NOVAL TECHNIQUE TO FIND SHORTEST PATH PROBLEM A* ALGORITHM FOR ENHANCING THE DURING REAL TIME IMFORMATION

Parveen Kumar[1], Dr. Rishu Bhatia[2]
[1]M.Tech (ECE), [2]Asst. Prof., Dept. Of ECE, Gitam, Kablana

ABSTRACT: To facilitate intelligent traffic communication, a reliable, realistic, and network is required for Adhoc communication between vehicles. Many requirements compete in the different existing vehicular networking systems. However, one solution to suit all purposes has not been determined. While the vehicle-to infrastructure, such as standardized GPRS, HSPAM, and LTE, is currently used, vehicle-to-vehicle and vehicle to-roadside units are more or less under construction. 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. Here we are making use of AI 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. Position Tracking System to find shortest path to object aimed 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. This inclusion gives us more dynamic model and this real-time situation provides it a best model for future. This enhancement brings us more closely 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 saves time for travelers. Further we implement the results of tests using wireless sensor network (WSN) radios of different kinds to communicate with moving vehicles. Our aim is to provide measured background information about different kinds of vehicular networking methodologies in order to find an optimal solution for different kinds of usage scenarios. The proposed technique use of real time information gives us more dynamic model to find the shortest path. It provide real information makes it a best model for future and enhancement brings us more closely to real time situations & save time for travelers.

Key Word: Fuzzy logic & Artificial Intelligence, intelligent traffic communication, wireless sensor network (WSN), MATLAB

I. INTRODUCTION

Motor vehicles, as a kind of modern transport means, with the advantages of high speed and convenience, are important to people’s daily travel. Activities like going out, travelling to work, shopping, and visiting friends and relatives are often done by using motor vehicles. With the development of economics and technology, the number of motor vehicles has increased rapidly in the past decades. Davis in 2012 showed that from 1990 to 2009 in selected countries, the average annual percentage change of the number of cars is 2.3%, for trucks and buses the average annual percentage change is 3.9%, and the growth is still proceeding. Availability of more vehicles makes it more convenient for people to travel and merchandise transport. The increase of the number of vehicles also brings stresses to public traffic and pollution to the environment.

1.1 The Shortest-Time Path

In general, the shortest-time path, as the name implies, is the path which costs the shortest time of all possible paths from one place to another. In order to figure out such path, the time cost of each path should be able to be measured or calculated. However, in reality, the time cost of a journey not only depends on the fixed length of the roads which the driver chooses, but also depends on the complex traffic condition and the variable driving speed, so in reality, the shortest path is probably not the shortest-time path.

1.2 How Shortest Path Help For Navigation System.

Shortest Path problems are inevitable in road network applications such as city emergency handling and drive guiding system, in where the optimal routings have to be found. As the traffic condition among a city changes from time to time and there are usually a huge amounts of requests occur at any moment, it needs to quickly find the solution. Therefore, the efficiency of the algorithm is very important . Some approaches take advantage of preprocessing that compute results before demanding. These results are saved in memory and could be used directly when a new request comes up. This can be inapplicable if the devices have limited memory and external storage.

II. LITERATURE SURVEY

Mobile robots are more efficient than legged or treaded robots on hard as well as smooth surfaces, and have potential enough to find widespread application in industry, because of the hard, smooth plant floors in existing industrial environments [1]. Several configurations for mobility can be found in the applications as mentioned by Jones et al. [2]. The most common form single-body robots are the differential drive and synchronic drive tricycle or car-like drive, and omnidirectional steering robots [3]. Besides the relevance in applications, the problem of autonomous motion planning and control of mobile robot has attracted the interest of many researchers to view its theoretical challenges

