International Journal of Electrical and Computer Engineering (IJECE)
Vol. 6, No. 1, February 2016, pp. 212~222
ISSN: 2088-8708, DOI: 10.11591/ijece.v6i1.8606

Path Planning Based on Fuzzy Decision Trees and Potential Field

Iswanto*, **, Oyas Wahyunggoro*, Adha Imam Cahyadi*
* Department of Electrical Engineering and Information Technology, Universitas GadjahMada, Yogyakarta, Indonesia
** Department of Electrical Engineering, Universitas Muhammadiyah Yogyakarta, Yogyakarta, Indonesia

Article Info

Article history:

Received Jul 16, 2015
Revised Nov 2, 2015
Accepted Nov 18, 2015

Keyword:

Fuzzy Decision Trees
Loca minima
Non-Holonomic
Path Planning
Potential Field

ABSTRACT

The fuzzy logic algorithm is an artificial intelligence algorithm that uses mathematical logic to solve to by the data value inputs which are not precise in order to reach an accurate conclusion. In this work, Fuzzy decision tree (FDT) has been designed to solve the path planning problem by considering all available information and make the most appropriate decision given by the inputs. The FDT is often used to make a path planning decision in graph theory. It has been applied in the previous researches in the field of robotics, but it still shows drawbacks in that the agent will stop at the local minima and is not able to find the shortest path. Hence, this paper combines the FDT algorithm with the potential field algorithm. The potential field algorithm provides weight to the FDT algorithm which enables the agent to successfully avoid the local minima and find the shortest path.

Copyright © 2016 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Iswanto,
Departement of Electrical Engineering and Information Technology
Universitas Gadjah Mada,
Yogyakarta, Indonesia.
Email: iswanto.s3te13@mail.ugm.ac.id

1. INTRODUCTION

In general, a mobile robot navigation which moves in dynamic environments has a major issue namely path planning [1]. In path planning problems, the mobile robot requires algorithms to find a safe path to avoid a collision and needs an optimal path in achieving the targets in the unknown environment [2]. A graph theory algorithm for path planning is needed to overcome the problems.

There are three types of graph theories, namely visibility, Voronoi, and cell decompotion graphs. Some researchers such as Liu and Zhang [3] have conducted path planning researches by using the theories. The Voronoi diagram for the UAV path planning has been used. By using a global view, this graph splits equally large distance between obctacles in that the path for UAV is obtained. With the path, the UAV is able to head to the point of destination, but by applying merely this algorithm, the robot is not able to pass the local minima. Loca minima a condition that makes robot stops at many obstacles

The visibility graph algorithm was used by Jandt et. Al [4] for mobile robot path planning. This algorithm connects not only the angles of one obstacle to another but also the angles to the initial and destination points of the robot. With this algorithm, the mobile robot is able to move to the destination point in the safest path but regrettably by applying this algorithm, the robot is not able to pass the local minima.

The cell decompotition was used by Ryan [5]. This algorithm divides the environment into small grids. The grids are used to find the safest path for multi-robot. The multi-robot heads to the destination point by using the grid lines. Graph theory is used for planning the safest path. The weakness of the algorithm that it is not able to find the shortest path.

With the remaining problems, some researchers have combined the graph theory with some algorithms such as Dijkstra's algorithm by Feuerstein and Marchetti-Spaccamela [6]. This algorithm has been applied for planar path planning graph theory. By using this algorithm, the robot is able to find the shortest path. This algorithm was also implemented by Peyer et al [7] to search the shortest path in very large scale integration (VLSI).

The drawback of the Dijkstra's algorithm is that it is not able to detect obstacles when applied to the mobile robot path planning. It has been modified by Liu et al [8] by combining it with the Floyd algorithm.The Floyd algorithm is used to detect obstacles while the Dijkstr algorithm is used to find the shortest path.

Other algorithms such as A* can also determine the shortest path. This algorithm has been applied by Cheng Liping et al [9] to find a parking lot, but it is also not able to detect moving obstacles. The A* and Dijkstra's algorithms were not able to detect obstacles, thus the two algorithms were combined with the artificial intelligence algorithm to detect them. In order for A* algorithm to be able to be applied to a mobile robot, the algorithm has been modified by Duchoň et al [10] by adding phi algorithm.

In addition to Dijkstra's algorithm, the researcher has modified the cell decomposition graph theory into the tree graph theory [11]. Graph theory is often used to search for the fastest path. This theory has been supported by a decision support system by using fuzzy algorithm. This algorithm has been widely used to search data and decision makers [12]-[15]. In addition, the fuzzy algorithm has been used as a search path and to make a decision to find the shortest path. Many researchers have applied this algorithm as the search algorithm such as GH Shah Hamzei et al [16]. They use this algorithm for global planning.

