Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 105 (2017) 57 – 61

Procedia Computer Science

2016 IEEE International Symposium on Robotics and Intelligent Sensors, IRIS 2016, 17-20 December 2016, Tokyo, Japan

Maze solving robot with automated obstacle avoidance

Rahul Kumar, Peni Jitoko, Sumeet Kumar, Krishneel Pillay, Pratish Prakash, Asneet Sagar, Ram Singh, Utkal Mehta*

School of Engineering & Physics, Faculty of Science, Technology & Environment, The University of the South Pacific (USP), Suva, Fiji.

Abstract

A quick development of innovation moves us to plan the best choice for an accurate mission. Numerous independent automated innovations are intimated in the lives of individuals making their work much easier. It has been seen that automated vehicles are presented so far, with shrewd abilities after enormous measures of cash spent yearly on the examination. Here in this paper, autonomous maze solving robot is developed with independent mapping and localization skill. Firstly, the maze solving vehicle is designed with three infrared sensors of which two is used for wall detection to avoid collision and the third is for obstacle detection for picking and placing the objects to clear its pathway with the help of robotic arm. Also, it desires to use robot where an environment unreachable for human. In addition, there are also places where use of robots is the only way to achieve a goal. For this, appropriate placement of sensory devices is very critical. We have successfully implemented a maze solving ability onto the robot so called MazeBot. It has been tested that the robot can solve the maze successfully without any interruption with the walls and the objects. In this design, the accuracy of measurements and the real-time processing allied with minimum processing power are the key components in overall embedded design.

Peer-review under responsibility of organizing committee of the 2016 IEEE International Symposium on Robotics and Intelligent Sensors(IRIS 2016).

Keywords Mechatronics; Mazebot; Microcontroller; Maze Solving Robot; Infrared Sensors; Robotic Arm.

1. Introduction

Dense network of city streets, disaster prone regions and war torn environments have common navigational difficulties. These shortcomings range from navigating through obstruction or wreckage on a possible route, navigating around dead ends to figuring out new and complex paths [1]. Specific missions, hazardous surroundings, inhabitable conditions serve as reason for the shift to autonomous technology and decision oriented mechatronics. Another reason to opt for autonomous technology or unmanned vehicles is that these are introduce conservative ways or rescue of survivors due to its impeccable, refined and numerously reviewed decision making and being able to enter a scene or an environment, locate objectives and exit the quickest and most effective way possible whilst eliminating obstructions. The primary worries for this type of technology are processing power, real-time processing and the measurement accuracy [1-3]. It is well-known that the idea of decision making algorithms for applications like maze solving leads an entrance to a goal. Regardless of the challenge of maze solving being close to three centuries old, it still holds a vital stature in robotics. Maze solving is part of the most imperative section of robotics design, which is the Decision Making Algorithm. If a robot is positioned in an unfamiliar environment, it should have a good/exceptional decision making algorithm in order to

*Corresponding author: Tel: (+679) 3232337
Email address: mehta_u@usp.ac.fj

Peer-review under responsibility of organizing committee of the 2016 IEEE International Symposium on Robotics and Intelligent Sensors(IRIS 2016).
doi:10.1016/j.procs.2017.01.192

successfully solve the maze. There has been development in the maze solving or rather decision making algorithms under the micro-mouse algorithm type. The micro-mouse algorithm has developed from wall follower algorithm to flood filling algorithm, where specific algorithms require either full vision of the maze or just mediate portions. It is reported that much needed integration of mechatronics systems is the maze solving robot with a 5 degree of freedom robotic arm for the elimination of obstacles on the robot's path. This is a new feature of a maze robot since previous versions of maze solving robots only concentrated on the navigation and not the physical elimination of obstruction to introduce another possible path to the goal.

In literature, it was demonstrated a maze-solving robot designed to solve a maze, based on the flood-fill algorithm [2], based on Partition-central Algorithm [3]. According to "Seven Bridges of Konigsberg" [7], maze has a mathematical solution to solve a problem that is more a geographical problem. It was explained with an example that there were two Islands in the city of Konigsberg, Prussia which had been connected to the mainland and each other by seven bridges. The challenge set was the possibility if all the seven bridges were able to be crossed only once and eventually return to the starting point. Meanwhile, majority of the algorithms used these days share a close relation to graph theory where there exists a perfect maze or simply a maze without loops which in graph theory equivalence is a tree. Generally, algorithms with relations to graph theory are more superior, efficient, accurate and proficient whilst compared to non-graph theory algorithms when implemented practically on geographic locations. There have been several algorithms utilized in modern day maze solving [4-9] such as wall follower, pledge algorithm, recursive backtracker algorithm and Trémaux's algorithm. In the Wall follower algorithm, either the right hand rule or the left hand rule can be used to solve the maze. This algorithm is very fast and uses no extra memory to solve the maze while having a simple logic of movements. The logic of the wall follower algorithm is observing being in a dark room and finding one's way using the walls of an enclosure and doing this (either with your left or right hand), the solver would eventually make its/his/her way out of the maze. The pledge algorithm is another such algorithm which is similar to the wall follower method and has the feature of solving imperfect mazes i.e. mazes which have loops in them. The pledge algorithm counts the number of turns it makes and if its sum comes to 0 (+1 being right turns and -1 being left turns), when it does come to 0, it leaves the island/loop and continues with using the right hand rule and counting turns. The Recursive backtracker algorithm uses stored coordinates in arrays and does not take the same path again since if it finds a wall ahead or a dead end, it marks it and does not go to those coordinates again but reverses to the last known option or rather detour and continues from there. This algorithm does not necessarily find the shortest path and takes up memory depending how big the maze is. Finally, the Trémaux's algorithm is similar to the previous algorithm (Recursive backtracker) but what this algorithm does is that it does not use memory to store coordinates but physically marks the maze. This algorithm only marks the routes leading to dead ends after locating one. This may ensure least memory used but may use limited materials for marking and may run out of materials like paint or ink.

2. Structure modeling and design

Any mechatronics application consists of four stages: Mechanical, Electrical, Computer and Information Systems. Firstly, it is desired to build mechanical structure and electronics sensor circuit. At the final stage, microcontroller is programmed as per the solving technique adopted. In following, all main tasks are discussed to assemble autonomous maze solving robot. Remember, the aim is not only to solve maze automatically but also to avoid obstacles in the way of propagation.

2.1 Design of overall structure or chassis

The main structure of the system consists of Perspex glass material and the design is circular in shape to incorporate all the components to be seated perfectly onto the base. The main objective is to pick and place the objects on its path and solve the maze successfully. The chassis hosts a robotic arm which is held intact by four screws and protected in-between the two Perspex plates is mounted the Arduino board, motor driver and distribution board. Under the chassis there are housed of two servo motors which control the direction of the fully autonomous robot. The servo rotates the vehicle at any desired direction by using differential drive logic. Fig. 1 shows the basic power platform and chassis for the maze solver robot, namely called here after 'MazeBot'.

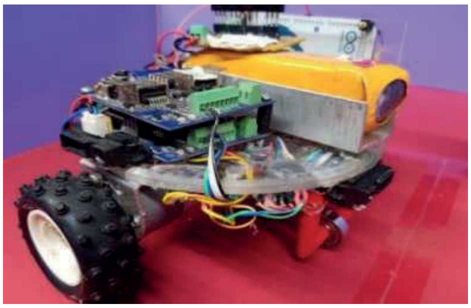

Fig. 1 The basic power platform and chassis for the MazeBot.