

Lecture 15: Other Areas of Evolutionary Computing

- ▶ We have already covered a few areas of EC in some depth
 - ▶ Genetic Algorithms
 - ▶ Practice
 - ▶ Theory
 - ▶ Genetic Programming
 - ▶ Now we'll briefly look at some other areas and applications of EC
 - ▶ Memetic Algorithms
 - ▶ Evolutionary Strategies
 - ▶ Estimation of Distribution Algorithms
 - ▶ EC in Machine Learning and Modelling

- ▶ *Memetic algorithms* combine EC (typically a GA) with local-search heuristics
- ▶ The name derives from Richard Dawkin's coining of the term *memes*, referring to cultural replicators
 - ▶ An ongoing dispute in the early days of evolutionary theory was whether adaptations acquired during an individual's lifetime could be inherited by their offspring
 - ▶ E.g. the strong arms developed by a blacksmith would be passed on to his children
 - ▶ This was ultimately resolved in the negative by the *central dogma* of molecular biology
 - ▶ Information can pass from DNA to protein, but not vice-versa
 - ▶ In Dawkins' ideas, memes are ideas that spread, and ideas can of course be changed into 'fitter' ideas by the experiences of their bearers, and then promulgated

Memetic Algorithms

- ▶ Anyway, that's all a historical footnote...
- ▶ *Memetic algorithms* combine EC (typically a GA) with local-search heuristics
 - ▶ E.g. we might produce offspring through selection, crossover and mutation, then use local search of some other heuristic to improve those offspring
 - ▶ This is related to the idea of problem-specific repair heuristics in constrained optimisation problems
 - ▶ Similar questions are raised, such as whether offspring fitness should be based on original or improved quality of the solution
- ▶ Memetic algorithms are also often called *hybrid* algorithms, e.g. hybrid genetic algorithms

Estimation of Distribution Algorithms

- ▶ The expressiveness of our model will affect the complexity of probability distributions we can approximate
 - ▶ The simplest model is to assume that each locus is independent
 - ▶ Then the probability of each allele occurring in a high-fitness solution according to our model will depend only on the frequency with which that allele is observed in high-fitness solutions sampled
 - ▶ Such models will clearly fail to approximate epistatic problems well
 - ▶ We can increase the degree of non-independence we consider in our model

Evolutionary Strategies

- ▶ We have already come across *evolutionary strategies* in the context of selection operators
 - ▶ (λ, μ) and $(\lambda + \mu)$, etc.
 - ▶ See the 'Advanced Operators and Techniques' lecture for further details
- ▶ The defining difference of ES compared to other EC approaches such as GAs is the representation...
 - ▶ GAs (for example) used discrete, often binary encodings
 - ▶ ES use real-valued vectors
- ▶ ...and the operators
 - ▶ Mutation is performed by adding gaussian noise to an element in the vector
 - ▶ The variance of this noise can be adapted on a locus-by-locus basis during the optimisation process
 - ▶ Crossover and selection can be performed as normal
- ▶ Now we are getting very close to a 'representationless' algorithm
 - ▶ Particularly if we use a crossover operator with no positional bias, such as UX

Estimation of Distribution Algorithms

- ▶ An EC algorithm generates a random population, then performs a biased search in the set of possible problems, based on the solutions represented in the population, and their objective values
- ▶ An EC algorithm can thus be viewed as sampling points in the search space, according to an *unknown probability distribution*
 - ▶ The population and the selection, crossover and mutation operators are thus tools by which this distribution is sampled from
- ▶ *Evolution of Distribution Algorithms* dispense with the operators, which indirectly sample from the distribution, and directly choose points for evaluation in the search space according to this underlying probability distribution
- ▶ To sample from this distribution, however, we must know it first
 - ▶ Knowing the distribution implies total knowledge of the problem, in which case we don't need to apply an optimisation algorithm
 - ▶ We must estimate the distribution and construct a model, but even gaining a realistic estimate will be non-trivial