[4]. The motion control of wheeled mobile robots has been able to draw considerable attention over the past few years. The nonholonomic behavior in robotic systems is particularly interesting; since it points out that the mechanism can be completely controlled by using a reduced number of actuators. Particularly, these systems are typical examples of nonholonomic mechanisms due to the perfect application of the rolling constraints on the wheel motion [5]. Several controllers have been proposed for the motion control of mobile robots with nonholonomic constraints, where the two main approaches to controlling mobile robots are posture stabilization and trajectory tracking. The procedure of modelling can be inspired by definition of a wheeled mobile robot according to Muri and Neuman [6] as this, “A robot capable of locomotion on a surface solely through the actuation of wheel assemblies mounted on the robot and in contact with the surface. A wheel assembly is a device that provides a relative motion between its mount and a surface on which it is intended to have a single point of contact.” However it is required that the vehicle kinematic design has the appropriate degrees of freedom (mobility) so that it can adapt to the variations in the surface and the wheels roll without slip. Mobility is enhanced by the use of omnidirectional wheels instead of conventional wheels [7]. The necessity of ideal rolling without sideways slipping for wheels enforces non-holonomic (non integrable) constraints on the motion of the wheels of mobile robot [8]. The relation between the rigid body motion of the robot and the steering and drive rates of wheels has been developed by Alexander and Maddocks [9] based on constraint as “rolling without sliding”. Slippage due to misalignment of the wheels is investigated here by minimization of a non smooth convex dissipation functional that is derived from Coulomb's Law of friction. This minimization principle is equivalent to the construction of quasi-static motions. Three related but different kinematical aspects have to be considered when designing a robot. They can be listed as mobility, control and positioning [10, 11]. The first one, mobility, deals with the possible motions that the robot can follow in order to reach its final destination in any orientation. The second aspect, control, relates to the choice of the kinematical variables: generalized velocities or coordinates. Finally, the third aspect, positioning, considers the localization system that is used to estimate the actual position and orientation of the robot by reducing the robot’s region of uncertainty based on sensor measurements necessary to achieve an autonomous operation [12].

III. EARLIER WORKS

3.1 Navigation Using Fuzzy Logic Controller for Mobile Robot

Fuzzy Logic technique plays an important role in the designing of an intelligent controller for mobile robot. This technique is used for navigation of mobile robots. Fuzzy set theory provides a mathematical framework for representing and treating uncertainty in the sense of vagueness, imprecision, lack of information and partial truth. Fuzzy control systems employ a mode of approximate reasoning that resembles the decision-making process of humans. A fuzzy system is usually designed by interviewing an expert and formulating the implicit knowledge of the underlying process into a set of linguistic variables and fuzzy rules. In particular for complex control tasks, obtaining the fuzzy knowledge base from an expert is often based on a tedious and unreliable trial and error approach. Fuzzy set theory was introduced by Lofti Zadeh in the mid sixties. In 1965 Lotfi Zadeh proposed fuzzy set theory, and published a paper. Fuzzy logic has been applied to diverse fields, from control theory to artificial intelligence. This section presents a variety of fuzzy logic techniques which address the challenges posed by autonomous robot navigation.Autonomous mobile robot navigation in uncertain and dynamic environments demands adaptation and perception capabilities. Reactive control strategies imply a strong dependency on sensed information about the robot’s environment. Thus, imprecision and uncertainties in perception from sensors have to be considered. While the rules are based on qualitative knowledge, the membership functions defining the linguistic terms provide a smooth interface to the numerical process variables and the set-points.

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.

4.1 How Shortest Path Calculation Processed By Real Time Information Model

- Shortest path is being calculated by Grid Algorithm.
- It also give 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. RESULT AND SIMULATION

The proposed work include A* algorithm and improved TEEN algorithm which is simulating on MATLAB and result come out in graphical and numeric as below.

5.1 Parameter Consideration

5.1.1 TEEN:

TEEN (Threshold sensitive Energy Efficient sensor Network protocol) is for reactive networks.

Teen protocol is for a simple sensing applications.

Energy efficient protocol than conventional sensor network protocols

Functioning

Every node in a cluster takes turns to become the CH for a time interval called cluster period

At every cluster change time, in addition to the attributes, the cluster-head broadcasts to its members, a hard & a soft threshold.

5.1.2 A star PseudoCode*

Sometimes things are easier to understand in pseudo code.

Inputs

map

start and goal locations

Internal Data

fringe - a list of map locations to be evaluated, in ascending order of estimated distance

closedList - a list of map locations that have been fully evaluated

Data Structure

RouteNode, contains

a map location

pointer to this node's parent node

d, the actual distance traveled to reach this node

dPlusL2, which is d + linear distance to goal

Search()

Put start node onto fringe

endNode = findRoute()

findRoute()

if fringe is empty

// No route exists between start and goal.

return 0

else

node = remove first fringe node (it will have the shortest estimated distance to the goal)

if node's location is the goal

return RouteNode data for current location

else

if node's location is not on the closedList

add node to closedList

addChildrenToFringe(node)

return findRoute()

addChildrenToFringe(parentNode)

for all children of parentNode

if child's location is not on closedList

childNode = new RoutNode()

childNode.parent = parentNode

childNode.d = parent.d + linearDistance(parent, child)

L2 = linearDistance(childNode, goal)

childNode.dPlusL2 = childNode.d + L2

Add childNode to fringe, maintaining ascending dPlusL2 order

Rendering the Path from the Closed List

After all the iterations, and the goal has been reached, the path is then rendered by backtracking from the goal to start.

