VICTORIA UNIVERSITY OF WELLINGTON
TE HERENGA WAKA

EXAMINATIONS – 2022

TRIMESTER 2

AIML426

EVOLUTIONARY COMPUTATION AND LEARNING

Time Allowed: TWO HOURS

CLOSED BOOK

Permitted materials: Only silent non-programmable calculators or silent programmable calculators with their memories cleared are permitted in this examination.

Instructions: Attempt ALL EIGHT Questions. Skip to the next question if the current one is hard.

The exam will be marked out of a total of 100 marks.

Answer in the appropriate boxes as close to the questions as possible. If you write your answer elsewhere, make it clear where your answer can be found. Please use the blank pages in the end for any extra space you need, for working or for answers.

Questions

	Marks
1. Introduction to Evolutionary Computation	[5]
2. Genetic Algorithms	[5]
3. Genetic Programming	[20]
4. Evolutionary Multi-objective Optimisation	[15]
5. Evolutionary Programming and Evolution Strategy	[15]
6. Swarm Intelligence (ACO, PSO)	[10]
7. Other EC Paradigms (DE, CC, EDA)	[20]
8. EC for Computer Vision and NeuroEvolution	[10]

1. **Introduction to Evolutionary Computation** **(5 marks)**

(a) **(2 marks)** As two main types of evolutionary computation techniques, evolutionary algorithms and swarm intelligence are developed from different inspirations. Briefly describe the inspirations of evolutionary algorithms and swarm intelligence, respectively.

(b) **(3 marks)** For each of the following problems, select a method from genetic algorithm, genetic programming, and particle swarm optimisation for solving it, and *briefly explain your reasons*.

(i) Learning a model to represent the relationship between the genomic features of a fish and its growth rate without knowing the model structure.
(ii) Training the weights of a neural network for image classification.
(iii) The traveling salesman problem, which finds the tour that visits all nodes in a graph exactly once with minimal total distance.

2. **Genetic Algorithms** **(5 marks)**

Steve developed a GA to solve the knapsack problem. He uses the binary string representation, and defines the fitness function as

$$fit = totalValue - \max\{totalWeight - Q, 0\}.$$

where Q is the knapsack capacity. The evolution at each generation works as follows:

(i) Initialise an empty offspring population;
(ii) Select two parents randomly (uniform distribution) from the current population;
(iii) Apply the one-point crossover to generate two offspring, and add them into the offspring population;
(iv) If the offspring population is full, directly set it as the population of the next generation. Otherwise, go back to Step (ii).

He observed that the best individual in the population fluctuates and cannot improve over time. In addition, most individuals are infeasible, i.e., total weight is larger than Q.

List *three* ways to modify the GA components (e.g., selection, fitness evaluation, genetic operators) to improve its performance.

3. **Genetic Programming** **(20 marks)**

(a) **(2 marks)** List two tasks that standard GP can easily solve, but standard GAs cannot.

(b) **(2 marks)** In GP, the *full* method and the *grow* method are two common ways of generating initial programs. Briefly describe the differences between them.

(c) **(3 marks)** Briefly describe *three* ways to use GP to do *multi-class* classification.

(d) **(3 marks)** When designing the terminal and function sets, *sufficiency* and *closure* are important for the success of GP. Briefly describe the meaning of sufficiency and closure.

(e) **(4 marks)** GP can be used to learn dispatching rules for job shop scheduling to minimise makespan. Briefly describe the steps to *calculate the fitness* of a GP individual (i.e., dispatching rule).

(f) **(4 marks)** Anna developed a GP for the following symbolic regression problem:

$$f(x) = \begin{cases} -2x^2 + 20, & \text{if } x < 0, \\ x + 20, & \text{if } 0 \leq x < 5, \\ x^2 - x + 5 & \text{otherwise.} \end{cases}$$

For fitness evaluation, she generated 10 fitness cases, by uniformly sampling x from the range $[-5.0, 5.0)$. However, the model learned by GP performs very poorly on the test set. Explain the issues on the fitness cases and suggest *two* ways to improve it.

