International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 1, January- 2013

Autonomous Robot Navigation Based On Reinforcement Algorithm

T Anitha

Abstract

This paper presents the development of a robot NI (single board reconfigurable input output) sbRIO-9631 which navigates autonomously in the unknown dynamic environment based on reinforcement algorithm. The robot will not have the prior knowledge of the environment so this algorithm is used to train the robot in order to reach goal by avoiding obstacle collision.

1. Introduction

Reinforcement learning has no previous knowledge about its working environment. It learns about the environment through interacting with it.

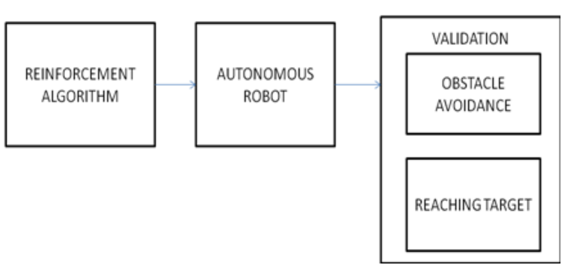


Fig 1: Block diagram

Reinforcement algorithm is used to train the robot to move autonomously in an unknown environment. Reinforcement algorithm teaches the robot to choose the path in order to avoid the obstacles and to reach the target.

The robot surrounding environment consists of different obstacles, some of them are static and the others are dynamic, such as another robot moving in the same indoor environment. The initial robot location and the goal distance are predefined to the robot, where the robot will try to reach the goal with free collision path inspite of the presence of obstacles in the robot's surrounding environment.

The robot environment consists of its goal distance and the obstacles. The first step for applying the Reinforcement algorithm in such environment is to define the robot location [R_x, R_y] , closest obstacle location [O_x, O_y] and target location [T_x,T_y].

The distance between obstacle and robot is calculated by

$$D_{R-O} = \sqrt{(R_x - O_x)^2 + (R_y - O_y)^2}$$

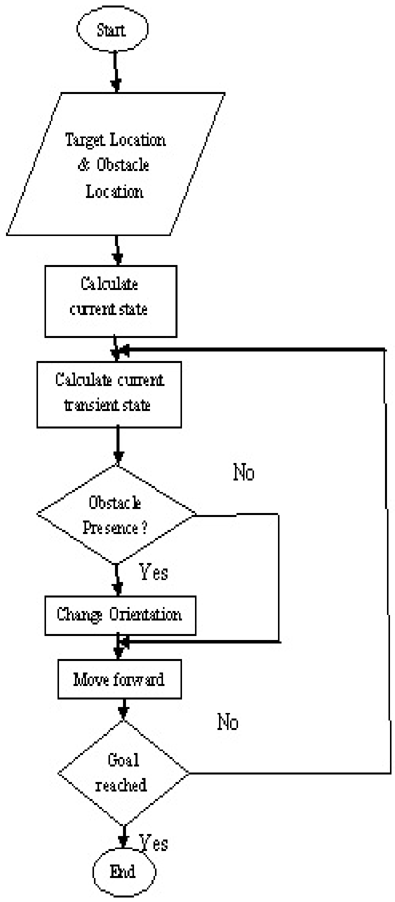
The Target is predefined by assigning a particular distance. The distance between robot and target is calculated by

$$D_{R-T} = \sqrt{(R_x - T_x)^2 + (R_y - T_y)^2}$$

This distance is used to find out, how far the robot has moved from its starting location. When predefined distance is equal to D_{R-T}(distance between robot and target), then the goal has been achieved.

2) Reinforcement Algorithm

1) The obstacle location is supplied to the robot through ultrasonic sensor
2) The target location is supplied to the robot by giving a particular distance.
3) Then the current location of the robot is used to calculate the current state
4) If the robot finds any obstacle within 60cm range then it changes its orientation and moves forward
5) If ultrasonic sensor has not detected any obstacle then it moves forward to reach the goal. After travelling predefined target distance, process is terminated.
6) If the target has not been reached yet, the robot calculates the new current state and repeats the process.



7)
Fig 2: Flow chart of Reinforcement algorithm

3) Hardware Description

The NI Robotics Starter Kit uses NI Single-Board RIO 9631 embedded control platform and an ultrasonic range finder. The Single-Board RIO controller integrates a real-time processor, reconfigurable field-programmable gate array (FPGA), and analog and digital I/O on a single board. It is powered by NI LabVIEW Real-Time and LabVIEW FPGA technologies. The robot has 2 DC motors and 4 wheels. The DC motor for each side of the robot is installed on the front wheels with a 400-tick encoder. Thus, the motor for each side can be controlled independently. The steering method for this robot is called skid-steer.