The Fuzzy Decision Tree algorithm faces a problem when applied to the environment which has many obstacles. At many scenario, the agent will stop at a local minima and is not able to find the shortest path. Based on the problems, this paper combines the Fuzzy Decision Tree algorithm with potential field algorithm. The potential field algorithm provides weight to the fuzzy decision tree algorithm. With the given weight, the agentis expected to avoid local minima and determine the shortest path.

2. RESEARCH METHOD

Path planning is an algorithm that solves the problem of moving an agent from an initial to a goal position by passing obstacles. A research method by merging several path planning algorithms presented this paper are graph theory, fuzzy decision tree, and the potential field algorithms.

2.1. Graph Theory Algorithm

This paper presents global path planning method to establish an environmental model. Global path planning obtains data information that is the size of the environment and the data used for path planning such as agent position, position of obstacles and goal position as shown in figure 1. It shows that in the environment there is an initial position denoted by a blue star, a position of obstacle denoted by a red circle and a goal position of agentdenoted by a small blue circle. The environment model is divided into grids which have same size.

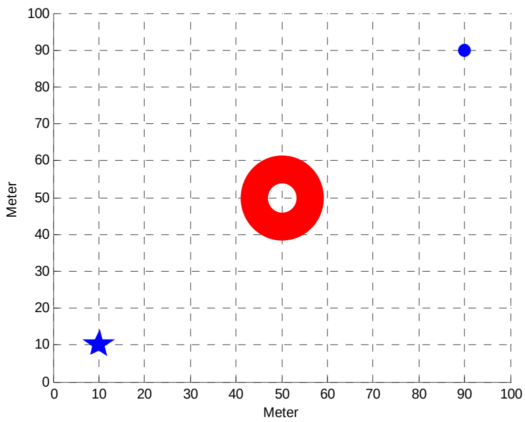

Figure 1. Environment modeling with cell decomposition

The environment is modeled bycreating a map divided into grids having same size using exact cell decomposition algorithm with source code program as shown in figure 2. It shows that (*I,J*) is the position of the agent, (X_G,Y_G) is the goal position of the agent, l_J is length of the grid, l_I is width of the grid, n is size of the environment, and (X_O,Y_O) is the position of obstacle. RG is the formula of the distance between the agent and the goal positions, Ro_1 is the formula of the distance between the agent and the obstacle positions. Then R_G and R_{O4} are summed up to become an environment matrix value.

$l_I = 1;$
$l_J = 1;$
$k = 1;$
$n = 100;$
for $I = k : l_I : n$
  for $J = k : l_J : n$
$$R_G = \sqrt{\left(\frac{J}{10} - Y_G\right)^2 + \left(\frac{I}{10} - X_G\right)^2};$$
$$R_O = \sqrt{\left(\frac{J}{10} - Y_O\right)^2 + \left(\frac{I}{10} - X_O\right)^2};$$
$$A(I,J) = \frac{1}{R_G} + R_O;$$
    if ($A(I,J) > 300$), $A(I,J) = 300$; end
  end
end

Figure 2. Source code program for environment map

2.2. Fuzzy Algorithm

Fuzzy algorithm was discovered by zadeh in 1994 used for robot navigation path planning. Some researchers such as Vachtsevanos and Hexmoor [17] used it for a wheeled robot path planning. With this algorithm, the robot could avoid static obstacles. The other researcher, Surmann et al [18] also used fuzzy algorithm for mobile robot path planning. It was used for a path planning in an environment that has many obstacles.

In a robot path planning research, fuzzy algorithm has many weaknesses, so that the algorithm is combined with cell decomposition to find the fastest and shortest called FDT. The research of determining the shortest path using fuzzy decision tree has been carried out such as by Hamzei et al [19]. In his research, entitled Self-organizing Fuzzy decision tree for Robot Navigation: An On-line Learning Approach, they implemented this algorithm to search robot path and fuzzy rule base to detect obstacles.

Turnbull et al [20] also used fuzzy decision tree algorithm for path planning of UAV. This algorithm requires a lot of data used for training. The training enables UAV to move toward the goal position and avoid obstacles. This algorithm has a weakness that is it needs a long training process. Due to the weakness a new algorithm is developed with the principle of seeing close neighborhoods. Fuzzy decision tree algorithm with close neighbor’s optimum was developed by Lertworaprachaya [21].

