1 EAs and SAT

In the Boolean Satisfiability Problem (SAT), one has a Boolean expression composed of conjunctions and disjunctions of literals. A literal is a Boolean variable or the negation of a variable. The variables v_i can be true or false. An example (in prefix notation) is $(\wedge\ (\vee\ v_1\ \ \neg v_2\ \ v_3\ )\ (\vee\ \neg v_1\ \ \neg v_4\ \ \neg v_5\ ))$. One class of these problems is called "3-SAT", which has the form given in the previous sentence. Expressions have the form $(\wedge\ clause_1\ \ clause_2\ \ ...\ \ clause_M\ )$. Each clause has the form $(\vee\ .\ \ .\ \ .\ )$ where each "dot" is a literal. The goal is to find any assignment to the variables such that the expression evaluates to true. There are N variables and M clauses.

1(a) EAs for 3-SAT

Assume the problems are 3-SAT in nature.

If we solve this with a genetic algorithm, an individual can simply be a bit string of length N. Bit i is 1 if v_i is true, or bit i is 0 if v_i is false. Note that mutation and crossover will work fine. Define a fitness function that evaluates better individuals as having higher fitness, with the maximum fitness being equal to 1.0 (this would be a solution that satisfies the whole expression). Give me a function that will work with general 3-SAT problems of N variables and M clauses.

1(b) EAs for SAT

Now assume that the Boolean expressions have arbitrary form. For example, we could have $(\wedge\ \neg(\vee\ v_1\ \ v_2\ )\ v_4\ (\vee\ \neg(\wedge\ v_2\ \ v_3\ )\ v_5\ \ (\wedge\ v_5\ \ \neg v_6\ )))$. In other words, the expression is an arbitrary combination of ANDs, ORs, NOTs and variables. Note that this time whole expressions can be negated. Again assume an individual is a bit string of length N, with 1 = true and 0 = false.

Define a fitness function that evaluates better individuals as having higher fitness, with the maximum fitness being equal to 1.0 (this would be a solution that satisfies the whole expression). Give me a function that will work with general SAT problems, not just the trivial example above.