Autonomous Intelligent Systems,
Institute for Computer Science VI, University of Bonn

Dr. N. Goerke
Friedrich-Ebert-Allee 144, 53113 Bonn, Tel: (0228) 73-4167
E-Mail: goerke@ais.uni-bonn.de
www.ais.uni-bonn.de

Exercises for Artificial Life (MA-INF 4201), SS11

Exercises sheet 6, due: Mon 30.05.2011

23.5.2011

Group	Name	35	36	37	38	39	40	$\sum$

Assignment 35 (1 Point)

Count, and sum up the points you have achieved in the assignments for the Artificial Life Lecture SS11 so far, and calculate the number of points that you still need to reach the 50% limit that is necessary for being admitted to the written exam.

Assignment 36 (2 Points)

Explain how a hypercube and a binary genome of an Evolutionary Algorithm are related to each other. Draw a sketch, visualizing this for a binary genome that has more than two bits.

Assignment 37 (3 Points)

Gradient descent is a widely used method of optimization, still it has some drawbacks. Name, and describe (**in your own words**) at least three properties of a gradient descent task, that could be called negative.

Assignment 38 (1 Point)

Discuss the following sentence:
Minimization problems and maximization problems are equivalent to each other.

Assignment 39 (5 Points)

The efficient use of Evolutionary Algorithms requires, that the fitness function is structured properly.
Propose a fitness function that measures the quality of a weekly time table of lectures and exercise groups.

Please take the following conditions into account:
Each time-slot for lectures and exercise groups is 2 hours, we have 5 timeslots per day and 5 days per week, with 4 lecture halls that can be operated in parallel, and 8 exercise rooms that are available in every timeslot.
The number of lectures is $N = 25$, each lecture is accompanied by 2 exercise groups, no room can be occupied by two things at the same time (collision), it is not allowed, that the exercise groups are operated in parallel to the respective lecture (that would be considered a collision as well).

It is wanted that the schedule enforces to attend as much lectures and exercises as possible, at the same time the students have asked for reducing the use of the first timeslot on Monday early morning (e.g. 8-10) and the last two timeslots on Friday afternoon (if possible).

Assignment 40 (3 Points)

Within an Evolutionary Algorithm a parent individual $X(i)$ with a genome of L bit has created N offspring $X(i) = Y(i)_n$ identical to the parent $X(i)$.
To yield the new generation $Y(i+1)_n$ the mutation operator is modifying each of these N offspring $Y(i)_n$ by flipping each of the $N * L$ bits with a probability of p.

Derive a formula that calculates the propability Q for the case that **none** of the N new individuals $Y(i+1)_n$ is identical to the parent $X(i)$.

Use your formula, and calculate the probability Q for the case that $N = 20, L = 100$ and $p = 0.01$.

Artificial Life - Exercise 6
Jaana Takis, Benedikt Waldvogel

Assignment 35

Sheet	1	PA-A	2	3	PA-B	4	5	**Total**
Points	15	14	15	11	10	14	13	**92**

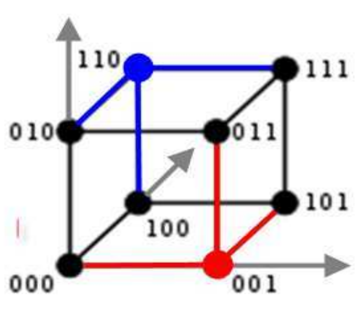
So we need 8 points (200 are possible) to be admitted to the written exam.

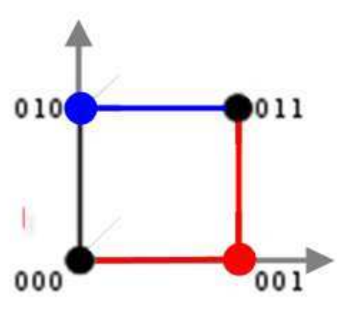
Assignment 36
We can view the binary genome of length *n* as a vector in R^n by treating each bit in the genome as a real coordinate. Two vertices are adjacent whenever their labels differ in a single bit and we can visualise this as a hypercube. In the scope of the Evolutionary Algorithm this hypercube model is often used in the inheritance stage for determining what are the possible offsprings of the parents. Ie. most evolutionary algorithms are designed so that the offspring is more likely to be an individual that is genetically close to its parents (ie. from the immediate neighbourhood of the parents). The area enclosed by the hypercube is considered to be the exploitation zone of possible offsprings - in the case of binary values, the possible offsprings lie in the vertices. The final selection of the offspring depends on the distinct algorithm itself (eg. how much weighting it gives each parent or what the crossover positions are). The hypercube model is also applicable for more than 2 parents.