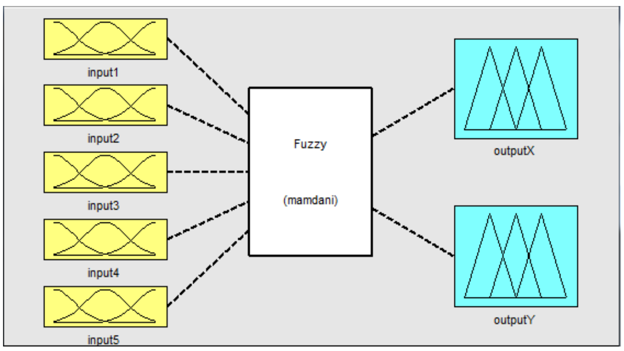

Figure 3. The environment model

This paper describes fuzzy algorithms for path planning with grid values as the fuzzy input obtained from cell decomposition in a global path planning and the fuzzy output is motion direction of an agent that the design is shown in figure 3. It shows that the fuzzy has 5 inputs and 2 outputs in which the inputs are grid values located around the agent that is in the front, in 45 degree left, 45 degree right, left and right of the grids. While the output of the fuzzy is the value of direction for the x and y-axis.

Figure 4a shows member set which has a range of 0-300 and three members i.e. small (S), medium (M), and large (B) in which each has a range of [0 90] [90 210] and [210 300] respectively. The set of fuzzy output has a range of -1 to 1, which is shown in Figure 4b. It shows that the set of members for output have three members, namely the right (R), the center (Z), and Left (L) which each has of a range [-1 -0.4] [-0.4 0.4] and [0.4 1] respectively.

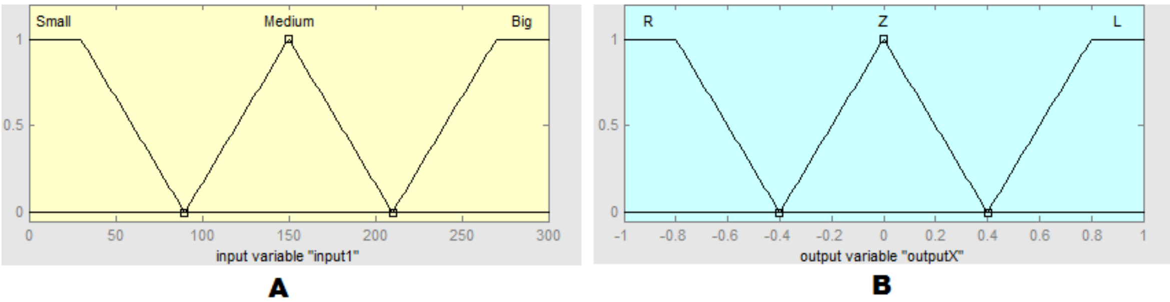

Figure 4. Fuzzy input and output

The fuzzy rule base is shown in figure 5. It is seen that there are two basic rules that is the basic rules of fuzzy X and for fuzzy Y. The basic rules of fuzzy X provides the value of direction for x-axis while the basic rules of fuzzy X provides the value of direction for x-axis. The value of direction of x and y axis move the agent towards the goal position and avoid obstacles.Variable A1 to A5 is a fuzzy input namely A1 is left input of the grids, A2 is 45 degree left, A3 is a front input, A4 is 45 degree right, and A5 is the right input of the grids.

```
if (A1<A2)&&(A1<A3)&&(A1<A4)&&(A1<A5)
   x=Negative;
   y=Zero;
else if (A2<A3)&&(A2<A4)&&(A2<A5))
   x=Negative;
   y=Positive;
else if (A3<A4)&&(A3<A5)
   x=Zero;
   y=Positive;
else if (A4<A5)
   x=Positive;
   y=Negative;
else
   x=Positive;
   y=Zero;
end
```

Figure 5. The x and y rule based fuzzy

2.3. Potential Field Algorithm

Potential field algorithm [22] is one of path planning algorithms which was developed by Khatib from the magnetic field. The equation of potential field algorithm is as follows:

$$F = F_d + F_o \tag{1}$$

where $F_d$and F_o is the attraction and repulsion. Attractive force is the force that will cause the agentto move to the destination. The attractive force equation is as follows:

$$F_d(x) = -k(x - x_d) \tag{2}$$

In which k is the attractive potential field constant. The value of k is $0 < k < 1$. x shows the destination point of the agentand $x_d$shows the coordinates of the agent.

The repulsion equation is as follows:

$$F_{xO}(\rho) = \vartheta\left(\frac{1}{\rho} - \frac{1}{\rho 0}\right)\frac{1}{\rho^2}\frac{\delta\rho}{\delta x} \tag{3}$$

In which ϑ is the repulsive potential field constant. The value of ϑ is $0 < \vartheta < 1$. ρ shows the coordinates of the agentand $\rho_O$shows the point of the obstacle. Many researchers have modified the potential field algorithm such as Rizqi et al [23] whohas modified repulsive potential field. Ren et al [24] has modified potential field by using Newton's laws and applied it to mobile agents. Velagic et al [25] has modified the potential field in order to avoid local minima.