This is often done by iterating backward through the closedlist, going from the goal node's parent node to each preceding node's parent until the start is found.

6. Simulation and Result

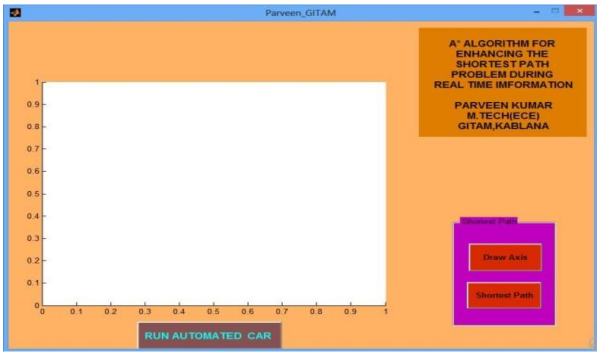

Fig 5.1: Main GUI of Proposed Work

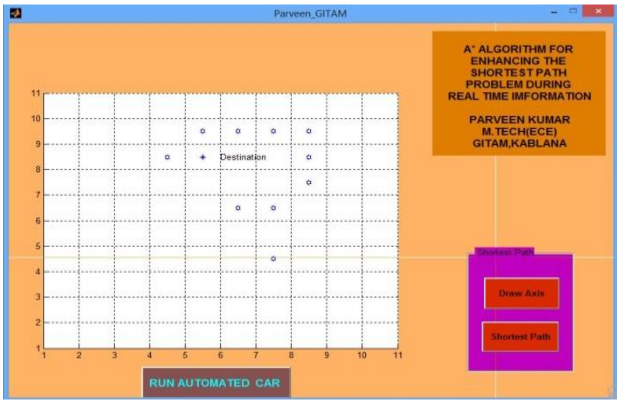

Fig 5.2: Calculation of Shortest Path for WSN

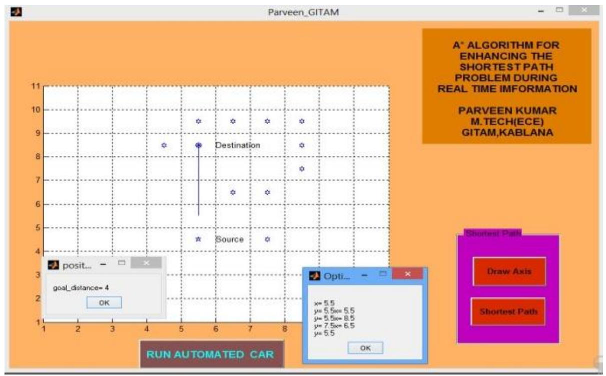

Fig 5.3: After the completion of shortest path (A result in shorted path in POP UP BOX)

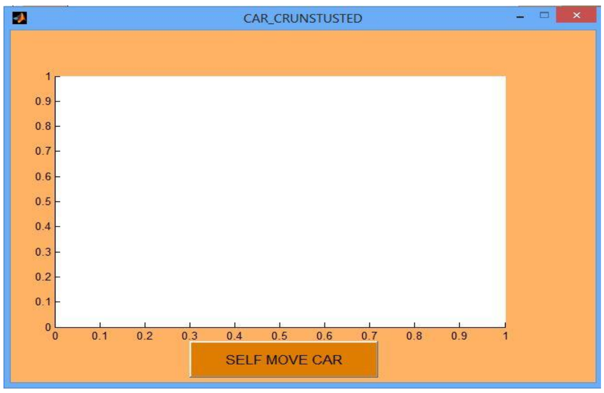

Fig 5.4: Area of moving CAR (One WSN Node Move Automatic)

