NAT Tutorial 2

1. (Goldberg) You are asked to minimize a function $f(x,y,z)$ where
   $-20 < x < 125$, $0 < y < 1200000$, $-0.1 < z < 1.0$
   and the desired precisions for x, y and z are 0.5, 10000 and 0.001 respectively. Using the 'customary' grid-based binary encoding idea of dividing ranges into some power-of-2 number of points, how many bits are needed?

   Answer: need 291 points for x, thus 9 bits. Need 121 points for y, so 7 bits. Need 1101 points for z, so 11 bits. Therefore 27 bits in all. And see below...

2. You want to try to represent the value of some integer quantity n in some binary-encoded way, but n only ranges over the integers 1..17. How might you do it? Discuss the advantages and disadvantages.

   Answer: there isn't a fair way, but that may not matter. E.g. use 5 bits, divide the range 1..17 into 32 points, map each to nearest integer. Then integer 0 is represented only by 00000, but integer 1 is represented by 00001 (ie 17/32), and by 00010 (ie 34/32). The success of the GA does not depend on all points being EQUALLY represented; you just want to avoid a genuine-needle-in-haystack problem.

3. Proteins are made up of sequences of amino acids connected by chemical bonds. The protein sequence "folds up" into a three-dimensional structure of low energy by rotation of the chemical bonds connecting the amino acid groups. The three-dimensional structure will then determine the biological function of the protein, so it is important to be able to predict this structure from the sequence of amino acids in the protein. Consider how you could apply a genetic algorithm to find the three-dimensional structure of lowest energy for a given protein sequence. Pay particular attention to how you would represent the candidate structures, the fitness function you would use, and the types of crossover and mutation.

   Answer: See Mitchell p.61 onwards.

4. (Mitchell) When is the union of two schemas also a schema? When is the intersection of two schemas also a schema? E.g. the union of 1* and 0* is **; the union of 10 and 01 is not a schema.

   Answer: reminder: union of two sets = set containing all members of EITHER
   intersection = set containing all members of BOTH
   Any schema is a set whose size is a power of 2. In considering the union of two schemas, e.g .

   1*1*0**
   1*0****

   it should be clear that the result is not a schema. It would have to be of the form 1***? ** but that ? cannot be a 1, cannot be a 0 and cannot be a *. A bit more thought on these lines suggests that if the union of two schemas is to be a schema, then you must look at where the defined bits are. If the two disagree about the value of a defined bit (one has 1, one has 0) then the result must have a * there. In which case all the other

defined bits in either schema must exactly match the other schema. A similar sort of analysis can be done for intersection, e.g. - look at each position:
- both defined? - they differ? --> intersection is empty (so not a schema)
- the same? --> potential schema has that value
- one defined? --> potential schema has that value
- neither defined? --> potential schema has wildcard there

5. (Computer exercise from Mitchell) Implement a simple GA with fitness-proportional selection, roulette-wheel sampling, population size 100, single-point crossover rate p_c=0.7, and bitwise mutation rate p_m=0.001. Try it on the following fitness function: *f(x)*=number of ones in *x*, where *x* is a binary chromosome of length 20.
Perform 20 runs, and measure the average generation at which the string of all ones is discovered. Perform the same experiment with crossover turned off (i.e. p_c=0.0). Do similar experiments, varying the mutation and crossover rates, to see how the variations affect the average time required fort he GA to find the optimal string. If it turns out that mutation with crossover is better than mutation alone, why is this the case?

   Answer: (first part) give it a try.
(second part) This is the claim of the building block hypothesis. There might be problems where the crossover does not make much of a difference or where its benefits are canceled out by the unavoidable disruptive effects, but in the present case it is obvious that parts of the optimal solution can be identified by different individuals and can be combined in their children, such that a net benefit of crossover should be observable.

6. Discuss the “criticisms” in the wikipedia article on GA.

   Answer: Criticisms of GAs and metaheuristic algorithms in general have helped to point out the need for a better understanding of the algorithms, a better understanding of the problems and a better understanding of learning, optimisation and information processing in real-world tasks in general. We will come back to this after ACO and PSO.

7. [Selection mechanisms] Investigate whether binary tournament selection (i.e. tournament size 2) is equivalent to linear ranking selection (i.e. selection in which the fittest of *N* gets *N* chances, the next-fittest gets (*N*-1) chances, etc., and the least fit gets one chance). Tournament selection is selection where *m* individuals are chosen randomly from the population and the best *n* of those m are selected. So in binary tournament selection $m = 2$ and $n = 1$.