In this paper, the potential field algorithm was modified to provide weight for the value of environmentmatrixes. To obtain the weight value, the Khatib's equation of the attraction potential field was modified into the following:

$$F_{x_d}(x) = k\sqrt{(J - Y_G)^2 + (I - X_G)^2} \tag{4}$$

In which k is the attractive potential field constant. (X_G, Y_G) shows the destination point of the agentand (I, J)shows the coordinates of the agent.

While Khatib's repulsion potential field equation was modified into the following:

$$F_{xOn} = \frac{K_{Rn}}{\sqrt{(J - Y_{On})^2 + (I - X_{On})^2}} \tag{5}$$

In which k_{Rn} is the attractive potential field constant. (X_{On}, Y_{On}) shows the coordinates of the obstacle with On is number of obstacle.

2.3. Environment Model Modified

This study explains fuzzy decision tree algorithm merged with cell decomposition algorithm for an agent path planning. The agent moves the goal position in an unknown environment. In the simulation, the merger of the two algorithms are unable to find the shortest path, therefore it is necessary to modify the method in order for the path planning to be optimal in finding the shortest path. The modification is conducted by adding potential field algorithm in the environment model with the source code program shown in figure 6.

for $I = k:l:n$

  for $J = k:l:n$

$$F_{x_d} = K_A\sqrt{\left(\frac{J}{10} - Y_G\right)^2 + \left(\frac{I}{10} - X_G\right)^2};$$

$$F_{x_O} = \frac{K_R}{\sqrt{\left(\frac{J}{10} - Y_O\right)^2 + \left(\frac{I}{10} - X_O\right)^2}};$$

    $A(I,J) = F_{x_d} + F_{x_{O4}}$;

    if $A(I,J) > 300$, $A(I,J) = 300$; end

  end

end

Figure 6. Source code program for environment map with modified potential field algorithm

Figure 6 shows that F_{x_d} is the attractive of potential fields as shown in equations 4 while F_{x_O} is the repulsive of field potential as shown in equation 5. K_A is the constant of the potential field attractive, and and K_R are the constant of the field potential repulsive. The sum of potential field force is used for the weight value of columns and rows in an environment matrix. If the sum of attrative and repulsive is a very large, then a limit value is given.

3. RESULTS AND ANALYSIS

The researchers used two softwares namely ROS (Robot Operating System) and Matlab. By using the matlab, the design and simulation for the Fuzzy decision tree Artifial Potential Fields (FDTAPF) algorithm could be created. Experiments were carried out with the design environment that was conducted by LIANG [29] with size of 100x100. Afterward, the environment Matrix derived from Liang's was created to simulate the mobile robot. There were 2 experiments created ie in environments with multi obstacles and in environments with local minima.

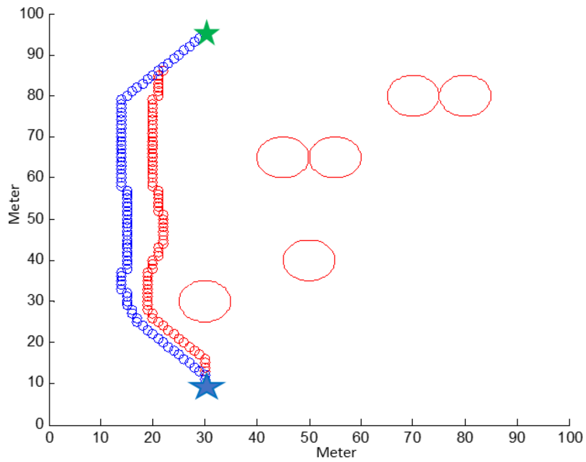

Figure 7. Experiment 1 in environment 1 with FDT and FDTAPF algorithm

In experiment 1 in environment 1, the goal is to move the agent from the initial point to the destination point as shown in Figure 7 with FDT and FDTAPF algorithms. It shows that there is an environment that has six obstacles. The agent is placed in the position of (30, 95) denoted by green star. The agentmoves towards the destination point (30, 10) denoted by blue star. In the figure it can be seen that there are two lines, namely red and blue. It is seen that the FDTAPF algorithm with red line is better than the FDT algorithm with blue line in term of that the red line has a shorter distance to reach the final destination.

