# Robot Navigation Using Decision Trees

Erick Swere and David J Mulvaney

***Abstract*—This paper describes ongoing research project on decision tree learning systems and their application to mobile robot navigation. The project will investigate novel decision tree architectures that are able to configure themselves automatically according to the structure of the problem under consideration. To meet the requirements of the mobile robot application, the decision tree learning software will need to learn incrementally, require only the limited memory of an embedded system and operate in real time.**

***Index Terms*—machine learning, decision trees, mobile robot navigation.**

## I. Introduction

DECISION trees represent decision procedures for determining the most suitable output from a range of candidate possibilities. A range of techniques is available that allow them to encapsulate a decision procedure automatically from labelled instances [1]. Decision trees can be applied to many fields such as pattern recognition, classification, decision support system, expert systems [2].

The most popular decision tree algorithm is currently C4.5 [3], but, due to their use of dynamic memory allocation, this and the other available algorithms are not suited to real-time embedded applications. Robot navigation systems need to be able to learn and operate in real time in order to be capable of continual adaptation to changes in the operating environment. This project will design a decision tree system that is suitable for real time application, but that is also capable of learning incrementally and is able to run in the limited memory afforded by embedded systems. The decision tree system will be used in the design of intelligent controller, in this case with the purpose of performing navigation. Robotics covers a wide area of disciplines, including Mechanical Engineering, Electronics and Electrical Engineering and Computer Science. Researchers in these areas are working together with the aim of building a fully functional intelligent mobile robot.

The current research combines machine learning and intelligent mobile robotics. Machine learning is an area of Artificial Intelligence research concerned with developing computational theories of learning processes and building machines that learn. A popular definition of learning itself is a 'change in a system that allows it to perform better the second time on repetition of the same task or on another task drawn from same population' [4]. The structure of the intelligent control system that can be applied in robot system is shown in Figure 1 and includes the following subsystems.

### A. Perception

This is used to get information about the surrounding environments. The information received as a raw data is processed into sensible information to be used by cognition subsystem.

### B. Cognition

This is the heart of the intelligent control system. It is processing information received from perception subsystem and making decision under condition of uncertainty. The decision is produced through learning process. Learning process can be supervised, unsupervised or re –enforcement learning.

### C. Actuator

This part of the control system drives the plant in its environment from one state to another by using the information received from the cognition subsystem. The information received is in terms of what action to take as decided by the cognition subsystem.

The aim of this research project is to investigate novel decision tree architectures that are able to configure themselves automatically according to the structure of the problem under consideration and to demonstrate their feasibility through their application to mobile robot navigation. This is to be achieved by carrying out the following steps.

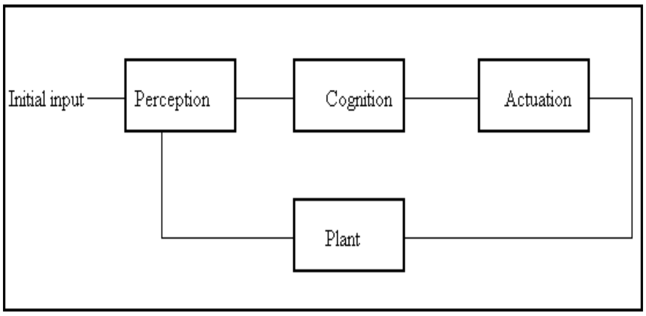


Fig 1. An intelligent control structure

- Designing a decision tree learning system that can perform incremental learning in real time and in the limited memory of an embedded system
- Designing an *evolutionary decision tree* method (one that is able to grow and divide, but also to combine with others)
- Designing and implementing suitable software in C that can be used both in simulated and real

Erick Swere is a Research Student in the Control and Systems Group (e-mail: E.Swere@ lboro.ac.uk).

David Mulvaney is a Senior Lecturer in the Control and Systems Group and the supervisor of this project (e-mail: D.J.Mulvaney@lboro.ac.uk).

