MTRN4010 – Laboratory Exercise (weeks 10-13)

Control of mobile robots – simulation studies

Objective:

In the laboratory task, you have to implement Matlab simulations for the control of mobile robots with regard to:

1. Apply the genetic algorithms technique to obtain a shortest travel path for a mobile robot
2. Design of a fuzzy controller for a single mobile robot
3. Use the particle swarm optimization method to minimize the pose error when the mobile robot reaches its target pose.

Procedure:

Simulation preparation:

In this preparative stage, you have to write a Matlab program to:

1. Open a figure window depicting the field that a mobile robot is moving. The size of the field is user given.
2. Construct a 'car' variable (use the structure data format) representing the vertices of a hexagon.
3. Plot the initial car shape at the centre of the field and return a graphic handle for subsequent animation plots.
4. Plot the initial car position at the field centre and return a graphic handle. This handle is used to show the trace of the car.
5. Construct a loop through time, from 0 to a final time. Within the loop, move the car according to the car model given below. In this task, you can use randomly generated velocity and turn-rate commands to move the car.

$$x(t+1) = x(t) + vt\cos(\theta)$$
$$y(t+1) = y(t) + vt\sin(\theta)$$
$$\theta(t+1) = \theta(t) + \omega t$$

   where (x,y) is the centre of the car, θ is its orientation, v is the velocity command, ω is the turn-rate command, and t is the time step.

6. Update the car graphic handle and trace handle. Your figure may look as Figure 1.
7. Store your program and demonstrate to the tutor at the mark-off period.

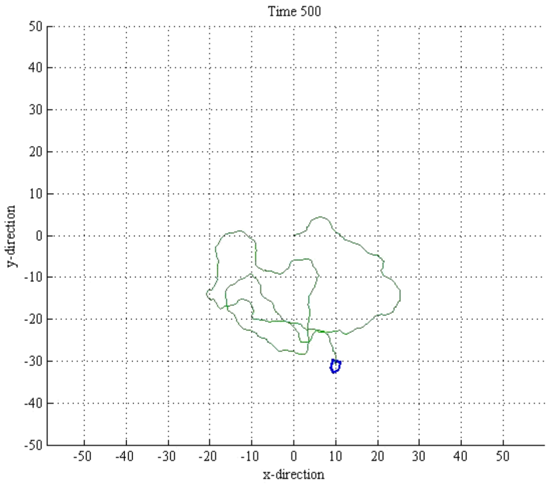


Figure 1 Car movement according to random commands.

A. Path planning:

You have to write a Matlab program, using the genetic algorithm (GA) method, to find a shortest travel distance path for the car. The car has to visit every given (more than 10) locations just once and then reaches its stopping destination.

1. In your program, you have to generate the start, stop, and visiting locations in a random manner.
2. Then you specify the GA parameters, e.g., number of chromosomes, generations, crossover, and mutation probabilities.
3. In the program, you have to include functions of fitness evaluation, selection, crossover, mutation, and the application of elitism.
4. You might have to implement a function to remove duplicated visiting locations due to the randomness arising from crossover and/or mutation operations.
5. Show the obtained path and the time history of fitness value.
6. Store your program and demonstrate to the tutor at the mark-off period. Your result may look like Figure 4 and Figure 5.

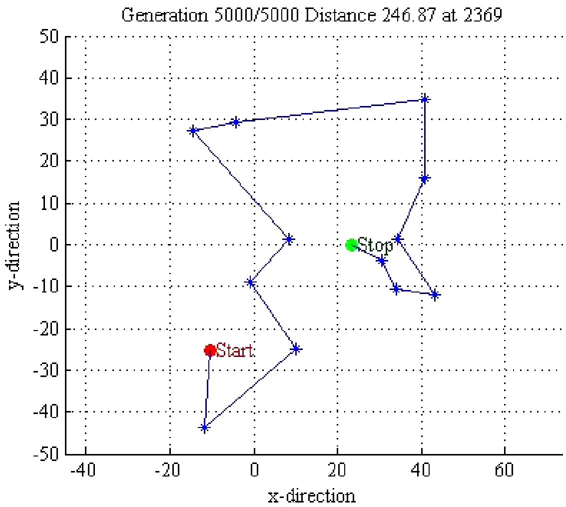


Figure 2 Planned path.

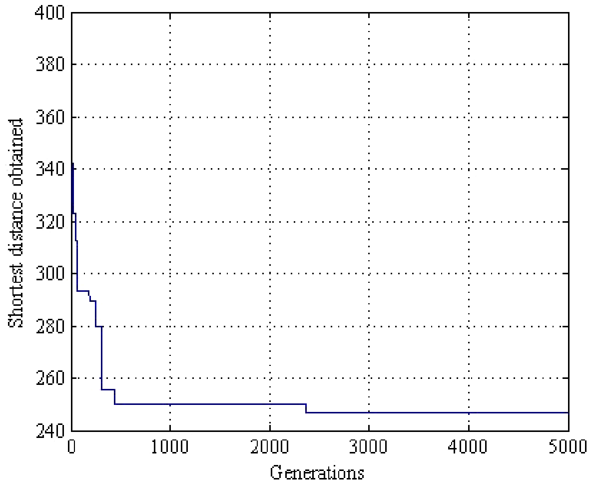


Figure 3 Evolution of the travel distance.

