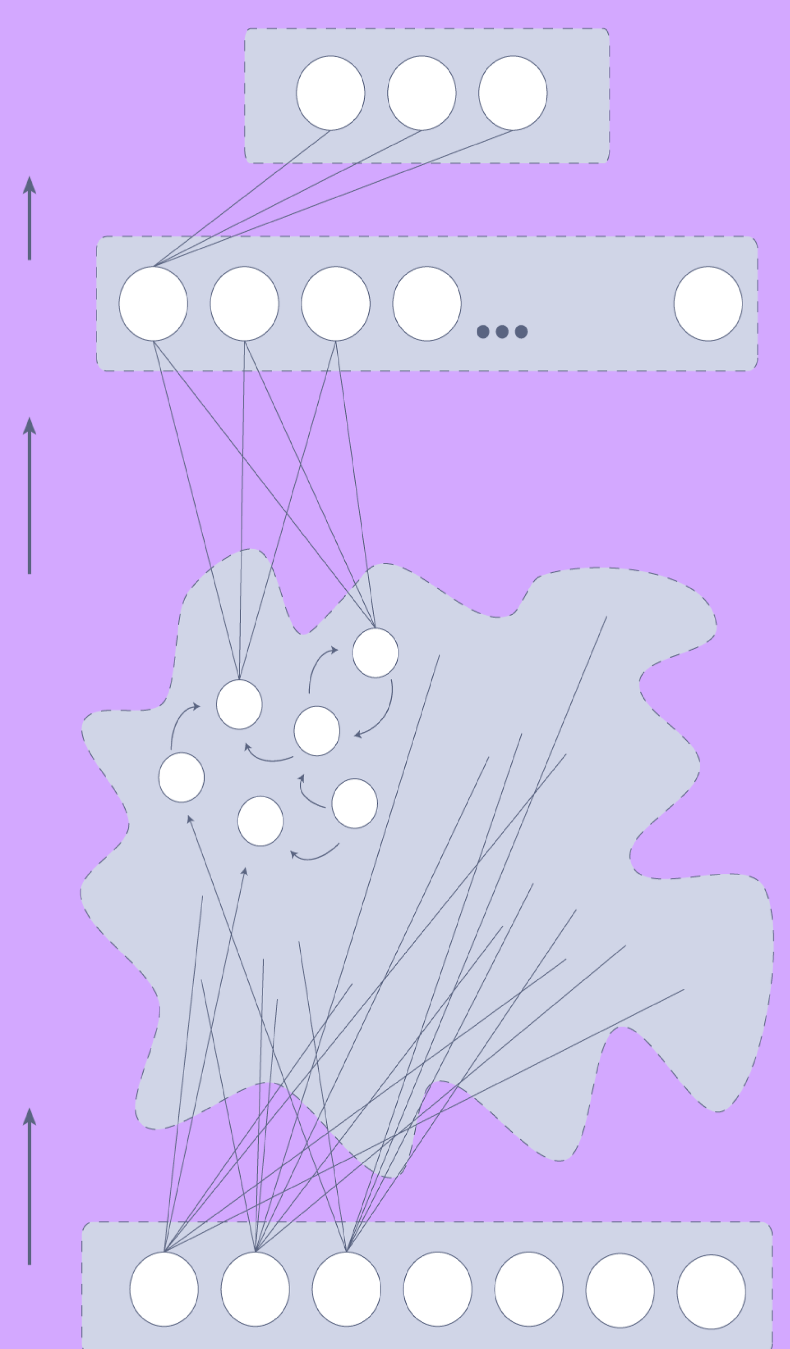
A self-organizing prediction system for autonomous learning in mobile robots

Jacqueline Kory '11, Professor Ken Livingston

Cognitive Science Program, Interdisciplinary Robotics Research Laboratory, Vassar College

This project explores the problem of how to program robots to learn autonomously, without being told how to operate in specific or restricted environments. We have developed a novel algorithm (SOAPS, or Self-Organizing Autonomous Prediction System) based on the idea that adaptive learning depends on being able to predict the consequences of possible actions in an environment and to make choices that achieve the robot's goals. The algorithm is implemented as a partially recursive neural network with the ability to grow new nodes and connections as it learns in real time. This summer we began to move this algorithm from the world of simulation, where it learns simple navigation tasks, to real world robots. The architecture was enhanced by the addition of processing of visual input and by increasing the complexity of goals to be achieved (e.g., seeking out particular locations in the environment while still navigating without collision).

SOAPS Architecture



Behavior Layer

Each node in this layer codes for a particular behavior. Connections to this layer are strengthened when the behavior has a positive result, and weakened when the result is negative.

Memory Layer

Each node in this layer is fully connected to the reservoir layer, and represents a particular pattern of activity in the reservoir. New nodes can adaptively be added to account for new patterns in the reservoir.

Reservoir Layer

The reservoir network is a sparsely connected recurrent network with fixed connection weights. The reservoir provides the network with the ability to discover patterns that occur through time.

Input Layer

Input to the network is coded in this layer. These nodes are randomly and sparsely connected to the reservoir layer.

