Thesis
Subtitle

Griswald Brooks

June 6, 2013

Abstract

Real-time path planning is a necessity for any autonomous mobile robot operating in unknown environments. This report describes an implementation of the Game-Theoretic Optimal Deformable Zone with Inertia and Local Approach (GODZILA) path planning algorithm on the Nao humanoid robot by Aldebaran Robotics. The algorithm is a lightweight local path planner that does not require building an environment map. Forward facing ultrasonic distance sensors on the Nao were used for occlusion detection. Theory, simulation, and implementation results will be discussed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Humanoid robots garner much interest not only in the fields of engineering and science, but over a wide range of social and non-technical contexts. The humanoid form makes robots more relatable and accepted by the general public. Socially assistive and service robots need this sort of acceptance in order to perform their jobs more effectively. They also offer unique opportunities for robots to operate in environments which are commonly designed for humans. These environments are typically difficult for many robots to operate in due to things such as narrow corridors, varying heights of door knobs and handles, and stairs. With the ability to traverse a wide range of terrains from rubble to stairs and the dexterity offered by multiple degree-of-freedom manipulators and fingers, humanoids as a platform have a flexibility not seen in many mobile robots. This makes them desirable for operating not only in human structured environments, but unstructured environments such as construction and disaster scenarios.

With such mechanical flexibility come many challenges. Efficient and adaptable gaiting for humanoids is an active and heavily studied area of research, as well as the perception capabilities necessary to take advantage of such robust possible modes of operation. Facial, object, and speech recognition, localization, mapping, path planning, neural and semantic networks are just some of the perception areas researched in order to get humanoids to operate in natural human environments. Aside from the algorithmic challenges of leveraging humanoid platform capabilities, designing platforms with efficient mechanical and electrical systems to accommodate difficult weight and power requirements make this field rife with interesting and novel research possibilities. Vision, structured light, planar laser scanning, series-elastic actuation, and harmonic drive systems are a few of the sensing and actuation topics particularly applicable to humanoid research.

For this project, the Nao humanoid platform by Aldebaran Robotics was used. Nao is approximately two feet tall, with 25 degrees-of-freedom, and various sensors including HD cameras, ultrasonic distance sensors, and inertial sensors. It is easily programmed with a graphical programming interface called Choreographe, or through a framework called NAOqi in which multiple text-based languages such as C++ and Python can be used. Through either API, prebuilt routines such as gaiting and object tracking can be used, allowing for other areas utilizing such commonly needed tasks to be explored, though custom algorithms can be written to supplement or replace the prebuilt ones.

In this project, the topic of path planning was explored. The path planning problem consists of moving a mobile platform or end-effector from a start location to a goal location. Examples vary from ground robots moving through an office building, UAVs navigating to GPS waypoints, or end-effectors avoiding objects on a cluttered table to grasp an object. Methods of solving such a problem depend largely on the construction and knowledge of the environment and the knowledge the robot has about its position within the environment. For example, a mobile robot operating in the plane within a static environment, *a priori* map, and direct position sensing has different challenges than a mobile robot operating in three-dimensions, within an unknown dynamic environment, and only platform velocity estimation. Popular approaches include graph search algorithms such as A* or D*, Bug algorithms, and potential fields approaches. Potential fields algorithms treat the mobile robot as a point mass that is repelled by obstacles and attracted by the goal.

The algorithm implemented was the Game-Theoretic Optimal Deformable Zone with Inertia and Local Approach (GODZILA) algorithm. It is a potential fields style algorithm, with additions that add a form of inertia to reduce limit-cycles and local mimima, a straight-line planner for when the goal is insight, and a goal-randomizer for when the robot is stuck. The ultrasonic sensors on the Nao were used to estimate distance to obstacles, while the prebuilt red ball tracker routine that uses Nao's HD cameras to estimate range and bearing to a 6 cm diameter red ball was used to indicate the location of the goal. Prebuilt gaiting algorithms were used to command the Nao to linear and angular velocities. This allowed the Nao to walk towards a goal with multiple obstacles interfering in its path.

Chapter 2

Path Planning Algorithm

The path planning algorithm implemented in this project was the Game-Theoretic Optimal Deformable Zone with Inertia and Local Approach (GODZILA) algorithm. This potential fields algorithm uses the closed form formulation of an optimized cost function that rewards motion towards a goal location and penalizes motion towards obstacles and in directions other than the current heading. In addition to the optimized cost function, GODZILA incorporates a straight line path planner when the goal is in sight of the robot and a randomization function to allow the robot to escape local minima.

