Dustbot: Bringing a Vacuum-Cleaner Agent to Life

Debra T. Burhans and Michael Kandefer

Canisius College
Department of Computer Science
2001 Main Street
Buffalo, NY 14208
burhansd@canisius.edu

University at Buffalo
Computer Science and Engineering
224 Bell Hall
Buffalo, NY
mwk3@cse.buffalo.edu

Abstract

A simple vacuum-cleaner agent is introduced in Russell and Norvig's artificial intelligence (AI) text [Russell and Norvig, 2003] to illustrate different agent types to beginning AI students. Underlying the different agent types are several simple actions that take place in a vacuum cleaner world consisting of a grid of squares, some of which contain dirt. The agent's actions include turning left or right, moving forward, and picking up dirt. Having students write a program to simulate the vacuum world is a useful way to provide them with a feeling for different agent types in a simplified environment. However, implementing a vacuum-cleaner agent using a low-cost robotics kit might teach students much more about agents in the real world and could serve to get them interested in and excited about AI in a way that working with purely simulated environments may not. This paper describes the design and implementation of "Dustbot", a robot based on the Russell and Norvig vacuum-cleaner agent. The Dustbot project was carried out as part of an independent study by a student who had already taken the undergraduate AI course. The purpose of the project was to test and debug the vacuum-cleaner robot and to develop a set of instructions that could be used in subsequent offerings of our undergraduate AI course.

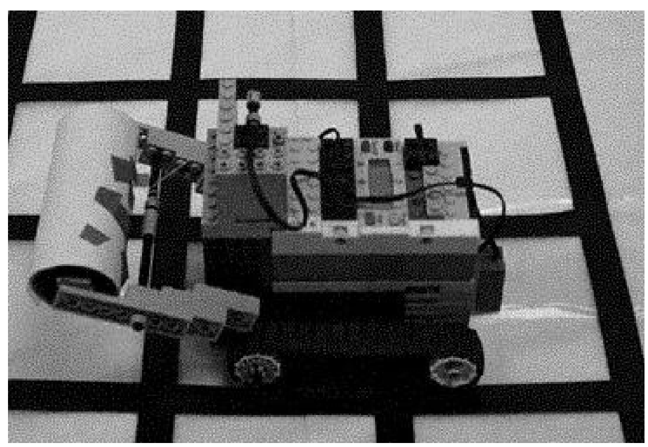
Introduction

The availability of low-cost, relatively simple-to-use robotics kits makes the introduction of robots into the undergraduate computer science curriculum a real possibility, even at relatively small institutions. There are several AI educators with considerable experience in this area [Greenwald, 2001; Greenwald and Kopena, 2002; Kumar and Meeden, 1998]. This paper describes the construction of a robot based on the vacuum-cleaner agent described in Russell and Norvig's introductory AI text [Russell and Norvig, 2003].

A simple vacuum-cleaner agent is introduced by Russell and Norvig to illustrate different agent types to beginning AI students. Underlying the different agent types are several simple actions that take place in a vacuum cleaner world consisting of a grid of squares, some of which contain dirt. The agent's actions include turning left or right, moving forward, and scanning for and picking up dirt. Having students write a program to simulate the vacuum world is a useful way of teaching them about different agent types in a simplified environment.

This impetus for the project described herein was to examine how feasible it might be to build an actual vacuum-cleaner agent using a Lego MindStorms Robotics Invention System 2.0 kit. A goal of the project was to develop a lab that could be used by students in subsequent AI courses. Developing the project outside the classroom allowed us to troubleshoot problems we encountered that would have made our initial ideas untenable in a classroom setting.

The purpose of introducing a robot-building component into an introductory AI course is to stimulate interest in AI and to teach students valuable lessons about what happens when AI is applied in the real world. While our department has not offered AI since the (recent) completion of this project, we hope that our suggestions might lead others to adopt a similar project: we look forward to incorporating it into our next offering of introductory AI.



