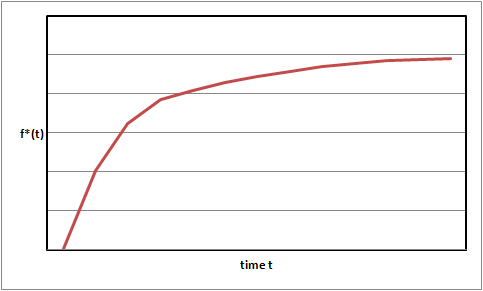
**Assignment 48** (2/2 points)

Advantages of Evolutionary Algorithms:

1. provides quick solutions - whilst optimum search algorithms might take a long time to find the solution, EA is able to produce solutions fast, even if they are suboptimal. This means that an interruption of EA processing will still provide some solution, instead of providing none (optimum search) where time is of essence.
2. provides good solutions for nasty functions where no other good algorithm (like gradient descent) is known to work for
3. offers a framework such that it is comparably easy to incorporate prior knowledge about the problem.

**Assignment 49** (0/1 points)



**Assignment 50** (3/3 points)

*How large is the percentage of offspring that are identical to either parent p1 or p2?*

100% of the offsprings will be identical to the parents.

*How will the result change, if the Hamming distance between the parents is 2?*

50% to 100% of the offsprings will be identical to the parents, depending on the selection of parents and the position of the split. If L is large enough and you run many simulations, the probability approximates (66%).

Explanation: The two bit flips (Hamming distance=2) divide the genome into three zones. The first and the last zone contain the flipped bit while the middle zone is always equal. The genome is *not* identical if the cross-over is chosen to be in the middle zone. The average probability (depending on the parents and the bit flips) of this is one third.

Eg.

Hamming distance = 1 produces *100% match*, ie. binary genomes that are genetically closest produce no variety in their offsprings.

|  |  |  |
| --- | --- | --- |
| 2 bit vectors | 3 bit vectors | 4 bit vectors |
| Parents:   |  | | --- | | 0|1  0|0 |   Produce offsprings:  00  01 | Parents:   |  |  |  | | --- | --- | --- | | 11|0  01|0 | OR | 1|10  0|10 |   Produce offsprings:  110  010 | Parents:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 1|110  1|111 | OR | 11|10  11|11 | OR | 111|0  111|1 |   Produce offsprings:  1111  1110 |

Hamming distance = 2 produces *50% - 100% match*

|  |  |  |
| --- | --- | --- |
| 2 bit vectors | 3 bit vectors | 4 bit vectors |
| Parents:   |  | | --- | | 1|1  0|0 |   Produce offsprings:  10  01  00  11 | Parents:   |  |  |  | | --- | --- | --- | | 11|0  01|1 | OR | 1|10  0|11 |   Produce offsprings:  111  010  110  011  Parents:   |  |  |  | | --- | --- | --- | | 00|0  01|1 |  | 0|00  0|11 |   Produce offsprings:  001  010  000 000  011 011 | Parents:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 1|010  0|000 |  | 10|10  00|00 |  | 101|0  000|0 |   Produce offsprings:  1000 1000  0010 0010  1010 1010 1010  0000 0000 0000  Parents:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 1|010  0|110 |  | 10|10  01|10 |  | 101|0  011|0 |   Produce offsprings:  1110  0010  1010 1010 1010  0110 0110 0110 |

**Assignment 51** (0/2 points)

TODO

**Assignment 52** (2/2 points)

A super-individual is an instance of the population with an outstanding fitness value. If the parent selection and/or the external selection is fitness based, the super individual will soon dominate the entire population and hence will limit the diversity of the population which is important for exploration. This is especially bad if the super-individual reflects a local maximum.

A counter measure is to assure that a certain amount of diversity will be preserved. This can be achieved eg. if duplicates (of the super-individual) are removed of the population or if the external/parent selection is not only fitness based but also includes random instances.

**Assignment 53** (2/4 points)

*Population*

P = 100, population size kept constant.

The genome is a vector of length 50:

genome x = (x1, x2, …, x50),

where xi stands for the percentage amount of an additive ai in the mixture (min 0% - max 10%) and the sum of all additives equals 100%. At any time, no more than 20 additives can be set.

Example genome with 11 additives out of 50 are set: (10,10,10,10,5,5,10,10,10,10,10,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)

Given the a priori knowledge, we can deduce that the maximum number of additives per mixture is 20 and the minimum is = 10. We shall hence instantiate an initial population of 100 individuals of 10 additives set to maximum value (we leave it to the inheritance to generate other values in between and genomes of different number of additives set).

*Inheritance*

k=2 parents, offspring genome calculated from the mean of the 2 parents. Eg.

(10,10,10,10,10,10,10,10,10,10,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)

+

(0,0,0,10,10,10,10,10,10,10,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,10,10,10)

=

(5, 5, 5, 10,10,10,10,10,10,10,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,5,5,5)

*Mutation*

We randomly pick 2 positions in the genome and switch values with a probability of 0,1.

Selection strategy

Fitness based elitism. We throw away illegal genomes where more than 20 additives are set. Rank proportional (wheel of fortune). Take the best μ individuals (μ + λ), with λ = 20, μ = 80.

*Fitness function*

Each student grades the soft drink on a scale of 0-4 (maximum grade is 4). Fitness function hence calculates the mean grade of the pool of students:

*Finishing*

Algorithm stops when fitness no longer improves or the mean performance of the best 5 individuals is > 3.

**Assignment 54** (0/1 points)

One example that became quite popular in 2008/2009 is the “Evolution of Mona Lisa” where genetic programming was used for polygon rendering:

<http://rogeralsing.com/2008/12/07/genetic-programming-evolution-of-mona-lisa/>

The procedure is described in 5 steps:

0) Setup a random DNA string (application start)

1) Copy the current DNA sequence and mutate it slightly (inheritance, mutation)

2) Use the new DNA to render polygons onto a canvas

3) Compare the canvas to the source image (fitness evaluation)

4) If the new painting looks more like the source image than the previous painting did, then overwrite the current DNA with the new DNA (external selection)

5) repeat from 1

Population size: 2 (parent + child)

External selection: only keep best individual

Fitness evaluation: pixel-by-pixel compare of the individual to the target image (mona lisa)

Mutation: Randomly add or remove points. Randomly change brush color (red/blue/green/alpha) and the points that contain to each polygon.