(g) **(2 marks)** Strongly typed GP (STGP) is an enhanced version of GP. List two advantages of STGP over standard (un-typed) GP.

4. **Evolutionary Multi-objective Optimisation** **(15 marks)**

(a) **(3 marks)** Given three points in the 2D objective space $A = (2, 1)$, $B = (1, 2)$, $C = (1, 3)$, where the two objectives are to be *minimised*, list all the three pairwise dominance relations (A vs B, A vs C, B vs C), i.e., whether they are dominated or not.

(b) **(2 marks)** List *two* advantages of evolutionary computation techniques for solving multi-objective optimisation problems.

(c) **(3 marks)** Briefly describe how Non-dominated Sorting Genetic Algorithm II (NSGA-II) addresses the design issues of fitness assignment, diversity preservation, and elitism.

(d) **(4 marks)** MOEA/D decomposes the original multi-objective optimisation problem into a set of single-objective optimisation sub-problems. The weighted sum approach is a common way for decomposition.

(i) Briefly describe how the weighted sum approach decomposes the original problem into sub-problems.

(ii) Describe a situation where the weighted sum approach cannot find the entire Pareto front, and suggest a way to resolve it.

(e) **(3 marks)** Briefly describe the steps of calculating the Inverted Generational Distance (IGD) of a solution set, assuming the true Pareto front is known.

5. **Evolutionary Programming and Evolution Strategy** **(15 marks)**

(a) **(5 marks)** Evolutionary programming (EP) algorithms were originally proposed to evolve/optimize *Finite State Machines* (FSMs). Consider the task of evolving a FSM with 3 internal states. The evolved FSM can handle a sequence of inputs with each input being one of $\{a, b, c\}$ and produce a sequence of outputs with each output being one of $\{1, 2, 3\}$. Propose a genotypic representation of the evolved FSM in the form of a chromosome. Explain why your genotypic representation can be easily utilized by the basic EP algorithm to evolve FSMs.

(b) **(3 marks)** Identify the difference between *fat-tailed* and *thin-tailed* probability distributions. Explain why the *Fast Evolutionary Programming* (FEP) algorithm decides to use a fat-tailed probability distribution to mutate solution vectors.

(c) **(3 marks)** Briefly discuss the major differences between $(\mu + \lambda)$-ES and (μ, λ)-ES algorithms. State their advantages and disadvantages towards solving both continuous and discrete optimisation problems, respectively.

(d) **(4 marks)** The 1/5 rule can be effectively utilised to adjust the level of randomness involved in mutating candidate solutions in the (1+1)-ES algorithm. Describe the main idea of this rule. Briefly describe what the $(\mu + 1)$-ES algorithm needs to do in order to use the 1/5 rule.

6. **Swarm Intelligence** **(10 marks)**

(a) **(5 marks)** In the basic ant system for pathfinding, each edge (i, j) maintains a pheromone value τ_{ij}, which is initialised to $\tau_{ij}^0 = c$, and updated after each iteration as follows:

$$\tau_{ij}^{t+1} = \rho * \tau_{ij}^t + \Delta\tau_{ij}^t.$$

(i) Briefly describe the meaning of the parameter ρ, and give its range.

(ii) Briefly describe the rationale for defining the new pheromone $\Delta\tau_{ij}^t$. *You can either give the mathematical formula, or describe the rationale by plain language.*

(iii) If the algorithm converges to poor local optima too fast, how do you modify the parameters c and ρ to improve its performance?

(b) **(5 marks)** The following shows the particle update formulas of the standard Particle Swarm Intelligence (PSO).

$$v_{id}(t+1) = w * v_{id}(t) + c_1 * r_1 * (P_{id} - x_{id}(t)) + c_2 * r_2 * (P_{gd} - x_{id}(t)),$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1),$$

where P_{id} is the dimension d of the personal best of particle i, P_{gd} is the dimension d of the global best of the entire swarm. That is, the topology is fully connected.

