

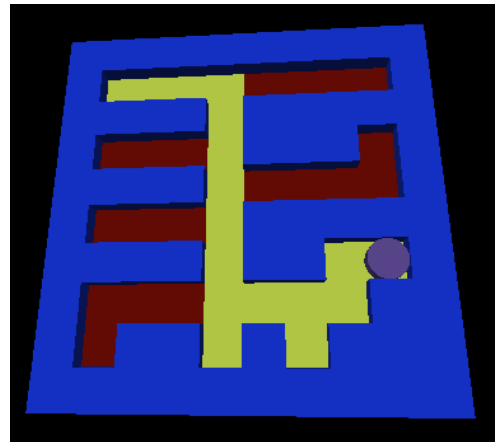
Evolving efficient picobot programs using genetic programming

Benyue Liu, Kaylin Spitz and Ran Libeskind-Hadas

Background: A picobot is a finite-state machine that travels through a given map. In our college's introductory computer science course, students are asked to program a picobot which fills (travels to every empty square in) an empty room and another which fills a maze. Our goal is to automate the creation of these programs.

Approach: We used genetic programming to automate the process. To generate programs, we created isolated "islands" with a set populations, which generally evolved separately but periodically exchanged some or all of their population with each other. Four types of mutation/breeding were used to evolve the programs, and the percentage of each was determined through testing.

Results: Picobot programs were successfully evolved which filled empty rooms very efficiently, and which filled rooms with a single obstacle. No evolved programs were able to fill more than half a maze.



An evolved picobot program filling a maze. Yellow is covered, red is empty space, and blue is walls.

Future work: How can the parameters be changed to encourage the evolution of more fit organisms? Can organisms be evolved which fill mazes? If a picobot is allowed to drop 'pebbles', can more efficient code be evolved?

Funding:
Baker Foundation