Final Project Report

Collision Avoidance

CECS 574 – Topics in Distributed Computing

Team Members

- Namrata Lomte 016254652
- Nishigandha Shendkar 017412419
- Sarveshwaran Sampathkumar 017387654
- Vandana Santosh 016149937
- Varun Godhani 016215808

Abstract

When we look at the motion planning along with its control in a multirobot scheme collision avoidance seems to be a critical issue. With avoiding collision there is a possibility that it might get into deadlocks. We suggest an idea dealing with motion control in this kind of systems with multirobot. In this paper we proposed the collision avoidance of a robot with another robot. It is very important for the robots to detect an obstacle and avoid collision to continue its work as it is critical for the whole system. In this paper, we are setting a range for all the robots. When a robot comes in the range of another robot it detects the obstacle by using a distributed algorithm which avoids collision as well as deadlock by stopping and resuming the robots repeatedly. In this final project we propose an algorithm which will help the multirobot to avoid the collision.

Introduction

In various sophisticated tasks, multirobot systems are used. There are many possibilities of the robots colliding with each other and this has been common but necessary problem to address. In case of a human- robot workspace if the robot collides with a human, it can cause serious injury to the humans. This is the primary reason for collision avoidance by the robots to provide safety. Human- friendly robots will soon become abundant in use as they can be used for providing services in industries, domestic work. They can even act as service assistants for physically challenging work or even in household. While operating in such environments it becomes prominent to address the problem of collision avoidance. The obstacle should be detected by the robots in an efficient manner to avoid any sort of physical contact in a safe as well as in a robust manner. Robots are also being used for surgery due to the problems occurring in endoscopic surgery. But due to robotic arm collision or collision of the robot with the instrument new problems are being introduced. As a result, collision avoidance is necessary. The algorithm used revolves around the idea of changing robots' trajectories. The other thing is the system consist of fixed path which means it has a predetermined environment.

Motivation

In multirobot system there are many robots operating in common area which may consist of other mobile or moving obstacles. Errors in programming as well as unexpected events in the robot workspace may lead to collisions which may cause various problems. The primary motivation of the paper is to avoid collision and prevent deadlocks in such systems. Robots are being used all over the world as service

assistants for solving complex problems as well as for physical work. While operating in such dynamic environments the tendency of a robot to collide with one another or any unknown obstacle is very high. In case of a stronger obstacle, the robot gets damaged. But if the robot collides with a human it can lead to a serious injury or sometimes even death. Therefore, robots should be designed in such a way that they detect an obstacle and avoid physical contact in a safe manner. Nowadays surgical robots are being used in endoscopic surgeries to avoid certain problems. But their use is giving rise to new problems due to collision of the robot with the endoscopic instrument or the collision of the robotic arms. To reduce these, obstacle detection and collision avoidance is very essential to prevent hazards in critical systems. Driverless car control is a leading technology in today's world. But there should be a well-defined system for the car to trace an obstacle and to avoid it. To prevent the human injuries and deaths that have occurred due to the collision of the cars, efficient obstruction detection and collision avoidance technique is essential. The scenario we considered in the simulation can be considered as a system where the robots are doing surveillance in a certain predefined path and needs to make sure there is no collision and at the same time they do not become stagnant at any point

Problem Statement

Robots move back and forth while moving in a path doing persistent motion, traversing the path they tend to collide with each other. The aim of this project is to prevent the collision of these robots by making them communicate with each other. Motion control of multi robot systems in real-time is an important functionality in prominent systems and collision avoidance can be a critical problem for this concept. Each robot executes the algorithm autonomously and in real-time by checking the state of its next consecutive robots to decide whether it can resume its motion or allow the robot with the higher ID to resume its motion along the predetermined path. The concept of collision avoidance can be further applied to other real time problems of distributed computing like a multi robot system has a capability to accomplish the assigned tasks without colliding with the other robots in its range and following the predetermined and closed path.

Implementation

In the simulation model that we used, there are five robots initially moving vertically and two robots are moving horizontally. Each robot has a configured path (exact distance) and cover specific region. All robots are moving at same speed and are configured similarly. There is a possibility that robots moving vertically would collide with robots moving horizontally at the intersection point of their paths. When robots

start moving along their predefined path, one robot can detect if there is other robot present in its region with the help of message sharing. During the motion of the robots, when they start coming in range of the other robots, the robot check the state of the two next consecutive robots. The robot with smaller ID immediately stops after checking the states and allows the robot with higher ID to pass along its path. Moreover, when the robot with higher ID moves out of the range of the other robots then the robot with smaller ID which stopped earlier resumes its motion and in this way the collision between the robots is avoided as well as it makes sure that none of the robots become dead. The robots keep rotating on their axis and hence their trajectories also change while they avoid collision.

Result

This implementation of collision avoidance was tested on both the simulation environment and on the kilobots. In case of the simulation environment, in order for the bots to collide with each other, we came up with an idea of having a defined path for the bots. By doing so, every bot will move in their own path and it will be easy for us to configure the direction and the position of each bot. Once we finalized the path for the bots, we decided on the number of bots which would be required to show this concept of collision avoidance. We came up with 7, 5 moving vertically and 2 moving horizontally. With this setup we were able to successfully demonstrate the collision avoidance. In case of real kilobots we made use of two bots and we made it to collide those with each other and the bots behaved as expected.

Conclusion

A distributed algorithm is proposed to prevent the collision of robots where each robot has a predetermined path with intersection. In the algorithm, each robot should check before resuming motion whether there is possibility of movement in its direction. It is achieved by frequently halting and restarting their motion. The simulation is performed on multiple robots and a predetermined path is designed for detecting the collision and avoiding it. The simulation results prove the effectiveness of the algorithm performed on five robots.

Future work

The algorithm assumes that the path of each robot is predetermined and only the collisions among the robots are taken into consideration not the external obstacles. The simulation results prove that the algorithm is efficient to avoid the collision among the robots. However, designing an algorithm which can avoid collision between the

external objects and robots is a bit challenging and can be considered as an advanced functionality of collision avoidance. If the decentralized collision avoidance is coupled with some assigned task to robots like collection of data for research will sure raise the number of real time control issues. The main concerns seen during distributed computing need to be evaluated and appropriate measures must be taken. The other issue is related to the stability of the system where in if one of the robots fails, and then the tasks must be rescheduled efficiently.

Reference

Yuan Zhou, Hesuan Hu, Yang Liu, Zuohua Ding, "**Collision and Deadlock Avoidance in Multirobot Systems: A Distributed Approach**", IEEE Transactions on Systems, Man, And Cybernetics: Systems, Vol. 47, No. 7, July 2017.