Answer the following two subquestions regarding velocity explosion:

(i) To avoid velocity explosion, what are the recommended value ranges for w, c_1, and c_2?

(ii) When $w = 1$, is it possible to completely prevent velocity explosion under suitable settings of c_1 and c_2? Why or why not?

7. **Other EC Paradigms (DE, CC, EDA)** **(20 marks)**

(a) **(3 marks)** *Canonical Differential Evolution* (DE) has four steps at each generation: (1) parent selection; (2) environmental selection; (3) crossover; and (4) mutation. State the order of these steps at each generation of DE.

(b) **(4 marks)** Many variants of the DE algorithm use different mutation operators to drive the search for near-optimal solutions. Consider the following continuous optimisation problem with two decision variables (i.e., x_1 and x_2):

$$\min f(x_1, x_2) = x_1^2 + x_2^2, \quad x_1, x_2 \in [-10, 10].$$

Answer the following two subquestions:

(i) Describe how each of the two mutation operators below works.

$$Mutation1 : V = X^{r_1} + F \cdot (X^{r_2} - X^{r_3})$$

$$Mutation2 : V = X^{best} + F \cdot (X^{r_2} - X^{r_3})$$

(ii) For this problem, which of the two mutation operators above may be more effective? Briefly justify your answer.

(c) **(4 marks)** For each of the following task, state whether it is better to be solved by *competitive* co-evolution or *cooperative* co-evolution.

i) Learn the policy of each robot in a robot soccer team.
ii) Learn the weights of different layers of neural network.
iii) Learn the controller of multiple modules (e.g., wheel, accelerator) of self-driving cars.
iv) Learn an image classifier along with an AI program to generate fake images.

(d) **(3 marks)** The *Estimation of Distribution Algorithm* (EDA) solves an optimization problem by keeping track of the statistics of the selected solutions from every population. Identify the key statistics to be computed respectively for the first-order EDA and second-order EDA approaches. Assume that the problems being tackled are binary optimisation problems.

(e) **(6 marks)** In order to construct accurate second-order probability models for solving binary optimisation problems, the *Mutual Information Maximisation for Input Clustering* (MIMIC) algorithm requires to compute the *conditional entropies* between some pairs of bits of selected binary vectors.

(i) Present the mathematical definition of the conditional entropy of bit 1 given bit 2 of a binary vector.

(ii) Using the selected binary vectors given below

$$x_1 = (1,0,0), x_2 = (0,1,0), x_3 = (1,1,0), x_4 = (0,0,0)$$

to compute the conditional entropy of bit 1 given bit 2.

8. **EC for Computer Vision and NeuroEvolution** **(10 marks)**

(a) **(4 marks)** The FLGP (feature learning GP) algorithm is designed to evolve a GP tree that serves as an image feature extractor. Every GP tree evolved by FLGP can contain multiple functional layers (i.e., layers with different types of function nodes for image feature extraction). Identify two functional layers supported by FLGP. For each functional layer, name two example function nodes that can be used in the respective functional layers of the evolved GP trees.

(b) **(2 marks)** Discuss why the *genetic* Convolutional Neural Network (CNN) algorithm for image processing is considered as a semi-automated approach for designing CNN architecture.

(c) **(4 marks)** The *NeuroEvolution of Augmenting Topologies* (NEAT) algorithm is a popular algorithm for NeuroEvolution. Answer the following two subquestions regarding NEAT.

(i) Identify two technical challenges addressed by NEAT to evolve neural network structures incrementally along with connection weights.

(ii) Briefly explain why disabled connection genes are not removed from the chromosomes evolved by NEAT.

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

* * * * * * * * * * * * * * *