.

Inverse Kinematics in Computer Graphics using FABRIK

Christian A. Melendez Gallegos

University of Manitoba

Course: Computer Graphics 2

Course Instructor: John Braico

April 13th, 2019

Robotics has been researched a lot, and, by consequence, advancing during the last decade. Some clear examples of progress in robotics are Atlas, developed by Boston Dynamics, and ASIMO by Honda. Robotics, like any other field of Computer Science, is a topic which looks to solve problems that a human will no be able to solve or do it faster and more accurate than a human. One interesting problem is the avoidance of obstacles in an unknown environment, which is something natural for a human, but a very complex task to achieve in autonomous robotics, but, if an environment would not suit for a human because of its characteristics like danger or size, then an autonomous agent would be needed. In this paper, the concept of behavior-based will be explored, which is one of the methods used to avoid obstacles in an unknown environment. Also, a basic implementation, made by whom is writing, can be found here <https://github.com/chris-mega/BehaviourAI.git>

Behaviour-based robotics combines artificial intelligence, engineering, and cognitive science. It is an activity package, which is a combination of all the abilities necessary to accomplish a goal (Anderson, 2011). For avoiding obstacles, the goals are to prevent collisions with objects in an environment by avoiding them and/or stopping before a collision, arrive at a specific point, to avoid paths that will lead closer to an obstacle and be able to watch where the robot is going (Mataric, 1998).

To prevent collisions, it is necessary to use object detection tools and have some actions to respond to it depending on the situation, like turning in the opposite direction, slow down before the collision, and detect how far away is the object in order to decide which action would be required. The object detection is performed with the help of cameras, ultrasound, infra-red and/or tactile sensor. It is difficult to obtain a precise mathematical model of the robot’s interaction with its environment. The lack of precise and complete knowledge about the environment, limits the application of conventional control systems, therefore, an intelligent control and decision-making system with the ability to reason under uncertainty and learn from experience is needed (Hoffmann, 2003).

After researching throughout all the references of this paper, an algorithm is commonly used to solve the problems stated before. The sensors will provide their readings for the distance between the robot and the walls or obstacles; the bigger the number, the best desirability to choose that direction. As shown in Urrea and Munoz’s research, a decision will be made depending on the combination of sensors with better readings for an optimal choice of path. This reading can include how strong and how close is the source. The entirety of this reasoning is called Fuzzy reasoning, which is the combination of measurements and giving a fuzzy response of related activities together, like moving forward and turning left at the same time if the sensors found a closer obstruction on the right and more space to turn to its left. In addition, other aims can be used depending on the difficulty and the necessity of the environment.

It is often difficult for robots to perceive correctly and learn in a physical world because of the great levels of uncertainty, owing to incomplete and noisy information about a dynamically changing environment. Different methods are added to have a better accuracy and decision making. Reinforcement learning is a popular method for learning in mobile robotics. Adding this approach to the algorithm stated before, behavior-based robots have learned to walk, navigate and divide tasks; there are even some cases when they will compete against other robots in soccer and score goals (Mataric, 1998). Other researches, like Urrea’s implementation, add following trajectories. In this research, the robot tracks a trajectory in crops. If there is an obstacle in the trajectory, the robot will go around it and then go back to the trajectory. Finally, Graves suggests a fusion of teleoperation and the autonomous robot

In the implementation attached to this paper, the algorithm was partially applied. The program uses a simulation of a robot in a region bounded by walls. For now, it can detect a future collision when moving forward. The “sensors” are showed as red lines, which determines the distance between the robot and the wall. If the wall is very far, a default value is given. When the sensor reads a small distance between the obstruction and the robot, it will suddenly stop. This is only one behaviour; the purpose is to not only stop, but to gradually move left or right. Because this is a simulation, a search for walls was needed, but in a real-world example, the algorithm will only require the reading of distance from the sensors. Turning was explored in minimum, but at a certain point the system gets confused because of the different readings. More cases need to be implemented to deal with all the reading better and not having the confusion of where to move. Reinforcement learning and teleoperation were not applied for environment adaptation.

It is safe to say behaviour-based robotics is a very interesting and useful topic to research and implement. It puts together different algorithms based on behaviours for each situation. Other approaches can be added like reinforcement learning or teleoperation. This can be used for any purpose that requires exploring an unknown terrain and be a huge help in areas like rescuing, agriculture, house cleaning, robot soccer competitions and more.

**References**

Aristidou, A., Lasenby, J. (2011). FABRIK: A fast, iterative solver for the Inverse Kinematics problem. *Graphical Models, 73,* 243-260.