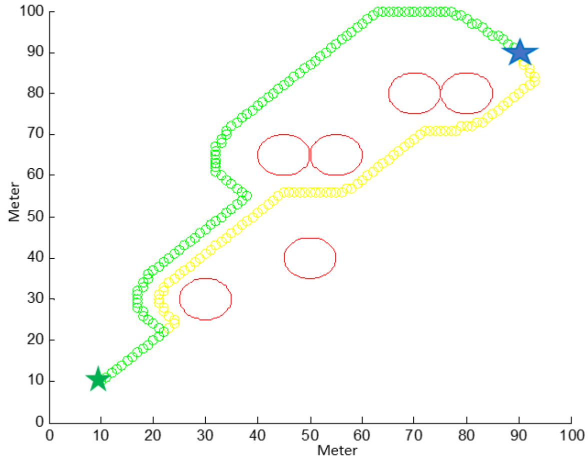

Figure 8. Experiment 2 in environment 1 with FDT and FDTAPF algorithm

In the following simulations, the initial and destination position point of the agnet is changed as shown in Figure 8 with FDT and FDTAPF algorithms. In experiment 2 in the environment 1, the initial position of the agnet is placed in (10, 10) denoted by green star and the destination point (90, 90) denoted by blue star. In the figure it can be seen that there are two lines, namely yellow and green. It is seen that the FDTAPF algorithm with yellow line is better than the FDT algorithm with green. By using FDTAPF algorithm, the agent can find the shortest path in both experiments in environment 1.

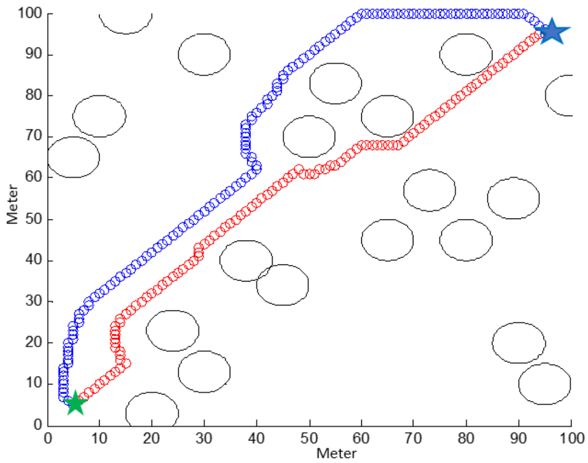

Figure 9. Experiment 1 in environment 2 with FDT and FDTAPF algorithm

In the first experiment in environment 2, the goal is to move the agent from the initial point to the destination point as shown in Figure 9 with FDT and FDTAPF algorithms. It shows that there is an environment that has obstacles. The agentis placed in the position of (5, 5)denoted by green star. The agent moves towards the destination point (95, 95)denoted by blue star. From the figure that it can be seen that there are two lines, namely red and blue. From the figure it is seen that the FDTAPF algorithm with red line is better than the FDT algorithm with blue line in term of that the red line has shorter distance to reach the final destination.

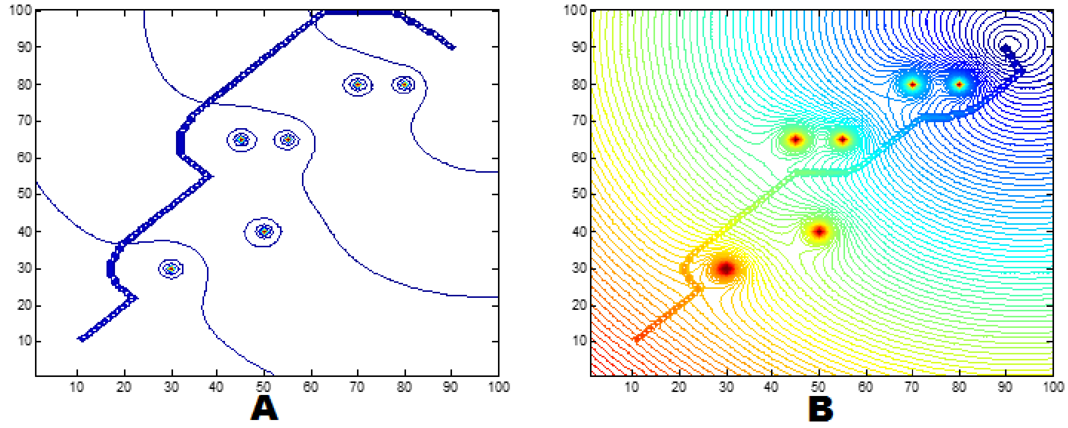

Figure 10. Experiment 1 in environment 2 with contour

Experiment 1 in environment 2 with FDT and FDTAPF algorithm can be analyzed by using contours as shown in Figure 10. In the figure, the contour difference between a weighted environment with a weightless environment is visible. By adding weights provided by potential algorithm, several contours exist in the environment which enable agent to find the shortest path. The starting point of the agent is shown in red color which has a high surface contour. While the destination point of the agent is shown in blue color which has low surface contour. Thus the agent will move to the goal from the higher contour towards the lower contour so that the path planning agentis more optimal.

