University of Waterloo
Department of Electrical and Computer Engineering
ECE 457A Adaptive and Cooperative Algorithms
Final Examination
August 1, 2012

Attempt all questions. If you are in doubt, make an assumption and attempt the question. Good Luck!

Question 1. [25 points]

(a) [10 points] Explain how the following techniques achieve exploitation and exploration. For each technique identify the control mechanism to achieve either modes of search:

I. Simulated Annealing
II. Genetic Algorithm
III. PSO
IV. Genetic Computation
V. Genetic Programming

(b) [2 points] Suppose you have an admissible heuristic h. Is h^2 admissible? Is $\sqrt{h}$ admissible?

Would using any of these alternatives be better or worse than using h in the A* algorithm?

Answer: h^2 may not be admissible, because $h^2 \geq h$ when $h \geq 1$, so it may exceed the optimal distance to goal. $\sqrt{h} \leq h$ for $h \geq 1$, so it is admissible (assuming integer values for the heuristic, which is typical). It will likely work worse than h, though, because its estimate is farther from the optimal value (so it is a worse estimate of the cost-to-go).

(b) [2 points] Explain the difference between a genotypic representation and a phenotypic representation. Give an example of each.

A genotypic representation is one where the underlying representation has to be decoded in order to determine the behaviour of the organisim.

2 marks

An example could be a bit string, which represents a set of real numbers with the binary digits having to be decoded in order to determine the real numbers, which can then be evaluated by carrying out the relevant function.

2 marks

The underlying representation for a phenotypic representation represents the problem in a natural way. The behaviour is expressed directly into the representation.

2 marks

An example, could be a robot arm, where the coding is directly related to the movement of the arm. For example, move right, turn 21°, move up etc. etc.

(d) [2 points] Outline the similarities and differences between Genetic Algorithms and Genetic Programming.
(e) [4 points] Name the components of the gene code in Evolutionary Strategies and explain the function of each component.
(f) [3 points] Describe how the crossover and mutation operators work in evolutionary computation. You can use a simple example to illustrate your answer.
(g) [2 points] Describe the terms recency with respect to tabu search.

The list of visited moves (which is sometimes called a history) does not usually grow forever. If we did this we might restrict the search too much.
Therefore, the purpose of a Recency function is to restrict the size of the list in some way (it is called a Recency function as it keeps the most recently visited states in the list – discarding the others).
The easiest (and most usual) implementation of this function is to simply keep the list at a fixed size and use a FIFO (First-In, First-Out) algorithm to maintain the list. But other methods are sometimes implemented, depending on the problem. Maybe the list-size parameter is dynamic in that it changes as the algorithm runs. One method of doing this is to keep the list small when states are not being repeated very often but, when repeated states keep being generated the list is made larger so that the search is forced to explore new areas.

Question 2. [20 points]
A budget airline company operates 3 planes and employs 5 cabin crews. Only one crew can operate on any plane on a single day, and each crew cannot work for more than two days in a row. The company uses all planes every day. A Genetic Algorithm is used to work out the best combination of crews on any particular day.

a) Suggest what chromosome could represent an individual in this algorithm

Answer: *On each day, a solution is a combination of 3 cabin crews assigned to 5 airplanes. Thus, a chromosome of 3 genes could be used in this algorithm with each gene representing a crew on a certain plain.*

b) Suggest a fitness function for this problem.

Answer: *You may come up with different versions, but it is important for the fitness to take into account the condition that cabin crews cannot work more than 2 days in a row. For example, the fitness function can take into account how many days each crew has left before a day off (e.g. 1 or 0). The fitness could be calculated as the sum of these numbers for all drivers in the chromosome.*

c) How many solutions are in this problem? Is it necessary to use Genetic Algorithms for solving it? What if the company operated more plains and employed more crews?

Answer: *The number of solutions is the number of times 3 crews can be selected out of 5 without replacement and without taking into account their order. The first crew can be selected in 5 different ways, the second in 4 ways and the third in 3 different ways. These numbers multiplied together will give us total number times how 3 crews can be selected randomly out of 5:* $5 \times 4 \times 3 = 60$ *times. However, there are 6 possible combinations in which 3 crews can be ordered, and because the order does not matter the answer is* $60/6 = 10$*. Thus, there are 10 possible solutions for this problem.*

It is not really necessary to use GA for a problem with such a small population, because solutions can be checked explicitly. However, as the number of crews and airplanes increases, so does the number of solutions, and the use of GA can be the only option. In fact, if n *is the number of cabin crews and* $k \leq n$ *is the number of airplanes, then the number of solutions is*

$$\frac{n!}{k!(n-k)!} \, .$$

For example, if the company operated 10 airplanes and employed 20 cabin crews, then the number of solutions would be

$$\frac{20!}{10!(20-10)!} = 184,756$$

Question 3. [15 points]

a) Define the acceptance function that is used by simulated annealing and describe the terms of this function.

The acceptance probability is defined as:

$$P = \begin{cases} 1 & if \Delta c \leq 0 \\ e^{-\Delta c / t} & if \Delta c > 0 \end{cases}$$

where:
Δc is the change in the solution cost,
t is the current temperature (control parameter, no physical role).
Notice that we dropped Boltzmann's constant

(c) Why/how can the simulated annealing based algorithm (SA) escape from a local optimum?

