Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

DESIGN AND IMPLEMENTATION OF SELF DRIVING VEHICLE USING DIJKSTRA'S ALGORITHM

Duda. Prasad[1], S.P. Manikanta[2], B. Shubhaker[3].

[1] Assistant Professor, Department of ECE, St. Martin's Engineering College, Dulapally, Secunderabad, Telangana, india-500100. (email: prasadece@smec.ac.in)

[2] Associate Professor, Department of ECE, St. Martin's Engineering College, Dulapally, Secunderabad, Telangana, india-500100. (email: spmanikantaece@smec.ac.in)

[3] Assistant Professor, Department of ECE, BVRIT Hyderabad College of Engineering for Women, Bachupally, Secunderabad, Telangana, india-500090. (email: shubhaker.subbu@gmail.com)

--************************--

Abstract:

This paper is navigation-based [1] architecture designed for the purpose of finding the navigation path between the nodes. Genetic algorithm is a graph search algorithm that solves the single-source path problem for a graph with non-negative edge path costs. This algorithm is often used in routing and as a subroutine in other graph algorithms. For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost between that vertex and every other vertex.

Keywords: ARM Processor, Dijkstra's algorithm, nodes, shortest path.

--************************--

1. OBJECTIVE:

The objective of this paper is to develop an efficient application for constructing the shortest path along with obstacle avoidance using ultrasonic sensor which provides a real trajectory for autonomous vehicle navigation in the outdoor environments

2. INTRODUCTION:

Autonomous Driving has been said to be the next big disruptive innovation in the years to come. Considered as being predominantly technology driven, it is supposed to have massive societal impact in all kinds of fields. According to Marlon G. Boarnet, a specialist in transportation and urban growth at the University of Southern California "Approximately every two generations, we rebuild the transportation infrastructure in our cities in ways that shape the vitality of neighborhoods; the settlement patterns in our cities and countryside; and our economy, society and culture" and as many believe, autonomous driving cars are this new big change everyone is talking about. Leading not only to high impact environmental benefits such as the improvement of fuel economy, through the optimization of highways, the reduction of required cars to only 15% of the current amount needed, and platoon driving that would save to 20-30% fuel consume, but also leading to societal aspects such as immense productivity gains while commuting, decline on the accident and death tolls considered as the eight highest death cause worldwide in 2013, stress reduction , and the decline of parking space to up to ¼ of the current capacity.

2.1.Existing system:

- Time consumption for path selection
- No layer architecture
- No monitoring sections.

2.2.Proposed system:

- Genetic algorithm is used
- Wireless communication
- Destination display
- Less time for selecting path
- Automatic transmission

2.3.Hardware:

- ARM Processor
- Robotic module
- Ultrasonic sensor
- Printed Circuit board
- Driver unit

2.4.Software:

- Keil
- MATLAB
- Orcad
- Flash Magic
- Proteus

3. BLOCK DIAGRAM:

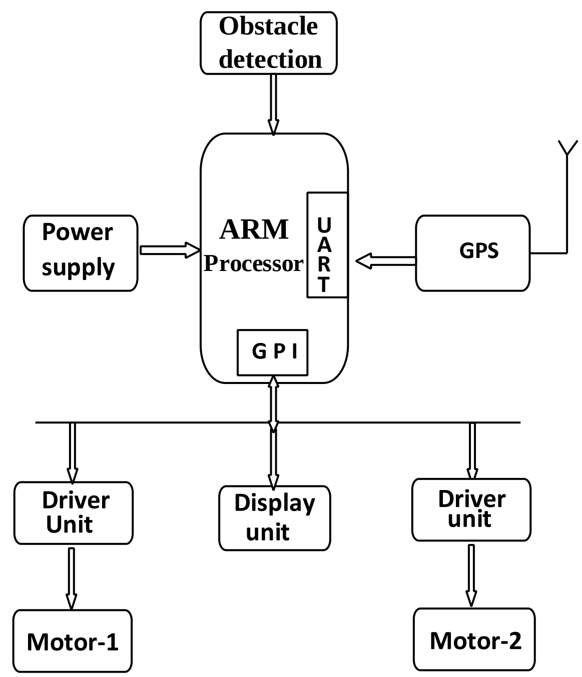

Figure.1. block diagram of Self Driving Vehicle

3.1 WORKING OPERATION:

The project deals with the vehicle navigation using Genetic algorithm. The data transmission time is increased with the protocol standard. One of the sections runs with driver unit and LPC2148 with display unit and another PC as server section runs on a Matlab platform. Communication between two nodes (hardware and application) is accomplished through IEEE 802.15.4. The user can give the source and destination node address to the server section. Using Genetic algorithm [3], the shortest path will be found out and graph plot will be displayed on the server section. Using IEEE standard communication protocol, the shortest path will be feed into the robotic module. Using the path as a reference, the robot moves in the ordered direction. After reaching the destination node, the display unit displays the name (particular place) of the particular node.