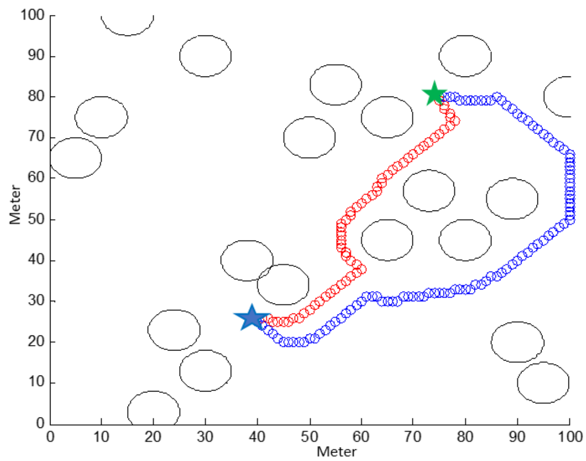

Figure 11. Experiment 2 in environment 2 with FDT and FDTAPF algorithm

In the following simulations, the initial and destination position point of the agentis changed as shown in Figure 11 with FDT and FDTAPF algorithms. In experiment 2 in environment 2, the inisial position point of the agent (45, 25) denoted by green starand the destination point (75, 80) denoted by blue star. From the figure that it can be seen that there are two lines, namely red and blue. From the figure it is seen that the FDTAPF algorithm with red line is better than the FDT algorithm with blue line in term of that the red line has shorter distance to reach the final destination. By using FDTAPF algorithm, the agen tcan find the shortest path in the environment 2.

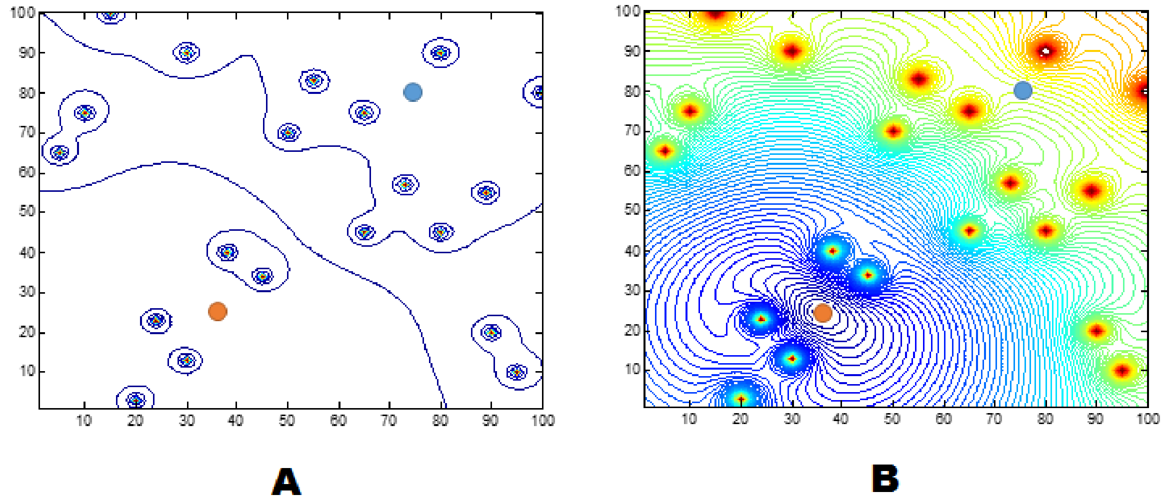

Figure 12. Experiment 2 in environment 2 with contour

Figure 12 shows that the agentpath planning environment can be analyzed using contour. Figure A is path planningcontour that do not use weight in the environment, while figure B is path planning contour which uses weights in the environment. The starting point of the agent is shown in red color which has a high surface contour. While the destination point of the agentishown in blue color which has low surface contour. Thus the agentwill move to the goal from the higher contour towards the lower contour so that the path planning agentis more optimal.

