CONSIDERATIONS IN ROBOTIC INVASIVE CONTROL: TREE OF HEAVEN[1]

Robert Daley

Department of Computer Science,
University of Pittsburgh,
5401 Sennott Square, Pittsburgh, PA 15260
daley@cs.pitt.edu

ABSTRACT

This paper describes the motivation, premises, and research plan for the Foresbott Project (http://www.cs.pitt.edu/~daley/foresbott/). The goal of the Foresbott Project is the development of autonomous robots that can navigate forests and carry out tasks associated with maintenance of forest health, including detection and control of invasive species, in particular, tree-of-heaven (*Ailanthis altissima*). Briefly, the physical capabilities (hardware) of robots are much farther advanced than the behavioral capabilities (software). The approach to the development of robotic behaviors suitable for forest health tasks is described along with results from earlier work by the author on the co-evolution of robot behaviors for tasks in a different application.

Motivation

The importance of forests cannot be overstated. Yet, the number of assaults on our forests by invasive species is increasing dramatically. In Pennsylvania most of the forestland is privately owned, but the majority of landowners don't address the issue of forest health. Moreover, parcelization of forestland and logging activities often promote the spread of invasives. Robotics offers a solution. Imagine a robot that could navigate a forest, identify a problem (e.g., a tree-of-heaven) and take corrective action (e.g., apply herbicide). Imagine further that it was possible (e.g., through some governmental grants program) for forest landowners to deploy this robot to periodically maintain the health of their forest. This is goal of the Foresbott Project. There are numerous other applications for such a robot were it to be developed, including military surveillance, and search and rescue.

Premises

A great deal of progress has been made in the field of robotics. The physical capabilities (hardware) of robots has advanced dramatically. One example of this is the Big Dog robot developed by Boston Dynamics (http://www.bostondynamics.com/). However, he ability to develop complex behaviors (software) for robots is far less advanced. For example, the DARPA program "Learning Locomotion" (BAA05-25) (http://www.darpa.mil/ipto/programs/ll/) has as it goal the development of control software that will enable a robot (viz., Boston Dynamic's Little Dog) to move across rough terrain. The approach is to develop algorithms that will enable the robot to "learn" this software ("It is expected that the performance of the algorithms developed will far exceed the performance of handcrafted systems, creating a breakthrough in locomotion over extreme terrain."). The Foresbott Project will also develop its robotic behaviors using learning algorithms.

Research Plan

The development of a robot that can navigate forests, identify invasive species and take corrective action is indeed a very ambitious undertaking. It should be understood that this development will take decades to

1 This work supported in part by USDA Forest Service Project FHP-FHTET-TD.01.M01.

complete. The strategy that has been adopted here is to develop those aspects that are specific to forestry applications (navigating in a forest, identifying invasive species, taking corrective action) and to complement that effort with state-of-the-art improvements in other aspects of robotics and artificial intelligence. Rather than undertake the costly effort of building and testing a robot in a forest, we will initially work with simulated robots in a simulated forest environment.

Robot Morphology

Most creatures that navigate forests have legs, sometimes very many of them, so it seems reasonable that the forest robot will be legged. But, how many legs? This will have to be determined as part of the platform development. Likewise, numerous other physical aspects (morphology) of the robot must be determined for the forest robot. Included in these are number and types of joints, size of the robot, lengths of inter-joint segments, sensors and actuators. Each of the three major functional components of the robot (navigation, identification, and corrective action) requires its own set of sensors and actuators. Navigation sensors will include an (active) vision system, sonar, infrared, LIDAR, touch sensors, electronic whiskers, and possibly others. Most of these types of sensors have been studied widely.

Sensors for identifying invasive species are specific to the forest health application and will include a (passive) vision system, an electronic nose (of some type) and a spectrophotometer or spectroradiometer, and possibly others. For example, the tree-of-heaven has a distinctive smell that might be detected by an electronic nose. Multispectral analysis has had some success in identifying tree species from airplanes, but it remains to be determined whether these techniques can be used for identifying tree-of-heaven at ground level. Identification using computer vision is an obvious choice, but in general object recognition using computer vision is very difficult. Considerable experimental effort needs to be made in order to ascertain which combination of sensors will enable to robot to detect invasive species. Finally, the robot will need sensors and actuators (as yet to be determined) that will allow it to take corrective action, e.g., apply herbicide to a tree-of-heaven specimen.