Eg.

a) Two parent binary genomes labelled 110 and 001 form a hypercube in 3-dimensional space with possible offsprings at its vertices: 111, 011, 010, 100, 101, 000

b) Two parent binary genomes 010 and 001 form a hypercube in 2-dimensional space with 2 possible offsprings at its vertices: 011 and 000.





Assignment 37
Some drawbacks of the gradient descent method:

- one can get stuck in local minima in the case of deep descents and one is hence unlikely to find the global optimum.

- can only be applied to continuously differentiable functions, ie unusable in cases with 0 gradient.
- cannot be used in cases of limited knowledge where function not completely given which is often the case of real-world applications - ie. discontinuities in the function.

Assignment 38

If one makes the assumption that minimising costs is equivalent to maximising performance, we can reduce the task of optimisation to only finding the global minima, rather than the maxima. Since we often already know the minimum possible value of costs, it makes the task of finding the global maximum easier. However, the latter also presents a problem in this case - if costs are zero and we assume $costs = \frac{1}{performance}$ then performance is infinity which is rarely the real-world case. Hence, our opinion is that the minimisation and maximisation problems are not always completely equivalent.

Assignment 39

For the genome, we propose to use a 25x12 table where each cell can either have no value or a value between 1 and 25. The value reflects the number of the lecture (in case of an exercise it's the number of the lecture the exercise is for). The first four rows reflect the four lecture halls while the last 8 rows reflect the exercise rooms. Furthermore, we assume that all 25 lectures take place as well as the 50 exercises while there's no collision. I.e. There are 75 cells set where each value occurs exactly three times.
To enable students to attend as many lectures and exercises as possible, it's good if the schedule is pretty much fragmented and as little events happen in the same slot as possible. The first table shows an example where such a fragmentation is pretty well while the second table shows an example of the pretty much worst case.
The fitness function should be designed in a way that schedules like the first one get a much better than schedules that tend to be like the latter one.

We propose to count the lectures on each of the 25 timeslots and take the inverse. We also take the exercises per timeslot and take the inverse. The first slot on Monday and the two last slots on Friday are furthermore multiplied with a malus factor of 0.5.
To calculate the final fitness value we take the average of the two sums. We see that the first table has a fitness value of $\frac{1}{2}(14.5 + 11.2) = 12.8$ and the second table has much lower fitness value of $\frac{1}{2}(2.1 + 1.3) = 1.7$

Room	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
L1		1		5	2			3		10		25					14		23			21				
L2					15				17					19				24			13					
L3			9	11			18						7				20						22			
L4			16							4					12				6		8					
E1																			3							
E2					8		1			6		9		25	18							22		5		
E3			1	24				17	22		16					12		7		13			14			
E4			19			15			8					2			21									
E5				2								14			24				25		9		11			
E6		13					11		21	3							5			17						
E7					10	20				19			6			16		23				4		18		
E8		7		15				4			12			20					10		23					
	0,0	1,0	0,5	0,5	0,5	0,0	1,0	1,0	1,0	0,5	0,0	1,0	1,0	1,0	1,0	0,0	0,5	1,0	0,5	0,0	0,5	1,0	1,0	0,0	0,0	**14,5**
	0,0	0,5	0,5	0,3	0,5	0,5	0,5	0,5	0,3	0,3	0,5	0,5	1,0	0,3	0,5	0,5	0,5	0,5	0,3	0,5	0,5	0,5	0,5	0,5	0,0	**11,2**
																										12,8

Room	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
L1	1	5	9	13																				18	22	
L2	2	6	10	14																				19	23	
L3	3	7	11	15																				20	24	
L4	4	8	12	16																			17	21	25	
E1	5	1	5	17																			16	22	1	
E2	6	2	6	23																			22	23	2	
E3	7	3	7	24																				24	3	
E4	8	4	8	25																				25	4	
E5	9	13	9	18																				13	18	
E6	10	14	10	19																				14	19	
E7	11	15	11	20																				15	20	
E8	12	16	12	21																				17	21	
	0,125	0,25	0,25	0,25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0,125	0,125	**2,1**
	0,125	0,125	0,125	0,125	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,5	0,125	0,125	**1,3**
																										1,7

Assignment 40

No single flip operation must happen. The Probability that the flip operation does *not* happen is $1-p$. Since there are $N*L$ flip operations, it follows from simple laws of probability:

$$Q(p,N,L) = (1-p)^{N*L}$$

$$Q(0.01,\ 20,\ 100) = 0.99^{2000} \approx 1.86 * 10^{-9}$$