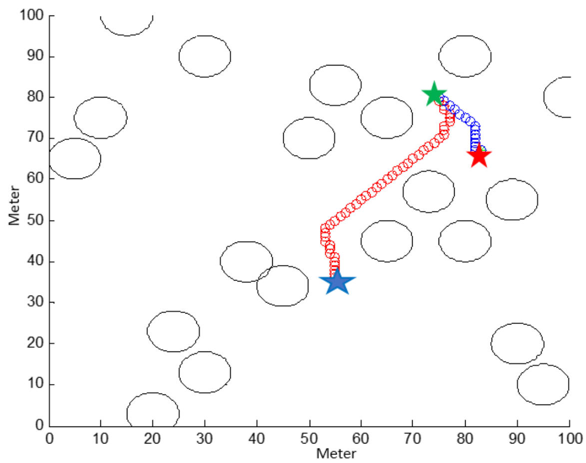

Figure 13. Experiment 3 in environment 2 with FDT and FDTAPF algorithm

In experiment 3 in environment 2 shown in figure 13 the agentis in the environment with local minima. The agent stops if the agentcannot find the smallest weight. It can be overcome by using a potential field algorithm to obtain weight. In the figure it is seen that there are two color lines namely red and blue. It is seen that the FDTAPF algorithm with the red line is better than the FDT algorithm with the blue line. The agentin the blue line is not able to move and avoid local minima. While the agentin the red line is influenced by repulsion and attraction of potential field in that the agentis able to avoid local minima.

4. CONCLUSION

Fuzzy decision trees algorithm has been used in the previous researches to find a mobile agent path. By applying the algorithm, the agentis able to move to the destination point and detect multiple obstacles. When applied to an environment that has many obstacles, however, the algorithm is not able to avoid loca minima. Therefore, this paper presents the modification of the FDT algorithm with potential field algorithm. Potential fiel algorithm provides weightto FDT algorithm. With the weight values provided by the potential field algorithm, the fuzzy decision tree algorithm is able to to find the shortest path and avoid local minima.

REFERENCES

[1] K.C. Brata, et al, Location-Based Augmented Reality Information for Bus Route Planning System, *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 5, No. 1, pp. 142-149, February 2015.

[2] N. Arman and F. Khamayseh, "A Path-Compression Approach for Improving Shortest-Path Algorithms," *Int. J. Electr. Comput. Eng.*, vol. 5, no. 4, pp. 772–781, 2015.

[3] L. Liu and S. Zhang, "Voronoi diagram and GIS-based 3D path planning," in 2009 *17th International Conference on Geoinformatics, Geoinformatics 2009*, 2009, pp. 12–16.

[4] J. A Jandt, R.C. Luo, and M.G. Kay, "The Essential Visibility Graph : An Approach to Global Motion Planning for Autonomous Mobile Robots", in *1995 IEEE International Conference on Robotics and Automation*, 1965, pp. 1958–1963.

[5] M. Ryan, "Graph decomposition for efficient multi-robot path planning", in *IJCAI International Joint Conference on Artificial Intelligence*, 2007, pp. 2003–2008.

[6] E. Feuerstein and A. Marchetti-Spaccamela, "Dynamic algorithms for shortest paths in planar graphs", *Theor. Comput. Sci.*, vol. 116, no. 2, pp. 359–371, 1993.

[7] S. Peyer, D. Rautenbach, and J. Vygen, "A generalization of Dijkstra's shortest path algorithm with applications to VLSI routing", *J. Discret. Algorithms*, vol. 7, no. 4, pp. 377–390, 2009.

[8] H. Liu, N. Stoll, S. Junginger, and K. Thurow, "A Floyd-Dijkstra hybrid application for mobile robot path planning in life science automation", in *2012 IEEE International Conference on Automation Science and Engineering (CASE)*, 2012, pp. 279–284.

[9] L. Cheng, C. Liu, and B. Yan, "Improved Hierarchical A-star Algorithm for Optimal Parking Path Planning of the Large Parking Lot", in *2014 IEEE International Conference on Information and Automation (ICIA)*, 2014, no. July, pp. 695–698.

[10] F. Duchoň, A. Babinec, M. Kajan, P. Beňo, M. Florek, T. Fico, and L. Jurišica, "*Path Planning with Modified a Star Algorithm for a Mobile Robot*", Procedia Eng., vol. 96, pp. 59–69, 2014.

[11] O. Amini, D. Coudert, and N. Nisse, "Non-deterministic graph searching in trees", *Theor. Comput. Sci.*, vol. 580, pp. 101–121, 2015.

[12] C. Olaru and L. Wehenkel, "A complete fuzzy decision tree technique", *Fuzzy Sets Syst.*, vol. 138, no. 2, pp. 221–254, 2003.

[13] A. Kumar, M. Hanmandlu, and H. M. Gupta, "Fuzzy binary decision tree for biometric based personal authentication", *Neurocomputing*, vol. 99, pp. 87–97, 2013.