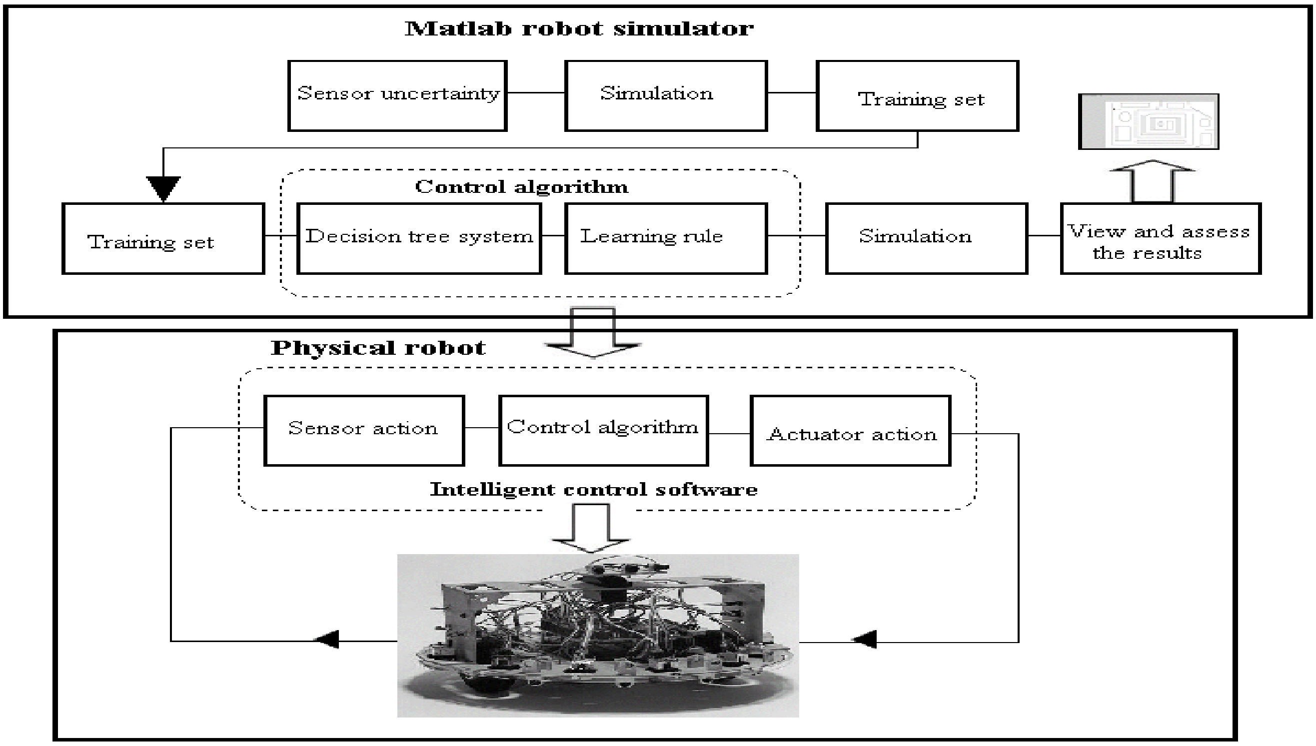


Fig 2. Development environment

environments

- ➢ Demonstrating the learning approach in the navigation of a mobile robot
- ➢ Comparing the performances of candidate decision tree and other learning approaches in their application to mobile robot learning

## II. DEVELOPMENT PROCESS

The overall development process of the intelligent control system is shown in Figure 2. The development process will take place in two stages, first by simulation using the Matlab robot simulator and then by implementation on a physical robot system.

### A. *Matlab robot simulator*

The Matlab robot simulator [5] is used for offline learning during the development process. This simulator runs on Matlab 6.5 under Linux. The process of designing intelligent control by learning involves a considerable number of repetitive operations and using simulation rather than the real robot makes the process shorter and more straightforward to set up. The simulator is using ultrasonic sensors to get the information about the environment.