References and Acknowledgements

References

Josh R. de Leeuw, Kenneth R. Livingston (2009) A Self-Organizing Autonomous Prediction System for Controlling Mobile Robots. Ken Chen, Kamal A.F. Moustafa, Dimitrios A. Karras (Editors) *Proceedings of the International Conference on Automation, Robotics, and Control Systems.* Orlando: ISRST Press, pp. 123-129.

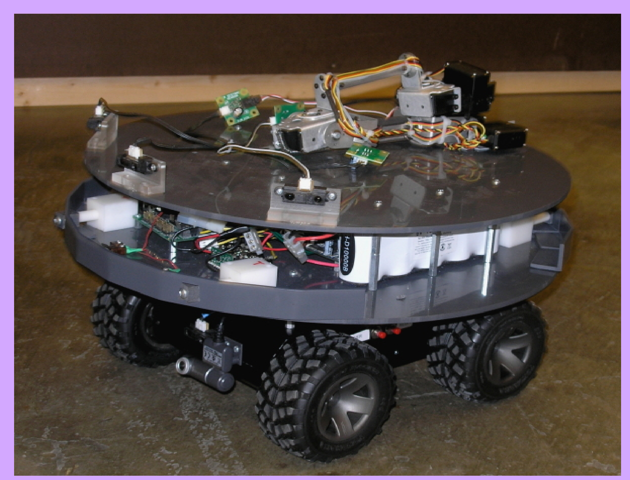
Acknowledgements

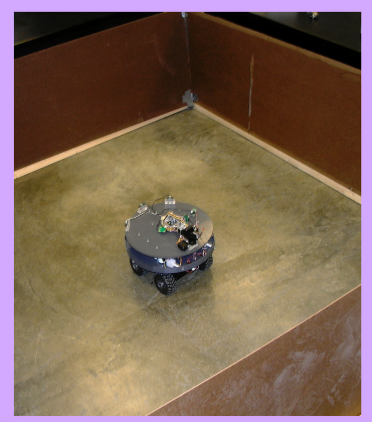
Our thanks to the members of the Interdisciplinary Robotics Research Laboratory, with a special thank you to Associate Director of the IRRL, Josh de Leeuw, for their advice, suggestions, and support.

Learning to Navigate in the Real World

Robot

We previously modeled a small robot that implemented the SOAPS architecture (de Leeuw & Livingston, 2009). This robot used infrared sensors and contact sensors to learn to navigate a simple environment. We ported this experiment to a Corobot (Coroware, Inc.), operating in the real world. The Corobot consists of a computer, contact sensors, infrared sensors, wheel encoders, a gripper arm, and a webcam. In this experiment, only the contact sensors and infrared sensors were used.



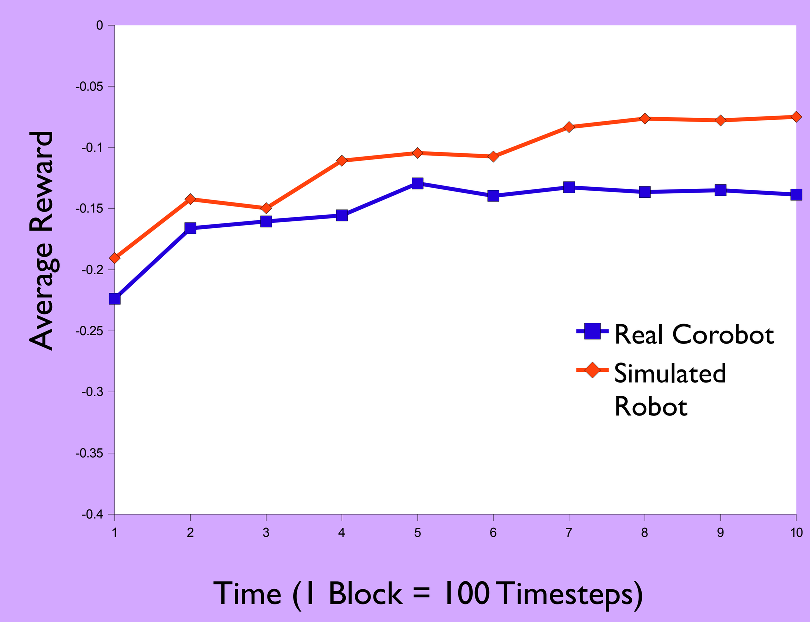


Experiment

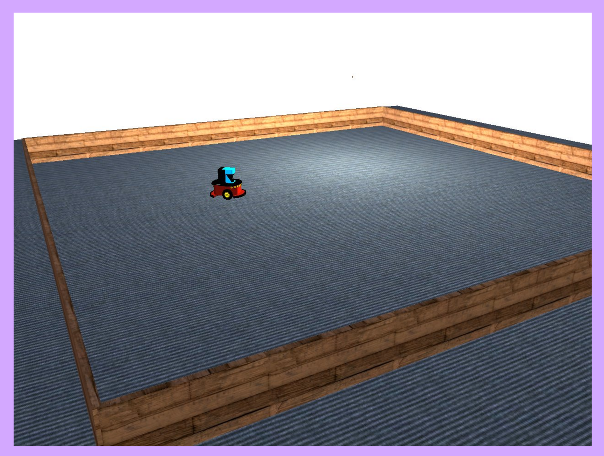
We implemented the SOAPS architecture on the robot. The robot was placed in an 8'x8' open environment. We ran 15 trials with the robot; for each trial, new instantiations of the networks were used, so each trial was a functionally different robot. The robot could move forward or backward, turn left or right, or stop. Moving forward and backward was rewarded slightly more than turning in place, while stopping and colliding were punished.