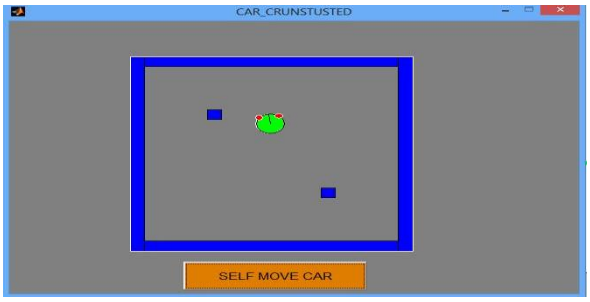

Fig 5.5: A Automated CAR moving for one day (A Green is Constructed CAR) and Blue is Obstacle

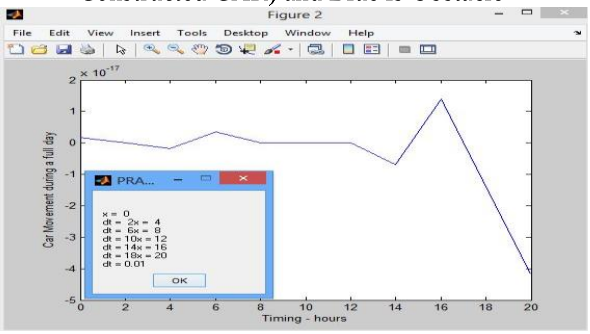

Fig 5.6: A Differential Value of movement of car during the whole day

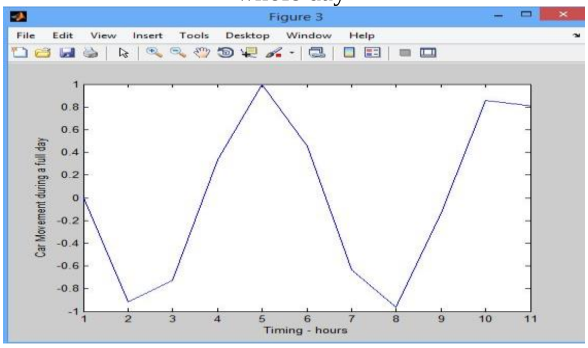

Fig 5.7: Decision of CAR when it drastically changes its movement long run up

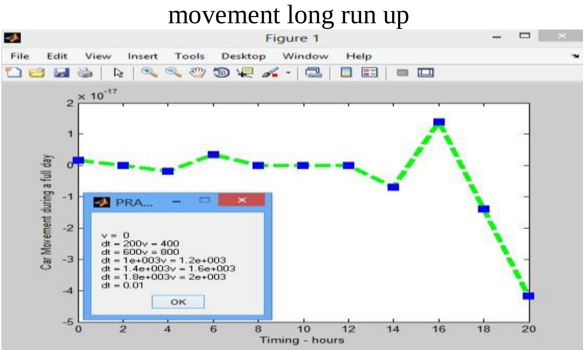

Fig 5.8: An operation point that has also come in pop up box that is speed time calculation of sensor node mounted on CAR.

All above figure represent the MATLAB figure which comes out after the run of simulation. It has noticeable that the shortest path ahs result in numeric but it is just a presentation of unit. In a large scale it has different GUI. This is just the basic part of real time shortest path and running the automated Car On the given area. Here a circular robot that represent the automated CAR travelling for whole day, but in simulation it is reduce the time. But the ultimate goal is to achieve the core function of automatization along with the shortest path.

5.4 Comparison

Table 5.1: Comparison of individuals with proposed integrated system

	A* Star Algorithm	TEEN Algorithm	Integrated System
JAMMING INFORMATION	No	Yes	Yes
SHORTEST PATH	Yes	No	Yes
REAL TIME INFORMATION	No	Yes	Yes
CALCULATION SPEED	Very Fast	Slow	Very Fast
PLATFORM	MATLAB	MATLAB	MATLAB

5.5 Concluded Result

- The use of real time information gives us more dynamic model to find the shortest path.
- This real information makes it a best model for future.
- This enhancement brings us more closely to real time situations & save time for travelers.
- It could apply for Automated Vehicles for long journey.