4. DIJKSTRA'S ALGORITHM:

Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks. It was conceived by computer scientist Edsger W. Dijkstra in 1956 and published three years later.

The algorithm exists in many variants; Dijkstra's original variant found the shortest path between two nodes,[2] but a more common variant fixes a single node as the "source" node and finds shortest paths from the source to all other nodes in the graph, producing a shortest path tree.

Suppose you would like to find the *shortest path* between two intersections on a city map: a starting point and a destination. Dijkstra's algorithm initially marks the distance (from the starting point) to every other intersection on the map with *infinity*. This is done not to imply there is an infinite distance, but to note that those intersections have not yet been visited; some variants of this method simply leave the intersections' distances unlabeled. Now, at each iteration, select the *current intersection*. For the first iteration, the current intersection will be the starting point, and the distance to it (the intersection's label) will be *zero*. For subsequent iterations (after the first), the current intersection will be the closest unvisited intersection to the starting point (this will be easy to find). From the current intersection, *update* the distance to every unvisited intersection that is directly connected to it. This is done by determining the *sum* of the distance between an unvisited intersection and the value of the current intersection, and relabeling the unvisited intersection with this value (the sum), if it is less than its current value. In effect, the intersection is relabeled if the path to it through the current intersection is shorter than the previously known paths.

To facilitate shortest path identification, in pencil, mark the road with an arrow pointing to the relabeled intersection if you label/relabel it, and erase all others pointing to it. After you have updated the distances to each neighboring intersection, mark the current intersection as *visited*, and select the unvisited intersection with lowest distance (from the starting point) or the lowest label as the current intersection. Nodes marked as visited are labeled with the shortest path from the starting point to it and will not be revisited or returned to.

Continue this process of updating the neighboring intersections with the shortest distances, then marking the current intersection as visited and moving onto the closest unvisited intersection until you have marked the destination as visited. Once you have marked the destination as visited (as is the case with any visited intersection) you have determined the shortest path to it, from the starting point, and can *trace your way back*, following the arrows in reverse; in the algorithm's implementations, this is usually done (after the algorithm has reached the destination node) by following the nodes' parents from the destination node up to the starting node; that's why we also keep track of each node's parent.

5. RESULTS:

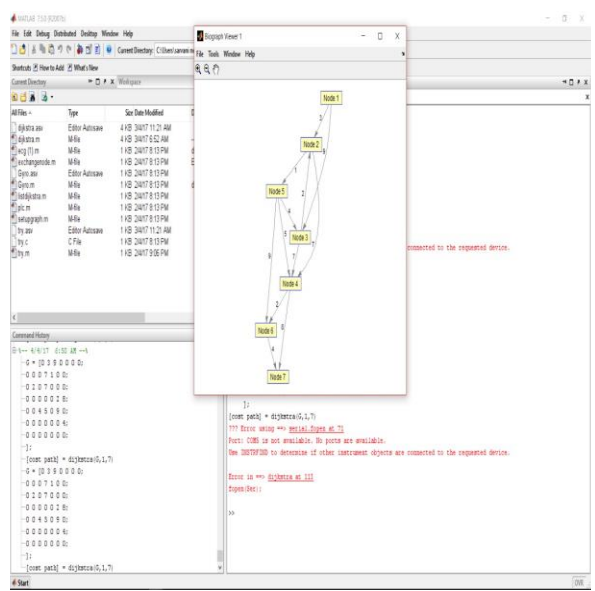

Figure.2. Error content: whenever the rover is not attached to the computer the shortest path is shown but there is an error shown saying that there is no serial connection to the rover.

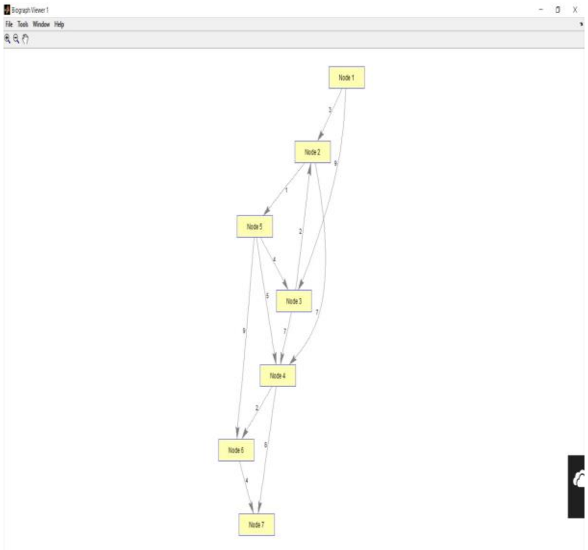

Figure.3. The array designed for the rover. The yellow boxes are the nodes.