Using the closed form formulation of the cost function in quadratic terms, the algorithm effectively generates forces that push the robot from obstacles and pull it towards the goal. The component penalizing motion in directions other than the current heading of the robot acts like an inertia term which helps prevent oscillations and forces any limit cycles to be larger, which helps eliminate certain local minima.

Occlusion forces are generated through use of sensor data from the robot's distance sensors. Each distance reading is used to generate a force magnitude which is applied as a vector to the robot along the orientation of the sensor relative to the robot. The goal force is generated in the same manner, with estimated distance to the goal generating a magnitude applied to the vector along the estimated orientation of the goal relative to the robot. The inertia term provides a constant bias towards the current heading of the robot.

The angle from the sum of these forces is then taken as the desired heading of the robot and converted into an angular rate with is passed to the vehicle.

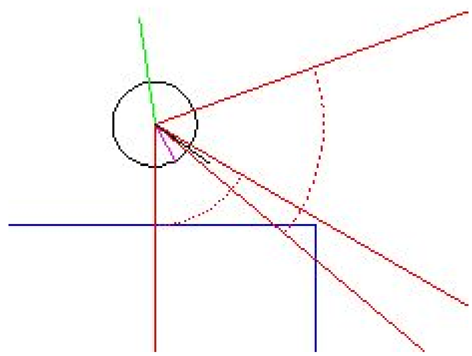

Figure 2.1: Simulation showing component forces. The green line shows the summed occlusion vectors, magenta is the goal vector, black is the inertia vector.

Chapter 3

Platform

Figure 3.1: Nao robot at Control/Robotics Research Lab

The platform utilized on this project was the Nao humanoid experimental robotic platform by Aldebaran Robotics. It is a 25 degree-of-freedom humanoid mobile manipulator with two HD cameras, two ultrasonic distance sensors, four microphones, one three-axis accelerometer, two one-axis gyroscopes, and four force sensors in each foot. The platform is programmed using a framework called NAOqi allowing development using various languages including C++ and Python.

Two ultrasonic sensors in the chest allowed for distance measurements to occlusions. The transmitters are mounted at an angle of 20 degrees from the forward direction of the Nao and the receivers are mounted at 25 degrees. They have a 60 degree viewing angle with a detection range between 0.25 and 2.55 meters with a 1 cm resolution.

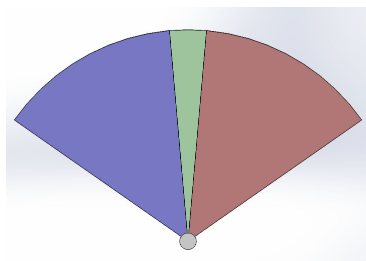

Figure 3.2: Nao sonar cones. The left cone is shown in blue and the right cone is shown in red. The green cone shows the overlap between the two.

As shown in Figure 3.2 the regions covered by the two ultrasonic sensors overlap. As the sensors do not report the angular position of the occlusion within the cone, uncertainty about the location of detected objects is high as they could be anywhere within the sonar cone.

The motion API for the Nao takes three different velocity commands, forward, lateral, and angular, in terms of fraction of maximum step length and step frequency. The maximum step length is 8 cm forward, 16 cm laterally, and 0.523 radians angularly. The maximum step frequency is 2.381 Hz. Due to a stability controller in the built-in gaiting algorithm for the Nao, it takes 0.8 seconds for the robot to react to new commands. At maximum step frequency, this equates to about 2 steps. [1]

Talk about built in red ball tracking for head that returns range and bearing estimates of a red ball 6 cm in diameter.

Chapter 4

Simulation

Algorithm development and robot simulation was done in MATLAB. The simulation constructed a virtual environment using line segments and approximated the robot as a point on the plane. The kinematic model of the robot was taken as a differential drive robot, as modeling the full kinematics of the Nao robot was not necessary to the efficacy of the algorithm. A basic approximation for inertia was made by low pass filtering all velocity commands. Motion noise was injected by adding uniform random noise between zero and one, which was then scaled by a constant, to the linear and angular velocities. The resultant velocities were integrated to gain robot position using RK4 integration. The sonar beams were approximated as cones in the plane with the sensor model returning the distance to the closest object within the region of the cone. These ranges were also perturbed by uniform random noise before being returned to the navigation algorithm.

Chapter 5

Results

Bibliography

[1] Locomotion control. NAO Software 1.14.3 documentation. Retrieved from http://www.aldebaran-robotics.com/documentation/naoqi/motion/control-walk.html#control-walk