Some Definitions

Computation performed via algorithms patterned after biological evolution.

A search procedure that probabilistically applies search operators to a concurrent set of points in the search space.

Copyright Tom Mitchell, 30 November 1995

Biological Evolution

Basically, Lamarck and other biologists had it right:

- Species do "transmute" over time.

What did Darwin and Wallace provide? A mechanism:

- There consistently exists heritable variation among individuals in a population.
- Evolution occurs via a process of natural selection over these variations.

Necessary and sufficient conditions for evolution at some trait:

1. variation across the trait in the population
2. differential survival and reproduction associated with the trait
3. a heritable component to the variation

Mendel and genetics

- elucidation of transposable hereditary elements
- genotype => phenotype mapping

Copyright Tom Mitchell, 30 November 1995

Algorithmic Applications of Evolution

An evolutionary problem solver:

Given a population of candidate solutions

Repeat ad nauseum {

1. Select from the best members of the population for reproduction
2. Reproduce those individuals with variation to form the next population

}

Requirements:

- a population of representations of potential solutions
- perturbation and selection operators that operate on the solutions in the chosen representation
- a fitness function

Canonical application: black-box function optimization

Copyright Tom Mitchell, 30 November 1995

Evolutionary Programming

"Artificial Intelligence Through Simulated Evolution" - Fogel, Owens, and Walsh (’66)

In some sense, the most general of the common sub-paradigms

- no constraints on representation

Chief characteristic: domain-specific mutation operators

Selection: typically probabilistic (tournament), but not necessarily

Evolving finite-state machines

- subject of initial work
  - initial goal was to predict symbol strings generated from Markov processes
- mutation: change output symbol, add a state, delete a state, etc.

Copyright Tom Mitchell, 30 November 1995

Evolutionary Programming cont.

Optimization of continuous functions

- a common application
- population of real-valued solution vectors
- mutation: Gaussian noise added to parameters

TSP

- population of tours in a graph
- mutation: swap adjacent cities in a tour

Copyright Tom Mitchell, 30 November 1995

Evolution Strategies

Rechenberg, Schwefel

- engineering students working on optimization of design parameters
- hit upon an iterative procedure of random mutation and selection
- initially, 1 parent and 1 offspring
- theoretical analysis yielded optimal mutation strategies for a given convergence velocity

Extended to ...

- Multiple parents and offspring (but retaining deterministic selection)
- Crossover
- Self-adaptation of strategy parameters (EP also)

Copyright Tom Mitchell, 30 November 1995

Genetic Algorithms

Holland ’62, ’75

Represent candidate solutions as bitstrings

Employ genetic operators

- principally: crossover
    - k-point, uniform, etc.
- also: mutation
- perhaps on occasion: inversion, duplication

Selection: probabilistic, fitness-proportionate

What kind of search do you get?

Copyright Tom Mitchell, 30 November 1995

Genetic Algorithms cont.

Schemas

- templates defined over {0, 1, *}
- describe a set of bit-strings of definite length (the order)
- defining length - the distance between the leftmost and rightmost defined bits in the schema
- fitness of a bit-string of length l gives some info about the fitness of the 2^l different schemas of which it is an instance
- explicit search through bit-strings can be viewed as an implicit search through schemas

Building Block Hypothesis

GA initially detects biases in low-order schemas, moving towards those of higher-order over time by combining lower-order schemas via crossover

Schema Theorem ...

... shows that the expected number of samples of the observed best schemas increases exponentially in time

Better things - fuzzy GA's

- looser chromosomal linkage

Copyright Tom Mitchell, 30 November 1995

Discussion

What do these things do?

- they run for a while and then stop
- where they stop depends on both the representation and operators

Representation

- quasi-Phenotypic (EP/ES) vs. quasi-Genotypic (GA)
- each paradigm argues for advantage

Copyright Tom Mitchell, 30 November 1995

The Cross-Over Controversy

Pro

- provides for faster evolution
- provides for "non-local" moves - an escape from local optima
- building block hypothesis, schema theorem
- sexual reproduction is ubiquitous in nature

Con

- premature convergence
- breaks up co-adapted traits
- still doing hill-climbing, but just in a re-mapped space
- the adaptive value of sexual reproduction in nature is actually not very well understood

The value of cross-over is based on the assumption that "two great tastes taste great together"

- sometimes true, sometimes not

Copyright Tom Mitchell, 30 November 1995

Discussion cont.