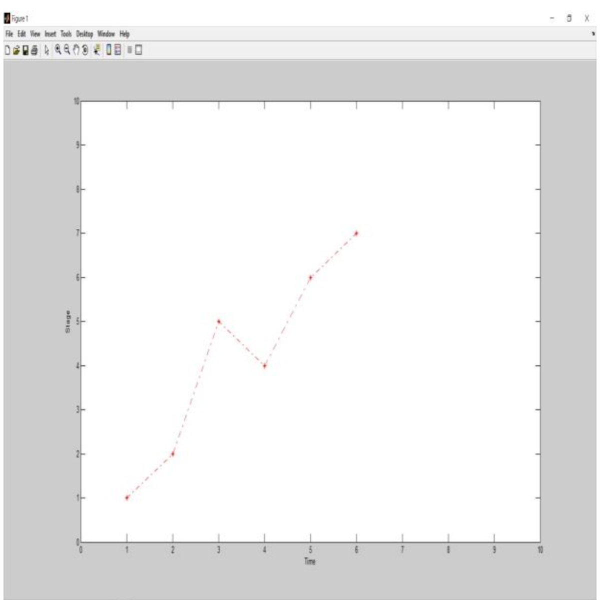

Figure.4. The actual path taken by the rover. The dots represent the nodes.

6. ADVANTAGES:

- ✓ Improved safety and security
- ✓ Simplified system using algorithm
- ✓ Low power consumption

7. FUTURE SCOPE:

The future is very excited to invite artificial intelligence. May it be a smart phone that can talk or a car that drives by itself. All these enhancements are driven by both hardware and software. So, when we can build a car that can drive without any human intervention, we can assume that the advent has gone further than man has ever imagined. This radar can be used in such projects. Google, Audi and many other companies are already working on the driverless cars. As it is the most innovative and research-based project, ultrasonic radars can be used in them. Also, ultrasonic radar has many applications in the field of military and navy. They can be made far more advanced and can be used to detect the trajectory of missiles and can also be used to spy on the enemy warhead. Although ultrasonic radars have a little tradition showing off in their working, they can be used in very intellectual products. As they provide vision to a machine, they are most important. Their advantages over radio-wave radars provide an added advantage to their applications. They are safe and also cost effective.

8. CONCLUSION:

The self-driving vehicles are the future of the modern technology. Google and Audi are still working on them. There are some lapses with the mode

ls that Google has introduced. And we believe that it will come back with more sophisticated and developed features making driving an easy task. This project was just an attempt to show that there is technology that can change and make easy human lives. Dijkstra's algorithm is an old algorithm and this project highlights the fact that old is gold. The path detection is solely through the algorithm whereas the obstacle detection is through the ultrasonic sensor. Therefore, when two or more components work together there are miracles produced. Although Google and Audi use GPS to track and monitor obstacles. The main intention was to point out that ultrasonic sensors are capable of doing it for low power and low cost. Hence, the result is self-driving vehicle using dijkstra's algorithm with obstacle detection using ultrasonic sensor.

9. REFERENCES:

[1] G. Bauzil, M. Briot, and P. Ribes, "A navigation sub-system using ultrasonic sensors for the mobile robot HILARE," in Proc. 1st Int. Conf. Robot Vision and Sensory Controls, Apr. 1981, Stratfordupon-Avon, UK, pp. 41-58 and pp. 681-698.

[2] J. Borenstein andY. Koren, "A mobile platform for nursing robots," IEEE Trans. Ind. Electron., vol. IE-32, pp. 158-165, May 1985.

[3] "Optimal path algorithms for autonomous vehicles," in Proc. 18th CIRP Manufacturing Systems Seminar, June 1986, Stuttgart, Germany.

[4] "Motion control analysis of a mobile robot," Trans. ASME, J. Dynamics, Meas., Contr., vol. 109, no. 2, pp. 73-79, June 1987.

[5] J. A. Boyle, "Robotic steering," Internal Project of the Trent Polytechnic, Nottingham, England, 1982.