VI. CONCLUSION & FUTURE WORK

In our study, all the hinder factors are fixed constructions, we provide the path with less hinders to the traffic and driving speed. The method may serve as a reference while doing urban planning. Careful planning of building roads close to those constructions (schools, hospitals, traffic lights and residential areas) will diminish the risk of traffic congestion. To the researchers and students of Geo-science and computer science, the method to implement the function of our application can be a reference when they do similar work and study. Besides this, as utilizing various techniques and tools in the application, like GIS, traffic analysis, web development and so on, our application will be useful to people who are interested in these techniques or tools. The application works according to the description, but there are problems that should be solved in the further study. The algorithm needs to be improved in order to speed up the long response time. In the future, we hope we can maintain the application project regularly and improve the method, to make our study be usable to the public. All the tests were made in the simulated environment. The interface was only tested under laptops, iPhone and iPad, and the compatibility with other mobile systems was not tested. We get the following conclusion from the tests shown in the result part: both the number of hinder factors and the weight of each hinder will affect the result of the shortest-time path. Although the hinder factors in our study show how they impact the travel of motor vehicles, and we provide a shortest-time path from one place to another, the real traffic is more complex than what we analyzed in our research. Besides the factors we discussed in our study, some other factors are also significant to the traffic and driving speed, but difficult to be implemented in the application. For example, the numbers of vehicles and pedestrians. Road traffic congestion in reality is the result of dynamic behavior of and interactions between many road users (Verhoef & Rouwendal, 2004). Actually, it is impossible to count exactly how many people and vehicles are on the roads, it is hard to collect the dynamic data, and it is difficult to put it as a factor into calculation, because the value is changing as time goes by. Weather condition is another important hinder factor to traffic, but the impact is hard to estimate before it happens. In reality, it is impossible to find out the real shortest-time path, we just introducing a new method to find the possible time efficient path according to factor considerations. If there are sufficient spatial data and statistics data, the researchers can choose more proper hinder factors and weight the factor in a more precise way, to make the result more reasonable.

REFERENCES

[1] Dudgeon, D.E. and R.M. Mersereau, Multidimensional Digital Signal Processing. 2014, Englewood Cliffs, New Jersey: Prentice-Hall.
[2] Castleman, K.R., Digital Image Processing. Second ed. 2009, Englewood Cliffs, New Jersey: Prentice-Hall.
[3] Oppenheim, A.V., A.S. Willsky, and I.T. Young, Systems and Signals. 2007, Englewood Cliffs, New Jersey: Prentice-Hall.
[4] Elias C. Eze, Sijing Zhang and Enjie Liu “Vehicular Ad Hoc Networks (VANETs): Current State, Challenges, Potentials and Way Forward” September 2014 , Proceedings of the 20th International Conference on Automation & Computing, Cranfield University, Bedfordshire, UK, 12-13 September 2014 @ IEEE
[5] Bin Tian, Kun Mean Hou, Hongling Shi, Xing Liu, Yibo Chen, Jean-Pierre Chanet “Application of Modified RPL under VANET-WSN Communication Architecture” 2013 International Conference , Blaise Pascal University Clermont-Ferrand, France, 978-0-7695-5004-6/13 © 2013 IEEE & DOI 10.1109/ICCIS.2013.387.
[6] Rua Qin, Zi Li, Yanfei Wang, Xuejia Lu and Wen sheng Zhang & Guiling Wang “An Integrated Network of Roadside Sensors and Vehicles for Driving Safety: Concept, Design and Experiments” 2014 by u.s. Government work not protected by U.S. copyright @ IEEE
[7] Mohammad Jalil Piran , G. Rama Murthy , G. Praveen Babu , Ehsan Ahvar “ Total GPS-free Localization Protocol for Vehicular Ad Hoc and Sensor Networks (VASNET) “ 2011 Third International Conference on Computational Intelligence, Modelling & Simulation, 978-0-7695-4562-2/11 © 2011 IEEE & DOI 10.1109/CIMSim.2011.77.
[8] Ayonija Pathre Chetan Agrawal Anurag Jain P.G.scholar Department of CSE RITS – Bhopal, India “A Novel Defense Scheme against DDOS Attack in VANET” 978-1-4673-5999-3/13/$31.00 ©2013 IEEE.
[9] Varsha Raghuwanshi, Simmi Jain “Denial of Service Attack in VANET: A Survey” International Journal of Engineering Trends and Technology (IJETT) – Volume 28 Number 1 - October 2015 ISSN: 2231-5381 http://www.ijettjournal.org Page 15.
[10] Vinh Hoa LA, Ana CAVALLI “SECURITY ATTACKS AND SOLUTIONS IN VEHICULAR AD HOC NETWORKS: A SURVEY” International Journal on AdHoc Networking Systems (IJANS) Vol. 4, No. 2, April 2014 DOI : 10.5121/ijans.2014.4201.