   Answer: Possible argument, courtesy of Peter Ross: in tournament selection, how would it pick the *K*-th ranked? By picking that one (at random) and also picking one that is *K*-th *or lower* ranked. Consider the chance of that: what you get is exactly linear ranking selection.

   Check this argument over by trying out this real example:

   C1: fitness = 4
   C2: fitness = 3
   C3: fitness = 2

C4: fitness = 1

There are 16 possible pairs: (C1,C1), (C1,C2), (C1,C3), (C1,C4),
(C2,C1), (C2,C2), (C2,C3), (C2,C4),
(C3,C1), (C3,C2), (C3,C3), (C3,C4),
(C4,C1), (C4,C2), (C4,C3), (C4,C4).

Of these, 7 pick select C1, 5 select C2, 3 select C3 and 1 selects C4. So, this is a linear ranking selection, but the number of chances is in fact equal to 2R-1 where R is the position of the candidate in reverse rank order (least fit = 1, next = 2,... most fit = *N*).

8. [Schema theorem] A population consists of the following strings. The probability of crossover is 0.75 and the probability of mutation is 0.1. How many instances of the schema *0***0 would you expect in the next population? [From an exam paper.]

String	Fitness
100100	20
001000	20
110111	30
100101	20
100010	10

Answer: Here, we should apply the schema theorem, as given in Lecture 4.

For *0***0, there are 3 instances, so m(H,t) = 3. The average fitness of these two is 50/3, i.e. 16.6. So we get:

E(m(*0***0,t+1)) = 3 * 16.6/20 = 3 * 5/6
Note that this does not yet include the disruptive effects of crossover and mutation, so this only covers the first bit of the formula. Defining length is 4, so there is a chance of 0.75 (1-4/5) that the schema survives crossover. Order is 2, so with probability (1-0.1)^2 it survives mutation. Taken together we have

E(m(*0***0,t+1)) = 3 *5/6*0.15 *0.81 = 0.3

Moral: The schema theorem predicts that this schema may not be present in the new generation. Note, however, that the schema theorem gives only a lower bound. The recombination may occur between individuals that both carry the schema or there is a chance that the fourth individual is mutated into this schema. Therefore, probably there will be one or two instances of the schema left. The most drastic effect is here by the crossover probability and the relatively high order of the schema.
Note also that high fitness schemata like the ones in the third individual realising there chances on the cost of the other ones and spread quickly through the population, low fitness ones are quickly destroyed.

9. If

$f(****) = e_0$
$f(***1) = e_0 + e_1$
$f(**1*) = e_0 + e_2$
$f(**11) = e_0 + e_1 + e_2 + e_3$

(these equations DEFINE $e_0, \ldots, e_3$), then what is $f(**01)$ in terms of $e_0, \ldots, e_3$?

Answer: here, f is the average fitness of the schema over all actual members of the set. So:

f(***1) = (f(**11) + f(**01)) / 2

Why is this?

Because (from Q3) ***1 (8 members) is the union of **11 (4 members) and **01 (4 members), so the average fitness of ***1 is the average of the average fitnesses of **11 and **01.

Rearrange this equation:

2 x f(***1) = f(**11) + f(**01)

So: f(**01) = 2 x f(***1) - f(**11)

= 2.e0 + 2.e1 - (e0 + e1 + e2 + e3)
= e0 + e1 - e2 - e3

10. (repeated from 1 tutorial) Discuss implications of the schema theorem for the following cases (recall the definition of the fitness of a schema):

- a single instance of a high-fitness schema
- two different non-overlapping schemas with the same fitness
- two partially overlapping schemas with a fitness that are both high but not the same
- a fitness function that depends on the presence of other individuals, such as in the evolution of an ecosystem consisting of rabbits and foxes.

Answers:

- a single instance of a high-fitness schema may not necessarily be selected or may get killed by mutation or recombination, so although the expectation is high it may not survive.
  Also it should be discussed that "high-fitness schema" refers to the average over all individuals with that schema, if there is only one then it is not clear whether it is "high-fitness" and in the next generation it may turn out to be not that great.
- this is an unstable solution; the average in the next generation will probably introduce an advantage for on of them such that it may take over later on (but possibly slowly)
- here the decision might be faster, the better one is likely to win, but in recombination they may be recombined into a new type of a scheme (if the partiuclar place is cut)
- here also stable (or oscillating) solutions with several surviving species are possible, this is very important in natural evolution, but will not be of further interest here.