The robot has a Parallax PING))) ultrasonic sensor that detects objects by emitting a short ultrasonic burst and then listening for the echo. The sensor emits a short 40 kHz (ultrasonic) burst. This burst travels through the air at about 1130 feet per second, hits an object, and then bounces back to the sensor. The PING))) sensor provides an output pulse to the host that terminates when the echo is detected; hence, the width of this pulse corresponds to the distance to the target. This sensor can sense obstacles in a range from 2 cm to 3 m. Moreover, the ultrasonic sensor is installed on a stepper motor. Thus, the ultrasonic sensor can rotate from -90 to +90 degrees. By rotating the ultrasonic sensor, objects around the robot can be detected.

4) Software Description

The sbRIO-9631 devices are programmed using the NI LabVIEW graphical programming language. The real-time processor runs the LabVIEW Real-Time Module on the Wind River VxWorks real-time operating system (RTOS) for extreme reliability and determinism. In addition, it can quickly program the onboard reconfigurable FPGA on sbRIO-9631 using the LabVIEW in FPGA Module for high-speed control, custom I/O timing, and inline signal processing.

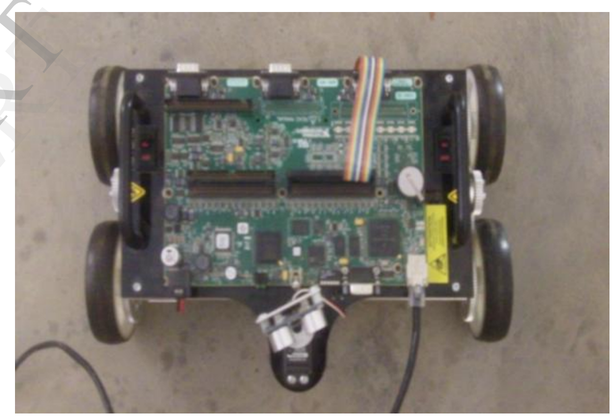


Fig 3: NI sbRIO-9631 Model

5) Simulation Results

The distance of an obstacle is measured using ultrasonic sensor. The sensor output is given to the FPGA kit of sbRIO 9631. The coordinates of the robot , target and obstacle were calculated. The robot is made to move in 3m*3m room in order to check the obstacle avoidance. The reinforcement algorithm is implemented by making the robot to avoid obstacle collision and achieve the goal.

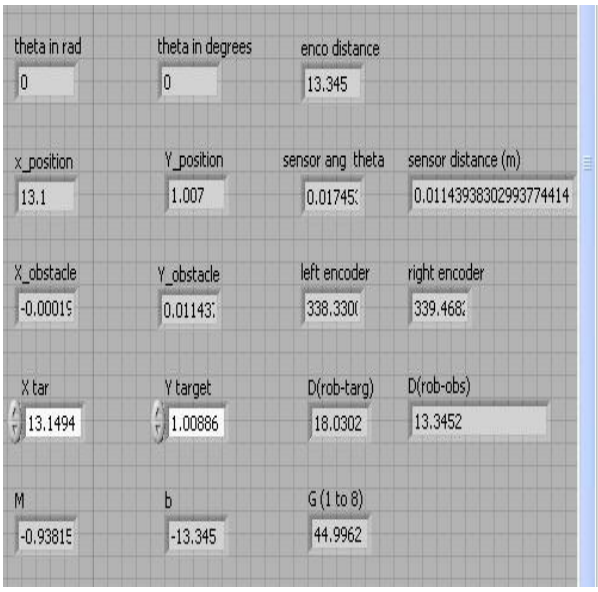


Fig 5: Output shown in LabVIEW front panel

The front panel of LabVIEW 2010 shows the robot location, obstacle location and target location. The distance between the robot and obstacle is measured in meters by using ultrasonic ping sensor which is mounted on top of the sbRIO-9631.

6) Conclusion

The Reinforcement algorithm on sbRIO 9631 has been simulated and implemented for obstacle avoidance in which the Ultrasonic sensor of the sbRIO robot was made to rotate in $+90^0$ to -90^0 and it stops after reaching a particular given distance which is able to avoid the static obstacle collision and reach the target.

7) References

1) Mohammad Abdel Kareem Jaradat, Mohammad Al-Rousan, Lara Quadan "**Reinforcement based mobile robot navigation in dynamic environment**" elsevier journal of robotics and computer-integrated manufacturing vol 27, pg no 135–149 2011.

2) Thomas Kollar and Nicholas Roy "**Trajectory Optimization using Reinforcement Learning for Map Exploration**", The International Journal of Robotics Research Vol. 27, No. 2, pp. 175–196, February 2008.

3) Florent Guenter ∗, Micha Hersch, Sylvain Calinon And Aude Billard "**Reinforcement learning for imitating constrained reaching movements**" Advanced Robotics, Vol. 21, No. 13, pp. 1521–1544 2007.

4) Bing-Qiang Huang1, Guang-Yi Cao1, Min Guo, **"Reinforcement Learning Neural Network To The Problem Of Autonomous Mobile Robot Obstacle Avoidance",** Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005.

5) www.ni.com.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 1, January- 2013