Results

The average reward in 10 blocks of 100 timesteps was calculated for each trial, then plotted over time. The learning curve we see is consistent with the results obtained in our work in simulation. This provides important evidence that SOAPS is a cross-platform system and works comparably across environments.

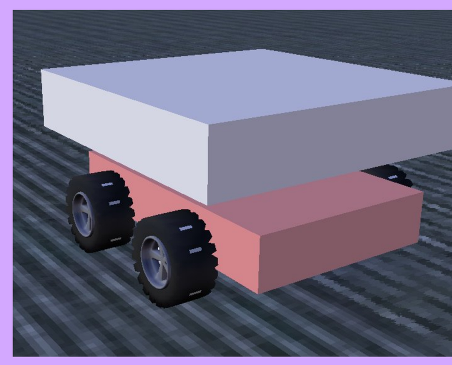


Future Work



SOAPS can be expanded to learn from more information and to perform more complex tasks. We have begun preliminary experiments with a model robot on which multiple networks are implemented. This robot uses input from a laser range finder, photocells, and contact sensors. Its goal is to follow a light gradient to find bright areas while also learning to avoid collisions.

Using a Camera to Navigate

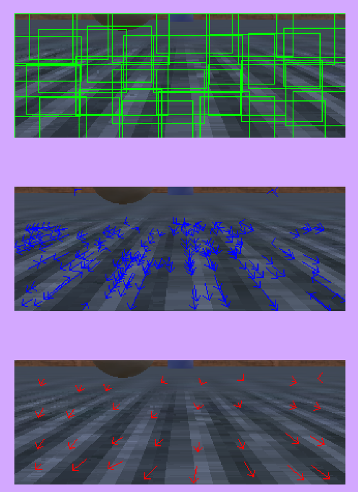


Robot

We modeled a Corobot (Coroware, Inc.) from our lab so that we could compare our results with future work using the robot. The Corobot consists of a computer, contact sensors, infrared sensors, wheel encoders, a gripper arm, and a webcam. In this experiment, only the contact sensors and webcam were used.

Simulation

We modeled the robot using Microsoft Robotics Developer Studio. The model had the same dimensions, wheel placement, and sensor placement as the real robot, and was placed in a complex maze environment with a variety of small obstacles.

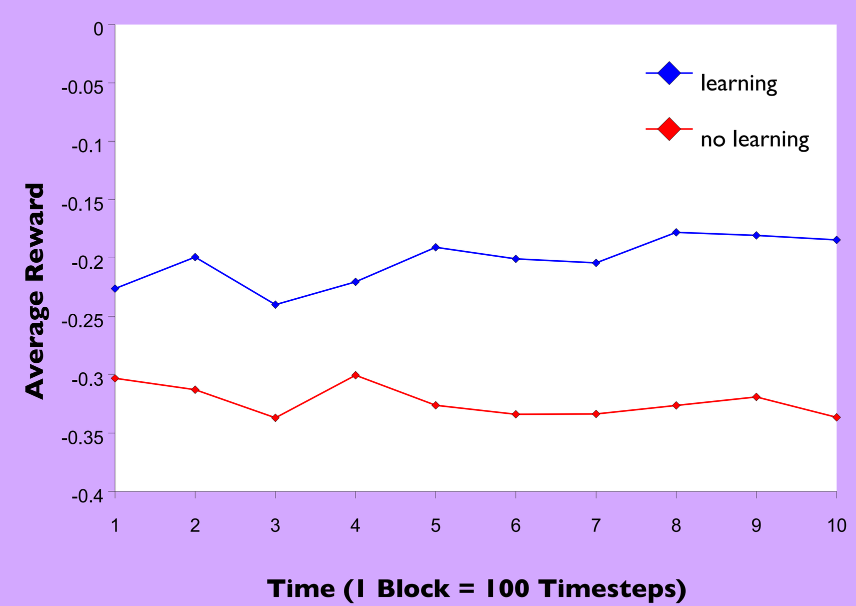


Experiment

We implemented the SOAPS architecture on the robot. Well-known image processing algorithms were used to extract optical flow information from the webcam images. Each image was divided into 32 overlapping regions; the average optical flow vectors from these regions were input to SOAPS.

The first 20 robots ran with the learning architecture in place; the next 20 used randomly selected actions and did not learn. The same movement options were available to the robot with the same rewards as previous experiments.

Results



The average reward in 10 blocks of 100 timesteps was calculated for each trial, then plotted over time. Consistent with previous work, the robot using SOAPS quickly learned to move in a pattern that returned the greatest reward, performing significantly better than the non-learning robot across all 1000 timesteps.