Robot Behaviors

Each of the three major functional components will also need complex control software. This we view as an assemblage of interconnected robot behaviors. Like the DARPA Learning Locomotion Project, we plan to "learn" this assemblage of behaviors. We will be using co-evolutionary algorithms to learn most of the (lower-level) robotic behaviors. To be sure other techniques from artificial intelligence (e.g., planning) will have to be employed also. The particular learning system to be used is the SAMUEL genetic algorithm (GA) system developed by John Grefenstette [4] at the Naval Center for Applied Research in Artificial Intelligence (NCARAI) of the Naval Research Laboratory. In fact we will be using the SAMUEL system to co-evolve both the behaviors and the morphology of the robot. We briefly mention some previous work [2,3] dealing with the co-evolution of robot behaviors using the SAMUEL system.

In [2,3] we built a 2D simulator for the Nomadic (wheeled) Robot and used SAMUEL to co-evolve a complex task consisting of two somewhat conflicting goals of tracking a second robot and periodically docking at a recharging station. We considered several regimes for co-evolving this complex behavior. One, the Monolith regime, consisted of one GA that tried to learn the entire behavior. Our results showed that this complex task could not be learned as a whole (Figure 1 below). We also studied several regimes where the task was decomposed into three behaviors: a Tracking behavior, a Docking behavior, and an Executive behavior that decided periodically which task to perform. Our results showed that for most of these decomposition regimes the complex behavior was successfully learned (Figure 2 below). Our main purpose in these studies was to determine the best regime for the co-evolution of this complex task. The relevant aspects of this work for the Foresbott Project is that indeed complex behaviors, if properly decomposed, can be learned using co-evolutionary methods.