[14] X. Liu, X. Feng, and W. Pedrycz, "Extraction of fuzzy rules from fuzzy decision trees: An axiomatic fuzzy sets (AFS) approach", *Data Knowl. Eng.*, vol. 84, pp. 1–25, 2013.

[15] A. Kumar, M. Hanmandlu, and H.M. Gupta, "Ant colony optimization based fuzzy binary decision tree for bimodal hand knuckle verification system", *Expert Syst. Appl.*, vol. 40, no. 2, pp. 439–449, 2013.

[16] G.H. Shah Hamzei and D.J. Mulvaney, "On-line learning of fuzzy decision trees for global path planning", *Eng. Appl. Artif. Intell.*, vol. 12, no. 1, pp. 93–109, 1999.

[17] G. Vachtsevanos and H. Hexmoor, "A fuzzy logic approach to robotic path planning with obstacle avoidance", in *1986 25th IEEE Conference on Decision and Control*, 1986, pp. 1262–1264.

[18] H. Surmann, J. Huser, and J. Wehking, "*Path planning far a fuzzy controlled autonomous mobile robot*", in Proceedings of IEEE 5th International Fuzzy Systems, 1996, vol. 3, pp. 1660–1665.

[19] G.H.S. Hamzei, "Self-organising Fuzzy Decision Trees for Robot Navigation: An On-line Learning Approach", 1998 *IEEE Int. Conf. Syst. Man, Cybern.*, pp. 2332–2337, 1998.

[20] O. Turnbull, A. Richards, J. Lawry, and M. Lowenberg, "*Fuzzy Decision Tree Cloning of Flight Trajectory Optimisation for Rapid Path Planning*", in Proceedings of the 45th IEEE Conference on Decision and Control, 2006, pp. 6361–6366.

[21] Y. Lertworaprachaya, Y. Yang, and R. John, "Interval-valued fuzzy decision trees with optimal neighbourhood perimeter", *Appl. Soft Comput.*, vol. 24, pp. 851–866, 2014.

[22] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", *Int. J. Rob. Res.*, vol. 5, no. 1, pp. 90 – 98, 1986.

[23] A.A.A. Rizqi, A.I. Cahyadi, and T.B. Adji, "Path planning and formation control via potential function for UAV Quadrotor", *2014 Int. Conf. Adv. Robot. Intell. Syst.*, no. Aris, pp. 165–170, 2014.

[24] J. Ren, K.A. McIsaac, and R. V. Patel, "Modified Newton's method applied to potential field-based navigation for mobile robots", *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 384–391, 2006.

[25] J. Velagic, B. Lacevic, and N. Osmic, "Efficient Path Planning Algorithm for Mobile Robot Navigation with a Local Minima Problem Solving", in *2006 IEEE International Conference on Industrial Technology*, 2006, pp. 2325–2330.

[26] X. Liang, L. Li, J. Wu, and H. Chen, "Mobile robot path planning based on adaptive bacterial foraging algorithm", *J. Cent. South Univ.*, vol. 20, no. 12, pp. 3391–3400, 2013.

BIOGRAPHIES OF AUTHORS

Iswanto was born in Sleman, Yogyakarta, Indonesia, in 1981. He received the B.S degree and M.Eng degree from Universitas Gadjah Mada, Yogyakarta, Indonesia in 2007 and 2009. Now, he is on Phd Program at Universitas Gadjah Mada. He has been a Lecturer and Researcher in the Electrical Engineering Department at Universitas Muhammadiyah Yogyakarta since 2010. His current research is focused on formation control, path planning and Control UAV.

Oyas Wahyunggoro was born in Yogyakarta, Indonesia, in 1967. He received the under graduate and master degrees from Universitas Gadjah Mada, Yogyakarta, Indonesia, in 1993 and 2001. In 2011, he received Ph.D. degree in Universiti Teknologi PETRONAS, Malaysia. He has been a Lecturer and Researcher at Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada since 1994. His research interests are in the area of Fuzzy Logic, Genetic Algorithms, Control

Adha Imam Cahyadi got his undergraduate degree in Electrical Engineering from Universitas Gadjah Mada, Yogyakarta Indonesia in 2002. After a year working in industry such as in Matsushita Kotobuki Electronics and Halliburton Energy Services as engineer, he pursued his master degree in Control Engineering from KMITL Thailand and graduated in 2005. Later he obtained Doctor of Engineering from Tokai University Japan in 2008. His research interests are in teleoperation systems and robust control for delayed systems especially process plant. Currently he is a lecturer at the Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada and a visiting lecturer at the Centre for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia, Malaysia