Simulated annealing provides a powerful tool for escaping local optima by allowing moves to lower quality solutions with a pre-specified probability. Another big plus of Simulated Annealing is its ease of implementation.

At each iteration of Simulated Annealing, the objective function value of the current solution and a new generated solution are compared. Improving moves are always accepted while only a fraction of non-improving moves are performed with the aim to escape local optimal solutions. The probability of accepting a non-improving move depends on the non-increasing parameter of temperature.

Can it always guarantee a global optimum solution?

Yes, Simulated Annealing always (or at least almost always) find global optimum.

(c) The following table shows six evaluations of a simulated annealing algorithm. For each evaluation give the probability of the next state being accepted. Assume the objective function is being maximised.

No.	Current State (Evaluation)	Potential New State (Evaluation)	Temperature	Probability of Acceptance
1	80	30	20	0.082085
2	80	30	234	0.807611
3	80	40	20	0.135335
4	80	40	234	0.842872
7	80	100	20	2.718282
8	80	100	234	1.089229

Question 4. [15 points]

(a) In the context of PSO explain the following function and explain all parameters

$$v_{t+1}^{id} = w * v_t^{id} + c_1 r_1^{id}\left(pbest_t^{id} - x_t^{id}\right) + c_2 r_2^{id}\left(Nbest_t^{id} - x_t^{id}\right)$$
$$x_{t+1}^{id} = x_t^{id} + v_{t+1}^{id}$$

The functions describe the velocity of each particle as it moves in the solution space. This velocity moves the particle from its current position (solution) x_t to a new position (solution) x_{t+1}.

- particle current position x_i,
- particle current velocity v_i,
- The best position the particle achieved so far, personal best, *pbest*$_i$, or sometimes p_i for short,
- The best position achieved by particles in its neighbourhood *Nbest*

(b) In the equation in (a) identify and explain the inertia, cognitive and social terms.

- Inertia term: $w * v_t^{id}$

This term determines how the previous velocity of the particle influences the velocity in the next iteration.

- Cognitive term: $+c_1 r_1^{id}\left(pbest_t^{id} - x_t^{id}\right)$

The cognitive behavior helps the particle to remember its previous visited best position.

- Social term: $+c_2 r_2^{id}\left(\text{Nbest}_t^{id} - x_t^{id}\right)$

Causes the particle to accelerate toward the social best. This term reflects an attraction towards the best particle in the neighborhood.

(c) Show how PSO can be used to find the optimal solutions to the following problem. You can answer this question using pseudo code, flowchart, or provide succinct steps to explain how PSO would address this specific problem.

$$\text{Max. } f(x) = \sum_{i=1}^{2} (-x_i) \cdot \sin(\sqrt{|x_i|})$$

where

$$-500 \le x_1, x_2 \le 500$$

Question 5. [15 points]

Assume you are asked to solve a regression analysis problem that you do not know the exact property of the function. Regression analysis tries to estimate the dependency among variables. In order to do that the values from data is compared to values found by the estimation function. In least square minimization, the square of the differences are minimized. Mathematically you have the collected data $z = d(x, y)$ for a set of x, y values and you are asked to estimate $z = f(x, y)$ in order to minimize $\sum_{\forall x,y} (f(x, y) - d(x, y))^2$

You know that the function can be represented by combination of addition, subtraction, multiplication, *sin*, *ln*, *exp* (e^x). Also constants in the range $[-100, 100]$ take part.

1. If you are asked to implement a tree encoded Genetic Programming solution to this problem, what will be your terminals and non-terminals.
2. Show how the mutation and crossover operations can be used to manipulate this tree to produce new solutions.

Question 6. [10 points] The Quadratic Assignment Problem (QAP) is defined as the assignment of a number of n facilities to n different locations while minimizing the product of the flow cost between every two facilities and the distances between the two locations that these facilities where assigned to. The flow costs between every two facilities is given in a matrix F while the distances between every two locations is given in a matrix D.

Ant Colony Optimization could be applied to solve this problem:

a) Mention the different parameters that need to be set to run ACO algorithm. Discuss the implication of selecting certain values (large, small, etc) for these parameters.

Answer

Mention the different parameters that need to be set to run ACO algorithm. Discuss the implication of selecting certain values (large, small, etc) for these parameters.

Number of ants: more ants more computations, but also more exploration.
Max number of iterations: has to be enough to allow convergence.
Initial pheromone: constant, random, max value, small value.
Pheromone decay parameter ρ
Parameters alpha and beta

b) Define mathematically the objective function to be used in the search if the solution is defined as a permutation **P**, where the index is the location and the element value is the facility,

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} F(P(i), P(j)) D(i, j)$$

c) What are the decisions to be taken by any ant m in every iteration?

At each step, the ant selects an assignment for one of the unassigned facilities to a location. This can be represented as a fully connected graph of (facility, location) pairs and the ant moves over this graph avoiding repeated assignments.
Or
At each step, the ant selects an unassigned facility to consider and then select a location for this facility from the list of locations.

d) How the pheromone trails τ_{ij} are applied and updated for such a problem?

Each facility is assigned a pheromone value and each location assignment for this facility is assigned a pheromone value. The pheromones are initially assigned to the same value. Each ant selects the next move based on a probability calculated based on the pheromones values (give equation).
The pheromone values are updated at the end of iterations by doing evaporations and then update based on the best solution (give equations)