming trick that simulated nested loops using just one For loop search next grid node

```
  for i = 1:N
    if X(i) >= XHi(i)
      if i < N
        X(i) = XLo(i);
      else
        bGoOn2 = 0;
        break
      end
    else
      X(i) = X(i) + DeltaX(i);
      break
    end
  end
 end
 while bGoOn2 > 0
 XCenter = XBest;
 DeltaX = DeltaX ./ NumDiv;
 XLo = XCenter - DeltaX .* NumDiv / 2;
 XHi = XCenter + DeltaX .* NumDiv / 2;
 X = XLo; % set initial X
 bGoOn = 0;
 for i=1:N
   if DeltaX(i) > MinDeltaX(i)
     bGoOn = 1;
   end
 end
end
while bGoOn > 0 && () && ()
```

C. MATHEMATICAL ANALYSIS OF PROPOSED ALGORITHM

Integrate the real time model to the Grid algorithm
Inputs: Size of the Grid (N); No. of Paths: Z; Source and Destination Coordinates of each of the individual Z paths
Given: Each path comprises of the cells which are ADJACENT to each other and NOT diagonal. Time taken to cross each cell is EQUAL and is an unit time.
Output: Z no. of paths which are SHORTEST possible without clash.
Z(t) is time domain for AI
$\acute{Z}(t) = \text{Mean}(z)$;
$\acute{Z}(t+\delta t) = \acute{Z}(t) + R^{-1}(Z) * \blacktriangledown_z * z(t+\delta t)$

D. HOW SHORTEST PATH CALCULATION PROCESSED BY REAL TIME INFORMATION MODEL

- Shortest path is being calculated by Grid Algorithm.
- It also gives the position of origin and destination.
- After that we include the factors that affect the shortest path.
- We get the shortest path in real time information model that is more suitable for user.

V. ADVANTAGES OF NEW ALGORITHM

New algorithm including every factor that we face in our daily life. It give user more ways to choose and provide shortest path in term of very less time as compare to previous model.

VI. CONCLUSION

This inclusion give us more dynamic model and these real time situation provides it a best model for future. This enhancement brings us more closer to real situations. So real life implication of this model give us more real parameter to the world and real time scenario for path travelling. This save tim e for travelers.

REFERENCES

[1] Fuzzy Air-Fuel Ratio Control by S.H.Lee, R.J.Howlett & S.D.Walters 2004
[2] Fuzzy condensed algorithm applied to control a robotic head for visual tracking by E V Cuevas, D Zalfidar and R Rojas in 2004
[3] Using Fuzzy-Logic with radio frequency, Shruthi Reddy Sarikonda , August 2010
[4] Dense, Robust, and Accurate Motion Field Estimation,Clemens Rabe , Thomas M¨uller, Andreas Wedel, and Uwe Franke Daimler Research, Sindelfingen 2012.
[5] Interpretation of Complex Situations in a Semantic-Based Surveillance Framework, Carles Fern´andez a, Pau Baiget a, Xavier Roca a, Jordi Gonz`alez in 2008
[6] Artificial Intelligence Approach of Optimal Route Selection in Telematics. 2008
[7] Fuzzy Logic Based-Map Matching Algorithm S Syed and M E Cannon , 2004
[8] Image Acquisition Toolbox Product Description 2014
[9] earning and Application for Time-Series Prediction Peter Y.C Hwang , Qun Song & Nikola Kasabov, 2002
[10] Fuzzy interpolation reasoning via scale And move transformations Z H Huang & Q.Shen , 2006
[11] "Shortest paths in a weighted digraph" Edger Dijkstra
[12] "Shortest paths on planner network in time from K.Mahlhorn.
[13] Dijkstra's Shortest Path Algorithm.
[14] Highway Dimension, Shortest Paths, and Provably Efficient Algorithms by Ittai Abraham , Amos Fiat
[15] Shortest Path Algorithm lecturer notes of mit.edu/opensource
[16] Experimental Evaluation of a new Shortest Path Algorithm Seth Pettie, Vijaya Ramachandran and Srinath Sridhar
[17] Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems , Gaurong chen,Trung Tat
[18] Position tracking and motion prediction using Fuzzy Logic , Patrick Rodjito
[19] Probabilistic Path Queries in Road Networks , Ming Hua, Jian Pei