Initially the robot is allowed to move around the unstructured environment under the demands provided by a very simple controller and the sensors are used to collect information for training. Only useful data are stored for future use and those that appear to improve navigation are applied to the training of the decision tree. These useful data may be repeatedly applied while the robot learns the useful behaviour such as wall following behaviour or obstacle avoidance.

### B. *Physical robot*

It may be expected that the results obtained through simulation should give the same results on the real application. For robotic experiments, this is not always the case. This is because there is high risk of missing the relevant aspects of robot-environment interaction that occur when the robot operates in real environment. Hence, it is important to carry out experiments to confirm results that have been obtained through simulation. This project will use the Talrik II mobile robot [6] for experimentation in a real environment. The Talrik robot is supplied with 12 infrared sensors mounted circumferentially and these will be locally augmented by sonar transmitters and receivers also mounted circumferentially. The controller is being implemented on an ARM evaluator 7T development board running the eCos real-time operating system. As shown in Figure 3, the decision tree system will be called as a separate executable file programmed in C. This means that transferring the

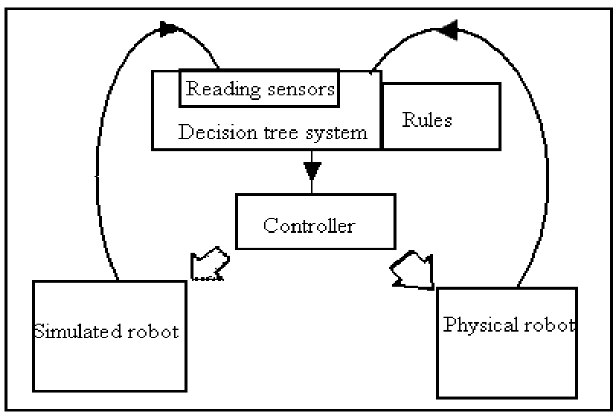


Fig 3. Architecture for the design process

experiment between simulated robot and physical robot will need very little modification. The only main modification needed to move the experiments from the simulated to the physical robot would be to recompile the decision tree system to the ARM target.

## III. Conclusion

The goal of the research described in this paper is to build a decision tree learning system for real-time applications. The decision tree learning system will be able to perform incremental learning in real time and in the limited memory of an embedded system. The decision tree architecture selected must be capable of providing a platform for a reliable, robust robot navigation system that will fulfill the following requirements:

- navigate in unmodified environments;
- cope with unexpected changes;
- achieve the task assigned without requiring any prior information

The next stages of the work will be to implementing the decision tree learning approaches on a physical robot and to learn incrementally in any environment, starting with little or no knowledge of operating environment

## References

[1] J.R. Quilan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81-106, 1986.
[2] L.Atlas, R. Cole, Y. Muthusamy, A. Lippman, J. Connor, D. Park, M.E Sharkawi, and R.J. Marks II, "A performance comparison of trained multi-layer perceptrons and trained classification trees," Proc. IEEE, Vol. 78, No. 10, pp. 1614-1619, October 1990.
[3] J.R. Quinlan, “C4.5: Programming For Machine Learning”, Morgan Kaufmann Publishers, 1992.
[4] H.A Simon, "Why should machines learn?", Machine Learning, An Artificial Intelligence Approach, R.S. Michalski, J.C Carbonell and T.M. Mitchell (Eds.), 1983, ISBN: 0-938382-05-4
[5] Autonomous mobile robotics toolbox, Department of Control, Measurement and Instrumentation, Faculty of Electrical Engineering and Computer Science, Brno University of Technology, Czech Republic, http://wes.feec.vutbr.cz/UAMT/robotics/simulations/amrt/simrobot_en.html.iso-8859-1 .
[6] Talrik II robot, http://www.mrrobot.com/.