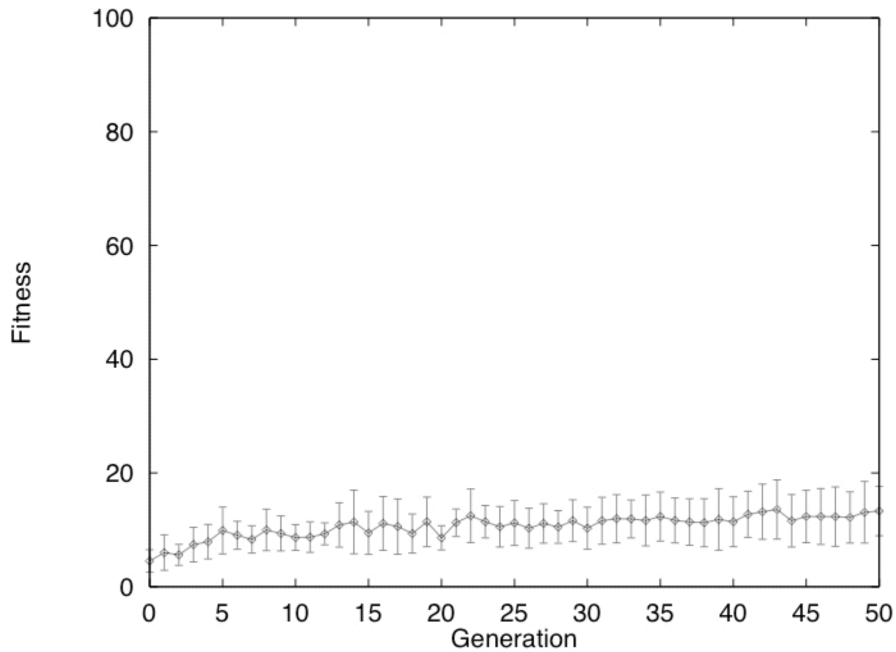

Figure 1. Monolith Learning

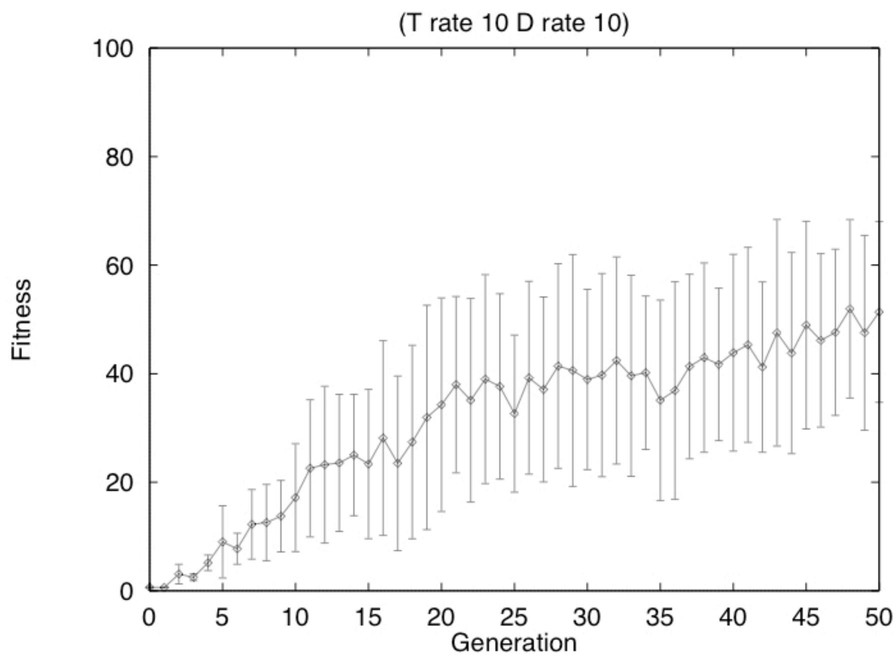

Figure 2. Decomposed Learning

Over the last few years the author has been developing at NCARAI a distributed computation platform for the SAMUEL system that will run on cluster computers and that will allow co-evolutionary methods to be applied to large scale problems, i.e., complex tasks consisting of a large number of interconnected behaviors. The requirements for this platform involve fault-tolerant computing within a network of computers, since there will be many GAs simultaneously and jointly evaluating their populations over an extended period of time. This is critical for the co-evolution of our forest robot behaviors.

Simulation Tools

The initial phases of the Foresbott Project will work with simulated environments. There are many advantages to this approach. First of all, it is much less costly to simulate a robot than to build one and test it (and possibly damage it, and have to repair it). Second, since we plan on learning the robot behaviors, using a real robot would take too long – learning on a simulated robot, where time can be sped up, is the preferred approach. Similarly, we plan to evolve the robot's morphology as well, and "tinkering" with small changes in the robot's physical components would consume an inordinate amount of time. Finally, a robot simulator can "fake" unsolved problems. For example, robotic vision is an extremely hard problem and for a real robot could prevent progress on many of the other behaviors. In a simulated environment, development of the vision component can be postponed until further progress on the robotic vision problem is made. Instead, the simulation's god's eye view can be used to pass information to the behaviors that depend on the vision system.

The real challenge in using a simulated robot is in finding high quality simulators for the robot and its sensors and actuators. For legged robots, a 3D simulator that can perform the necessary kinematics (physics) calculations is essential. Fortunately, such simulators are available and we describe two such here. The first is the Open Dynamics Engine (ODE) (http://www.ode.org). ODE provides the basics for 3D simulations including several types of joints and the corresponding kinematics. In addition, there are several simulation systems that use ODE as their basis but provide simulation software for many of the sensors that are of interest including pan-tilt-zoom (ptz) cameras, sonar, lidar. Included in this list are: Player-Stage (http://sourceforge.net/projects/playerstage), URBI (http://www.urbiforge.com/), and Webots (http://www.cyberbotics.com/).

However, another possible sources of 3D simulators are computer game engines. Of particular promise is the Unreal Tournament Game (UT2004) (http://www.epicgames.com/). The Urban Search And Rescue Simulation (USARSim) [5] (http://sourceforge.net/projects/usarsim), currently being maintained by NIST, is built on top of the UT2004 platform. One important consideration for this approach is that the next version of Unreal Tournament (UT2007) will use the AGEIA physics coprocessor (http://www.ageia.com/). This will considerably speed up the simulations and allows for more complex environments. However, because we will be using the simulator for learning robot behaviors, it is important that the simulation clock can be run at a rate that is faster than real-time. Finally, Microsoft Corporation (http://msdn.microsoft.com/robotics/) has begun development of a robotics development platform that will also use the AGEIA physics coprocessor. One interesting feature of their approach is that their simulations will use software emulation (via the AGEIA PhysX SDK) in case the coprocessor is not installed.

In conclusion, the Foresbott Project is indeed a very ambitious undertaking, but also one that in the long run will be worthwhile.

REFERENCES

1. Daley, R., and J. Grefenstette, "Methods for competitive and cooperative co-evolution", Adaptation, Coevolution amd Learning in Multiagent Systems Workshop, 1996 AAAI Symposium, March 25-27, Stanford University.
2. Daley, R., J. Grefenstette, and A. Schultz, "Co-evolution of robot behaviors", SPIE International Symposium on intelligent Systems and Advanced Manufacturing (ISAM '99), 19-22 Sept, 1999, Boston MA.
3. Daley, R., J. Grefenstette, and A. Schultz, "Methods for Co-Evolution in Multi-Behavior Robots", unpublished manuscript.
4. J. J. Grefenstette and C. L. Ramsey and A. C. Schultz, "Learning sequential decision rules using simulation models and competition", *Machine Learning 5* (4), 355-381.
5. Lewis, M., J. Wang, and S. Hughes (to appear). "USARSim : Simulation for the Study of Human-Robot Interaction", *Journal of Cognitive Engineering and Decision Making*, Santa Monica, CA : HFES.