

# Glossary

©2003 by Dan Ashlock

**Argument.** An argument of a function is one of its input values. The argument of a node in a parse tree is one of the subtrees that provides input to it or the root node of such a subtree.

**Atomic.** Atomic is a synonym for indivisible. In an EC context we call the smallest elements of a chromosome that cannot be cut by whatever crossover operators are being used the *atomic* elements of the representation of that chromosome. Varying which elements are atomic can substantially change the behavior of a system.

**Cellular encoding.** Cellular encoding is the process of using directions about how to build a structure instead of the structure itself as the chromosome type in an evolving population. Cellular encodings are indirect, and come in many varieties.

**Chromosome** The chromosome type of an evolutionary computation system is the data structure used to store each member of the evolving population. The string evolver uses a string chromosome.

**Connection weights.** Connection weights are numbers associated with the connections between neurons in a neural net. Adjusting these weights is the primary method of programming a neural net.

**Copy number.** When we allow a solution to reproduce, its copy number is the number of copies we make.

**Crossover.** A variation operator that randomly blends parts of two structures to make one or more new structures is typically called a crossover operator.

**Cycle Type, of a Permutation.** An unordered list of the lengths of cycles in a permutation if it is given in cycle notation. See Chapter 7 for a definition of cycle notation. The cycle

type is a list of positive integers adding to  $n$ , e.g. the cycle type of  $(0\ 1\ 2)(3\ 4)(5\ 6)$  is  $3\ 2\ 2$ .

**Deceptive Fitness Function.** A fitness function is *deceptive* if, outside of a small neighborhood of the global optima, movement toward the optima actually reduces fitness.

**Direct encoding.** Using a direct encoding means to store, as the structures you are evolving, exactly the structures used by your fitness function.

**Discontinuity.** A discontinuity in a function is a place where a function makes an abrupt jump in value. A standard example of such a function is the cost of a package as a function of its weight. The cost jumps abruptly at certain weights.

**Finite State Machine.** A finite state machine is a device that takes inputs and looks up appropriate outputs from an internal table. It also stores an internal state which lets it look up different outputs depending on the history thus far. This internal state is a form of memory. The outputs may either be associated with the transitions to a next state or may be associated with the states themselves. These two different types of finite state machines are called Mealey and Moore machines, respectively.

**Fitness biased reproduction.** This is the analog of “survival of the fittest” in evolutionary computation. It is the practice of making better solutions more likely to reproduce.

**Fitness Function.** A fitness function is a heuristic measure of the quality of solutions. The fitness function is used to tell which solutions are better than others.

**Fitness Landscape.** The graph of the fitness function. The evolving population can be thought of as moving on this landscape. This landscape metaphor is useful for thinking about evolutionary computation.

**Global Optima.** A global optima is an optima that takes on the maximum possible fitness value. It need not be unique, but multiple global optima must take on the same fitness value. See also: optima, local optima.

**Indirect encoding.** Using an indirect encoding means the objects stored in the evolving population are interpreted or developed before they are passed to your fitness function.

**Lexical Fitness.** The practice of using a lexical partner fitness function.

**Lexical Partner.** A second fitness function used to break ties in fitness values is a *lexical partner* for the fitness function it is supposed to aid. The name comes from lexical or

dictionary ordering. Since the second fitness function, the lexical partner, can only break ties it is infinitely less important than the first fitness function, just as the first letter of a word is infinitely more important than the second in determining its position in an alphabetical list.

**Local Optima.** A local optima is an optima that does not take on the maximum possible fitness value. See also: optima, global optima.

**Mating Event** A single act of reproduction. In an algorithm that uses crossover a mating event is the selection of two parents, the process of copying them, possibly crossing the copies over, mutating the copies, and then possibly placing the copies back in the population.

**Mutation.** A variation operator that makes random changes in a single structure in the population is called a mutation.

**Neural net.** A neural net is a network of connected neurons. The connections have associated numbers called weights. These weights establish the strength of the connections and, together with the pattern of connections, control the behavior of the net. Neural nets are programmed by adjusting the strengths of the connections.

**Neuron.** Neurons are the units out of which neural nets are built. A neuron is connected to other neurons with weights, accepting the sum of the weights times their outputs as its input. The neuron also has a transfer function that transforms its input into its output.

**Node.** A node is an operation or terminal in a parse tree. Both operations and terminals are stored in the same sort of structure and are distinguished by the fact that operations take arguments (inputs) while terminals do not.

**Operation.** An operation is a node in a parse tree that both accepts and returns values.

**Optima.** An optima, of a fitness function or fitness landscape, is a point with a better fitness value than all the other points near to it. See also: local optima, global optima.

**Parse tree.** A parse tree is a dynamically allocated data structure, composed of individual nodes, that can store mathematical or symbolic formulae. The basic data structure stores a single operation, constant, or input value.

**Perato-frontier.** The set of all perato-optimal objects are the perato-frontier for a problem with two or more quality measures. The frontier exhibits the tradeoffs between the quality measures. See: perato-optimal.

**Perato-optimal.** If we are comparing objects with two or more quality measures then one dominates another if it is better in all quality measures. A strategy that cannot be dominated is said to be perato-optimal.

**Penalty function.** A penalty function gives a valuation to the violation of some condition or rule. Penalty functions are used to build up or modify fitness function by reducing the fitness of a member of a population by some function of the number of undesirable features it has.

**Phase Change.** A phase change is a frontier where the character of some quantity changes. The standard example of distinct phases are the ice, liquid, and steam phases of water. In a function a phase change is an abrupt change in some characterization of the function's behavior, e.g. a function that oscillated for positive inputs and remained constant for negative inputs might be said to have a phase change at zero.

**Population.** The population is the collection of solutions on which an EC-system operates. The term is draw from biology.

**Population Seeding.** An ordinary evolutionary algorithm generates an initial problem at random. Population seeding is the practice of adding superior genes to the initial population. These genes can be from previous evolution, designed according to heuristics, or created with expert knowledge. The fraction os seeds may vary from a few to entire population.

**Representation.** The method of representing potential solutions or of coding structures you intend to evolve is your representation. Changing the representation can completely transform system behavior and so choice of representation is critical.

**Root.** The root or root node of a parse tree is the topmost node in the parse tree. Its output is the output of the tree.

**Shortest path problem.** This is a problem of finding the shortest path connecting two points. In this text we explore a very simple version of this problem, finding a path from  $(0, 0)$  to  $(1, 1)$  across a featureless landscape.

**Subtree.** A subtree (of a parse tree) is a tree inside that parse tree, rooted at one of its nodes. Strictly speaking a whole parse tree is a subtree rooted at the root node of the tree. Subtrees that are not the whole tree are called *proper* subtrees.

**Subtree crossover.** Subtree crossover is a form of crossover used on parse trees. A node in each parse tree is chosen and the subtrees rooted at those nodes are exchanged.

**Symbol.** A type of very simple virtual robot.

**Terminal.** An terminal is a node in a parse tree that returns a value. It may be a constant or a means of passing valuse to the parse tree from outside.

**Variation operators.** An umbrella term for operations such as crossover and mutation that produce variations of members of an evolving population. A variation operator is called unary, binary, etc. depending on how many different members of the population it operates on. Mutation operators are unary variation operators. Crossover operators are binary variation operators.

**Weighted fitness functions.** Combining two or more measures of quality into a single fitness function by taking a sum of constants times the quality measures yields a weighted fitness function. The constants are called the weights and establish the relative importance of the different fitness measures.

