University of Waterloo

Department of Electrical and Computer Engineering

ECE 457A Cooperative and Adaptive Algorithms

Assignment 4: Due Date – Dec 3rd, 2021

- **The assignment should be done individually**
- **You should upload your answers as a PDF file on learn before 11:30pm of the deadline date**
- **Attach any codes used for the assignment separately as a compressed file to the same dropbox**
- **You can use any programming language (Matlab and python are preferred)**
- **Works with more than 30% similarity will be checked for originality**
- **Communicate any issues or concerns with the TAs**

Question1 (5 Points)

NetLogo has a model that implements a simple **PSO** in 2-D searching space [Models Library –> Computer Science –> Particle Swarm Optimization].

a. Run experiments with different populations (30, 80), speed limits (2 and 6), particle's inertia (0.60, 0.729), personal-best (1.7, 1.494) and the global-best factor the same as personal factor. Examine the PSO algorithm's characteristics (speed of converge and ability to find global optima), report your observations.

b. What is the difference between the motion formulation of the given NetLogo implementation and the classical PSO? Explain the difference by referring to the code tab in Netlogo.

Question2 (5 Points)

Consider below an implementation of **Particle Swarm Optimization** (PSO) algorithm

```
procedure [X] = PS(max_it, AC1, AC2, vmax, vmin)
    initialize X   //usually xi, ∀i, is initialized at random
    initialize Δxi //at random, Δxi ∈ [vmin, vmax]
    t ← 1
    while t < max_it do,
        for i = 1 to N do,          //for each particle
            if g(xi) > g(pi),
                then pi = xi,       //best indiv. performance
            end if
            g = i                   //arbitrary
            //for all neighbors
            for j = indexes of neighbors
                if g(pj) > g(pg),
                    then g = j,     //index of best neighbor
                end if
            end for
            Δxi ← Δxi + φ1⊗(pi − xi) + φ2⊗(pg − xi)
            Δxi ∈ [vmin, vmax]
            xi ← xi + Δxi
        end for
        t ← t + 1
    end while
end procedure
```

a) In the above PSO implementation, which update mode (synchronous or asynchronous) for personal best and neighbors' best is adopted? Explain shortly.

b) Comment on how to change the algorithm to work on asynchronous mode if your answer to part (a) is "synchronous" otherwise to work on synchronous if your answer to part (a) is "asynchronous".

c) Considering the effect of Parameters W, C_1, C_2 in the PSO model:

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

i. What happens when C_1 is set to zero?

ii. What happens when C_2 set to zero?

iii. What is the importance of the W parameter?

Question3 (10 Points)

Implement a **genetic programming** algorithm and use it to solve the "6-multiplexer" problem:

In this problem there are six Boolean-valued terminals, {a0, a1, d0, d1, d2, d3}, and four functions, AND, OR, NOT, IF. The first three functions are the usual logical operators, taking two, two, and one argument respectively, and the IF function takes three arguments. (IF X Y Z) evaluates its first argument X. If X is true, the second argument Y is evaluated; otherwise the third argument Z is evaluated. The problem is to find a program that will return the value of the d terminal that is addressed by the two a terminals. E.g., if a0=0 and a1=1, the address is 01 and the answer is the value of d1. Likewise, if a0=1 and if a0=1 and a1=1, the address is 11 and the answer is the value of d3. The fitness of a program should be the fraction of correct answers over all 64 possible fitness cases (i.e., values of the six terminal). When "optimizing" you may wish to use only partial "test" sets however, when reporting performance you should report the performance on the full test set.

Use genetic programming to discover the solution to the 6-multiplexer problem. You should report:

a) your parameter choices

b) a plot of the progress of the best program fitness in each iteration

c) the fitness of the finalist program.

d) the tree of the finalist program.