*COSC 6367 Evolutionary Programming*

*Dr. Eick*

*Final Exam*

Tu. May 8, 2012

Your Name:

Your Student id:

Problem 1 --- Theory of Genetic Algorithms [10]:

Problem 2 --- Design an EC System and GP [13]

Problem 3 --- Interactive Evolution [6]

Problem 4 --- Multi-Modal Fitness Landscapes [6]

Problem 5 --- Classifier Systems and XCS [13]

Problem 6--- Numerical Optimization with EC [7]

Problem 7 --- ABM and NetLogo [12]

:

**Grade:**



The exam is “open books and notes” and you have 110 minutes to complete the exam; it will count approximately 25% towards the overall course grade.**1) Theory Genetic Algorithms [10]**

a) Assume the following schema GA is given: 11\*\*0001\*\*\*

What is the order and defining length of the schema? What is the probability that a single instance belonging to this schema is destroyed by a single-bit flip mutation that occurs with probability pm? [3]

Order: 6 [1]

Defining Length: 8-1=7 [1]

6\* pm [1]

No\_partial\_credit!

b) Under which circumstances will an instance of the schema 1\*\*0\*\*\*\*\*\*\*\*\* be actually destroyed by crossover—specify all conditions that must hold in order for an instance to be destroyed! [3]

An instance of 1\*\*0\*\*\*\*\*\*\*\*\* gets destroyed ↔

1. The crossover point is between position 1 and 4 (3 possibilities)

**and**

1. If the mate does not belong to 1\*\*\*\*\*\*\*\*\*\*\* or \*\*0\*\*\*\*\*\*\*\*\*\*\*

c) The equation of the schema theorem on page 191 equation 11.1 of the text book contains an “≥” and not an “=” sign—explain! What is the impact of the “≥” sign when equation 11.1 is used to make predictions about the growth of instances of a particular schema over time? [4]

It ignores the positive affect operators can take [2.5]:

* Mutation might create new instances of the schema
* Crossover does not lead to destruction due to condition b) observed in the previous question

Forgot one of the two bullits, subtract 1 point!

When making prediction the formula creates only a lower bound for the percentage of instances which belong to the particular schema; the average actual number could be significantly larger [1.5]

**2) Using Genetic Programming to learn a Boolean Function [13]**

Assume you want to learn a Boolean function f(a,b,c,d) with 4-input variables a, b, c, d, such as[[1]](#footnote-1) (a∧b) ∨ (c^~d) from training data, which have the following form:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | output |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |

For example, the Boolean function (a∧b) ∨ (c^~d) predicts the output of the first 4 examples correctly, but not the fifth training example.

Give a sketch of the design of a genetic programming system which can be used for this task; in particular, discuss which terminal set, non-terminal set, fitness function, genetic operators, initialization approach, selection strategy, and population management approach you would choose for your system!

**No answer given!**

*More space for Problem2*

**3) Interactive Evolution [6]**

Describe an application of your own choice that might benefit from interactive evolution. Give a sketch how interactive evolution will be used for this application. Limit your answer to 6-8 sentences!

**No answer given!**

**4) Multi-Modal Fitness Landscapes [6]**

1. What is genetic drift? How does genetic drift impact finding different good solutions in multi-modal fitness landscapes? [3]

Definition see textbook [1]

If there is a lot genetic drift, solutions tend to concentrate on a single hill rather that occupying multiple hills!

b) What is the key idea of the fitness *sharing approach* to main diversity in a population? Limit yourself to 2-4 sentences! [3]

also see textbook!

A fitness function is used which combines solution fitness with the density of the location of the solution with respect to the population density; in particular, solutions which are in dense areas with respect to the population are penalized, and the fitness of solutions which are in sparse areas obtain a higher fitness value.

**5) Classifier Systems and XCS[13]**

a) Evolutionary computing approaches for numerical optimization problems usually create a new population from the scratch when searching for good solutions, whereas classifier systems rely on a steady state approach in which new generated rules replace older rules in the population. Why do you believe this is the case? [4]

There is not much sense in destroying rules that are good in obtaining rewards from the environment. In optimization problems, on the other hand copying a solution does not lead to a better solutions; in general, the objective is to explore promising new solutions which have not been seen before.

b) One objective of XCS-style systems is to create rulesets with high coverage: for each possible input a rule or set of rules should exist that can process this input. What do XCS-style classifier systems do to accomplish this goal? [3]

* A new rule are created when there is no rule that can process the current input [1]
* Rules with large action set sizes are more likely to be destroyed, avoiding that too many rules are specialists for the same subproblem [2]

c) Assume a rule r’s current action set size parameter ar is 3, and r occurs with 5 other rules in the action set—how will this information be used to update ar? Compute r’s new value of ar, assuming a learning rate of =0.2 is used! [2]

3x0.8+5x0.2=3.4

No\_partial\_credit!

d) How does XCS select rules to process a particular input? [4]

a) Match Rule; find set of rules that match the current [1.5]

b) Partition matching rules based on their action [1]

c) Select the action/rule partition which has the higher average for prediction value weighted by their accuracy! [1.5]

**6) Using EC for Numerical Optimization [7]**

Assume the following function f

f(a,b,c) = cos(c)\*sin(b)+((a+b)/(1+ c\*\*sin(a-b)))

has to be **maximized** subject to the following two constraints:

(C1) (b\*\*2 + c\*\*2) =1

(C2) a+b<1

Assume you use the penalty function approach to solve this optimization problem. Give the fitness function you would use in this case and explain how it will be used! [7]

Use the following fitness function g[6]:

g(a,b,c)= f(a,b,c) – w\*(max(0, a+b-1)+ |b\*\*2 + c\*\*2-1|)

Major Errors: 0-1 points; partially correct solutions 2-3.5 points

w is increased as evolution evolves (other answers might also deserve credit!) [1]

**7) ABM and NetLogo [12]**

a) What are the unique characteristics of agent-based modeling compared with other modeling approaches? Limit your answer to 3-6 sentences! [4]

autonomous agents [1.5]

agent act in space [1]

agents interaction with each other [1]

different types of agent are supported [0.5]

no centralized control [0.5]

agents act probabilistically [0.5]

At most 4 points! Other answers might deserve credit!

b) Design a NetLogo system called *Emerging Colors* with the following features [8]:

* Setup the system by creating a 100 turtles with a randomly assigned color either ‘red’, ‘blue’ or ‘yellow’. Moreover, spread the created turtles in space.
* In each iteration, move each turtle forward by an arbitrary number of steps between 0 to 19. With probability 10%, change its color to the color of its nearest neighbor.

Write your procedures accordingly, and include your NetLogo source code below (see next page for useful commands, you might not be aware of).

Changing Colors NetLogo Code—a Demonstration of Genetic Drift

to setup

ca

create-turtles 100

[

set color randC

setxy random-xcor random-ycor

]

end

to go

ask turtles

[

fd random 20

if random 10 = 0

[

let color\_n [color] of min-one-of turtles [distance self]

set color color\_n

]

]

end

to-report randC

let r random 3

if r = 0

[ report red]

if r = 1

[ report blue]

if r = 2

[ report yellow]

End

*There are many different ways to solve this problem!*

1. “~’ represents the negation operator [↑](#footnote-ref-1)