Overview: Evolutionary Robotics

An overview of Evolutionary Robotics as written by Stefano Nolfi and Dario Floreano

General Concepts

Behavior Based Robotics - Environmental interaction. Competitive or Cooperative?

Robot Learning - Training on incomplete data

- Explicit instructions
- Reinforcement - Good or bad?

Artificial Life

A Design Perspective

Divide and conquer - Perception, planning, action
Building blocks - build layers upon layers
Distal vs Proximal descriptions of behavior
Example Scenario: Explore, avoid walls, approach target, discriminate target from wall

Examination Methods

Psychology vs. Neuroscience
Connectionism - Total neural network and nervous system
Embodiment - Physical aspects of a system

Evolution Problem

Natural Evolution and reproduction
Survival of the fittest leads to Bootstrap Problem
Solutions:

- More experimenter insight
- Incremental Evolution - Simple to Complex
- Self-Organized Incremental Evolution

The Basics of Genetics

Genetic Algorithms - Artificial chromosomes randomly modified repeated over generations.

Fitness - The higher the better

Selective Reproduction - Roulette wheel

- Rank based, truncation, tournament

Crossover

- One point, multi-point

Mutation

Schema Theory

Schema - Template for a family of strings
1*1 = 101 and 111
N^3 schemata processed (“Implicit Parallelism”)
Significant components written farther apart leads to a higher probability of being broken down

Artificial Evolution in Autonomous Systems

Goal is complex abilities through interaction with environment
Expected to survive on their own
Loose fitness functions for better adaptability

Neural Network

Layers: Input, Hidden/Internal, Output

Feedforward - Signals travel from input to output
Recurrent – Signals may travel within network

Signals travel independently on weighted channels

- Step - output is either 0 or 1 dependent on threshold
- Sigmoid - squashed between 0 and 1 with slope k
- Linear - graded input with slope k

Learning Rates

Supervised Learning - synaptic strengths modified by difference between desired output and output given

Unsupervised Learning - Updates weights based on input value only.

A new learning rate is derived by taking the old weight and adding a new modification weight to it times a small learning rate between 0 and 1

Learning Methods

Hebbian Leaning

- Hebb Rule - When two connected neurons are active, the synapse is strengthened
- Stent-Singer Rule - Based on postsynaptic unit

Supervised Error Based Learning

- Desired network output, changes made based on error rate between desired output and actual output

Reinforcement Learning

- Designed for coarse and sparse feedback
- Maximize the positive.

Learning in Recurrent Networks

- Output depends entirely on input pattern

Justifications For Evolving Neural Networks

- Smoother search space
- Varying evolutionary granularity
- Straightforward mapping from sensors to motor
- Robust to noise
- Biologically plausible
- GAs explore populations of networks, not singular.
- No constraints on type of architecture
- Detailed specifications of network not needed

More Concepts In Learning via Genetic Algorithms

Architecture - Encode blueprint on genotype

- Indirect encoding

Learning Rules - The value of the synaptic connection can be considered as a linear combination of the presynaptic and postsynaptic and current activities weighted by a constant

Lisp and Genetic Programming

Genetic Programming - Encode the solution not the problem

Based on Lisp expressions

(+,2(*,3,2))=2+(3*2)

F={+,-,*,%,IFLTE}

T={X,Y,Z,R}

Above functions and terminals spliced together and mutated over generations.

Overview: Evolutionary Robotics

Questions? Comments?

Hope I was interesting.