Bias of evolutionary methods:

> towards those solutions most easily found by hill-climbing on the surface that results from the choice of representation and operators

Is evolution a stupid thing to do?

Probably, if an acceptable, more direct method is available

However, there will always be problems of interest lacking direct solution methods

Moreover, these problems are likely to be THE most interesting problems

Copyright Tom Mitchell, 30 November 1995

Biological vs. Artificial Evolution

Are we capturing the essence of biological evolution?

Yes, in the sense that evolution is really a simple idea.

Yes, in the sense that evolving artificial populations do exhibit some basic characteristics of natural evolving populations.

Less so in the following senses ...

- continuous vs. discrete (like the central nervous system vs. ANN's)
- complexity (also like CNS vs. ANN's)
- co-evolutionary context
- open-endedness

Copyright Tom Mitchell, 30 November 1995

Reasons for popularity

Intuitive appeal

Only needs "black-box" function

Parallelizable

Copyright Tom Mitchell, 30 November 1995

Evolution-based Machine Learning

What is it?

- weak sense - it's all ML
- stronger sense - evolutionary search for a hypothesis, something that will generalize to or interact successfully with the unseen

Copyright Tom Mitchell, 30 November 1995

Evolution and ANN's

Evolving neural net weight vectors

- should do better on perverse surfaces
- basically more open-minded ...
- ... but way-stupid (read slow) relative to backprop
- ... and potentially vulnerable to other local minima which backprop would never find
- think about how operators apply to this: crossover, etc.

Perhaps more to the point: evolving topology

- more intriguing, because other approaches are limited
- do backprop afterwards
- can be full connectivity matrix, or higher level description
- can be full networks, or sub-units
- think about what the operators do: completeness? correctness? level of operation?

Copyright Tom Mitchell, 30 November 1995

Evolution and ANN's cont.

Other things to evolve

- pruning connections
- learning rate, momentum
- get backprop out of local minima
- other way around: evolve a reasonable starting weight vector, then do backprop
- evolve a learning rule

Copyright Tom Mitchell, 30 November 1995

Evolution and Rule-based Systems

Individuals as Rule-sets: Samuel (Greffenstette et al)

- Symbolic if-then rules

Classifier systems, Animats

- Booker
- Wilson
- recurrency, internal symbols via message list

Michigan approach - Holland

- classifiers as members of the population
- need way of assigning credit (e.g. Bucket Brigade algorithm)

"Pitt" approach - De Jong (Pitt), Smith (now CMU RI)

- sets of classifiers as members of the population
- sidesteps credit apportionment issue
- usually needs more trials for optimization

Copyright Tom Mitchell, 30 November 1995

Evolution and Rule-based Systems cont.

Concept learning via disjunctive rule sets

- Cogin - Steve Smith
- Rules as bit-strings, population forms disjunction
- Fitness calculated from information gain, misclassifications (tradeoff between simplicity and accuracy)
- Successors generated from crossover from current population, sorted by fitness, and cut-off when instance space is covered

Copyright Tom Mitchell, 30 November 1995

Genetic Programming

Basic idea: individuals are programs in a functional language

- initialize population to random function "trees"
- composed of terminals and non-terminals
- now, can do cross-over between sub-trees
- mutation also possible, a la EP
- extension to other languages

Copyright Tom Mitchell, 30 November 1995

Some Comments on Evolution-based ML

Hard to get a grip on this stuff - large systems, many tweaks, nonlinear behavior

The three methods presented here are more-or-less graph-structured

- Generalized graph perturbation operators (e.g. swap/duplicate subtrees, change node, change connection strength, etc.) might be applied to elucidate general properties . . .

Copyright Tom Mitchell, 30 November 1995