An Overview of Core CI Technologies

Lyndon While
Evolutionary Algorithms (EAs)

Reading:
Inspired by Darwinian natural selection:
- population of individuals exists in some environment
- competition for limited resources creates selection pressure: the fitter individuals that are better adapted to the environment are rewarded more than the less fit
- fitter individuals act as parents for the next generation
- offspring inherit properties of their parents via genetic inheritance, typically sexually through crossover with some (small) differences (mutation)
- over time, natural selection drives an increase in fitness

EAs are population-based “generate-and-test” stochastic search algorithms
**Terminology**

- **Gene**: the basic heredity unit in the representation of individuals.
- **Genotype**: the entire genetic makeup of an individual – the set of genes it possess.
- **Phenotype**: the physical manifestation of the genotype in the environment.
- **Fitness**: evaluation of the phenotype at solving the problem of interest.
- Evolution occurs in genotype space based on fitness performance in phenotype space.
The Evolutionary Cycle

Initialisation

Termination

Population

Parent selection

Parents

Genetic variation (mutation and recombination)

Survivor selection

Offspring
There are many different EA variants/flavours:
- differences are mainly cosmetic and often irrelevant
- the similarities dominate the differences

Variations differ predominately on representation:
- genetic algorithms (GAs): binary strings
- evolution strategies (ESs): real-valued vectors
- genetic programming (GP): expression trees
- evolutionary programming (EP): finite state machines

and also on their historical origins/aims, and their emphasis on different variation operators and selection schemes
Best approach for designing an EA:

- choose a representation to suit the problem, ensuring (all) interesting solutions can be represented
- create a fitness function with a useful search gradient
- choose variation operators to suit the representation, being mindful of any search bias
- choose selection operators to ensure efficiency while avoiding premature convergence to local optima
- tune parameters and operators to the specific problem

Need to balance exploration and exploitation:

- variation operations create diversity and novelty
- selection rewards quality and decreases diversity
Neural Networks (NNs)

Reading:
The Nature-Inspired Metaphor

- Inspired by the brain:
  - neurons are structurally simple cells that aggregate and disseminate electrical signals
  - computational power and intelligence emerges from the vast interconnected network of neurons
- NNs act as function approximators or pattern recognisers which learn from observed data

Diagrams taken from a report on neural networks by C. Stergiou and D. Siganos
The Neuron Model

- A neuron combines values via its input and activation functions.
- The bias determines the threshold needed for a "positive" response.
- Single-layer neural networks (perceptrons) can represent only linearly-separable functions.

\[
a_i = g \left( \sum_{j=0}^{n} W_{j,i} \cdot a_j \right)
\]
Multi-Layered Neural Networks

- A network is formed by the connections (links) of many nodes – inputs to outputs through one or more hidden layers.
- Link weights control the behaviour of the function represented by the NN.
  - adjusting the weights changes the encoded function.
Multi-Layered Neural Networks

• Hidden layers increase the “power” of the NN at the cost of extra complexity and training time:
  – perceptrons capture only linearly-separable functions
  – an NN with a single (sufficiently large) hidden layer can represent any continuous function with arbitrary accuracy
  – two hidden layers are needed to represent discontinuous functions

• There are two main types of multi-layered NNs:
  1. **feed-forward**: simple acyclic structure – the stateless encoding allows functions of just its current input
  2. **recurrent**: cyclic feedback loops are allowed – the stateful encoding supports short-term memory
Training Neural Networks

- Training means adjusting link weights to minimise some measure of error (the **cost function**)
  - i.e. learning is an optimisation search in weight space
- Any search algorithm can be used, most commonly gradient descent (**back propagation**)
- Common learning paradigms:
  1. **supervised learning**: training is by comparison with known input/output examples (a training set)
  2. **unsupervised learning**: no *a priori* training set is provided; the system discovers patterns in the input
  3. **reinforcement learning**: training uses environmental feedback to assess the quality of actions
Questions?