Dustbot, built by Michael Kandefer

Hardware Specifics

The robot was constructed from a Lego MindStorms Robotics Invention System 2.0 kit which includes an RCX brick as the basic building block for robots. Our initial plan was to mount a small dustbuster on a Lego platform. However, given the limitations of our hardware and the weight of the dustbuster, this proved impossible. We decided instead to mount a lint roller on the front of the robot that could be raised and lowered and rolled over the dirt when it was detected. This proved extremely effective in terms of dirt collection in addition to providing an additional construction challenge in terms of building the mechanism to raise and lower the roller.

The basic construction of the Dustbot uses a modified Diffbot design found in Baum's book [Baum, 2000: p. 233]. This design provides the basic chassis that is capable of moving forward and turning. The lint roller is an additional component that must be added to Diffbot.

The lint roller is spooled around a column of bricks. When placed between two stationary arms, in effect it becomes an elongated wheel and appears like a bulldozer. This would be sufficient if all the robot did was move forward and backward with the roller in a single vertical position. However, the vacuum-cleaner agent needs to decide when to pick up dirt based on its sensory input. Raising and lowering the lint roller was accomplished by using some durable string and attaching it to an axle between the two arms. A motor was then added on top of the robot that winds or unwinds the string. Thus, lowering and raising the lint roller in order to pick up dirt is a simple matter of turning on and off a motor that winds and unwinds a string that is attached to the roller's axle.

The robot includes three motors, one for raising and lowering the lint roller, one for steering, and one for moving forwards. In addition to the construction of the robot it was necessary to build a small "vacuum world" in which the robot would operate. We covered a cafeteria-style table with white paper and then a layer of heavy plastic upon which we created a grid using black electrical tape. Over this we placed another layer of plastic. We built a corral out of wood to prevent the robot from falling off the table.

We used cut-up green construction paper as dirt. There were some problems in getting the sensor to detect the dirt: a dark color of paper or other material is required. The dirt material must be lightweight if using a lint roller.

Sensors

The sole sensor is a light sensor for detecting the dirt. The use of additional sensors would make the robot more realistic, but would require additional programming and hardware construction on the part of students. Additional sensors might be employed, as described below, in order to enable to robot to detect collision with the edges of the corral and to sense the lines in the grid.

Supply List

Lego MindStorms Robotics Invention System 2.0 kit
Additional parts needed for this robot:
  2 16-tooth gears, 1 differential, 1 bevel gear
Electrical Tape
Lint roller
Durable thread
Robot corral
Plastic covering
Sturdy table
Construction paper

Programming

While the Lego MindStorms kit includes a graphical programming environment, it is geared primarily towards those without a background in programming. All programming of the Dustbot was done using NQC (Not Quite C) [http://www.baumfamily.org/nqc]. This simple language is easily picked up by undergraduate students.

The basic functions written for the robot perform the following: move forward, turn right, turn left, scan for dirt, and clean up dirt. The agent type is essentially encoded in the main method for the program. Thus, once the basic functions are in place it is relatively easy to encode different types of agents.

Robot Position

In this initial project we decided to forego line following and edge sensing. Therefore, the Dustbot simply keeps track of its absolute location in the vacuum world. While it initially seemed like this would be straightforward, it took a great deal of trial and error to find the correct constant values for moving forward a certain distance, or turning for a certain amount of time. These values were used to keep track of the robot position and orientation in the world.

Subsequent students might simply make use of the parameters established in this initial project, including the size of the grid and the robot, alleviating the problems of these calculations. Alternatively, two additional sensors could be added to the robot, one to sense the grid lines and one to sense the corral walls. With the addition of these sensors the program would have to change so that it counted grid lines in order to keep track of the robot's position in the world. Either the absolute or relative approach to keeping track of the robot's position should enable students to understand some of the problems that occur in even a simple world whose parameters are fully known to an agent.

Different Agent Types

The simulated vacuum world project used previously in our AI course required students to experiment with several vacuum world agents, including the user-controlled agent and random agent. Different agent types exhibit different types of behavior and varying degrees of autonomy. The following agent types were considered.

User-controlled. This agent is controlled solely by a user and has no autonomy. Implementing this with the Dustbot is simple once students become facile with a remote-control device to control the movements of the robot. This requires the construction of a controller that can be used to manipulate the robot actions. Functionality for the controller includes a button to move the Dustbot forward, a button to turn left, a button to turn right, and a button to pick up dirt. From there it is just a matter of user control to make the robot move through the grid world and clean. Additional considerations for this implementation are constraints on whether or not the user can move the agent out of the grid world.

Random Movement. This agent randomly chooses a direction to move in and does so. Once it reaches its destination square it scans and rolls over the dirt if it is found. Careful planning is required for this agent if line and bump sensors are not employed, and they were not in this project. Thus, this agent must keep track of its current position and direction in the world. Some terminating condition must be established for this agent as well. For example, terminate when the starting or home square is reached or when a certain amount of time is lapsed.

Brute-Force. This agent merely goes through the grid world row by row, moving forward one space scanning and then cleaning if necessary. Once it reaches the end of one row/column it must decide where to turn to go to the next row/column. The terminating condition for this agent occurs when it completes a scan of the entire grid and returns home.

Heuristic. For this implementation the Dustbot goes through the grid analyzing where dirt appears and recording its locations. The Dustbot will then use that information to plot a course through the grid world. A planning agent is considerably more complicated but allows students to experiment with simple planning algorithms. The terminating state for this robot occurs when all the dirt has been collected from the world.

In all of these cases the agents are operating within a simple static world. The goal of the agent is to clean the world completely. One activity that can be added to the robot activity that has been done with the simulated world exercise is to rate the agents in terms of time spent and whether or not they were able to remove all of the dirt from the world before shutting down.

Results

As previously mentioned, this project was carried out by a student working on an independent study on small robot projects. The Dustbot was his final robot. Thus, he had a significant amount of experience using the robot construction kit prior to beginning this project. It was clearly valuable for the student to see part of his AI textbook brought “to life”. In addition, the programming is so simple that it is very easy to implement a number of different agents of varying “intelligence”.

It is unclear how complex actually building the robot would be for a student who has no experience with robotics, but the MindStorms kits are designed specifically for children with no experience with robot building. In addition, most of the instructions for creating this robot are included in Baum [Baum, 2000] and are presented in a detailed manner. Instructions for adding the lint roller have been written up in step-by-step form for the benefit of students who are not familiar with robot building.

An additional suggestion would be to set up a competition among robots designed by different groups to determine which can clean up the world most effectively based on time and amount of dirt left in the world at the end of a run.

References

[Baum, 2000] D. Baum, 2000. *Dave Baum’s Definitive Guide to Lego Mindstorms*, APress, Emeryville, CA.

[Greenwald, 2001] L. Greenwald, 2001. “Tools for effective low-cost robotics”, AAAI Spring Symposium on Robotics and Education, pp. 58-61.

[Greenwald and Kopena, 2002] L. Greenwald and J. Kopena, 2002. “On achieving educational and research goals with small, low-cost robot platforms”. IEEE Robotics and Automation Magazine, Special Issue on Robotics in Education: An Integrated Systems Approach to Teaching.

[Kumar and Meeden, 1998] D. Kumar and L. Meeden, 1998. “A robot laboratory for teaching artificial intelligence”, Proceedings of the 29th SIGCSE Technical Symposium on Computer Science Education (SIGCSE-98), SIGCSE Bulletin, vol. 30(1), pp. 341-344.

[Russell and Norvig, 2003] S. Russell and P. Norvig, 2003. *Artificial Intelligence A Modern Approach*, Prentice Hall, Upper Saddle River, NJ.