B. Fuzzy control:

In this problem, you have to design a fuzzy controller, using the Matlab Fuzzy toolbox, to control the motion of the mobile robot from a starting position (with specified orientation) to a final position (also with specified orientation).

1. Use the Matlab Fuzzy toolbox to establish two fuzzy inference system (fis) for the velocity and the angular turn-rate commands. In each controller, you can use 3 fuzzy membership functions.
2. You have to decide the limiting values of car motion commands while constructing the membership functions.
3. Repeat the procedure given in the 'Simulation preparation' stage and include commands to import the fis.
4. Evaluate the fis to obtain velocity and turn-rate commands and use it to move the car in the simulation run. Your result may look like Figure 4.
5. Construct a virtual target that gradually moves at some distance (you can arbitrarily set the value at this stage) from the rear of the target. Move the car to track the virtual target. Your result may look like 5.
6. Store your program and demonstrate to the tutor at the mark-off period.

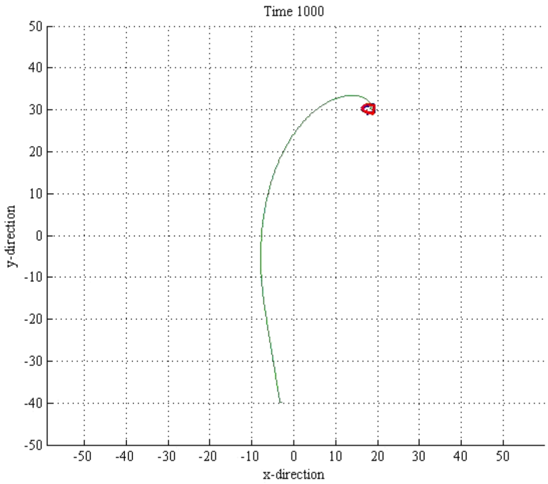


Figure 4 Car motion under fuzzy control.

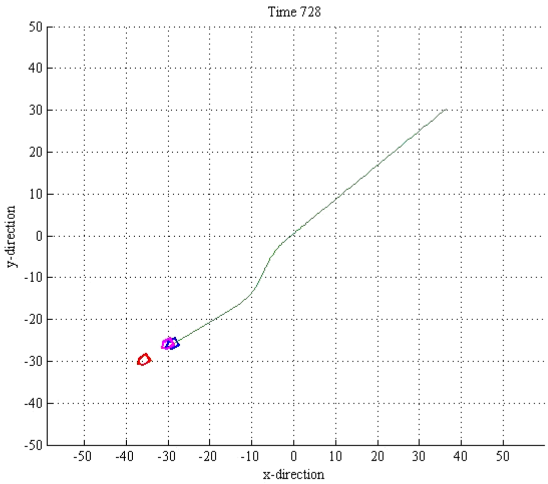


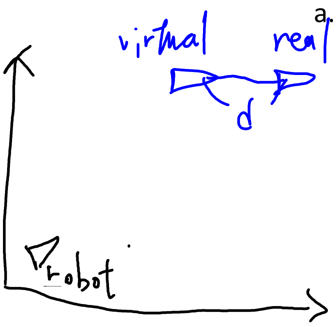
Figure 5 Car tracking virtual target using fuzzy control.

C. Optimizing Fuzzy control:

In this task, implement PSO to optimize the fuzzy system based mobile robot control in part B. Recall that an arbitrary distance was used as a virtual target that guides the mobile robot motion. It is envisioned that different settings will give different final pose error (the square root of the sum of squares of x, y, and angular error between the mobile robot and target). Your task is to modify the program developed in Part B to incorporate a best choice for the virtual target distance.

The following is an example pseudo code.

1. Define: filed range where the mobile robot travels, simulation time step, termination time
2. Initialize: figure window, mobile robot, virtual target, and target, import fuzzy systems
3. Define PSO parameters: search range, initial particles, inertia and random coefficients
4. PSO loop:
    a. For each generation
        i. For each particle
            1. Assign particle to virtual target distance
            2. Loop through time (as in Part B)



(error)
Fitness $= \sqrt{(x_r - x_t)^2 + (y_r - y_t)^2 + (\theta_r - \theta_t)^2}$

Particle $= d$

for generation

for particle

$error = \sqrt{(x_v - x_t)^2 + (y_v - y_t)^2 + (\theta_v - \theta_t)^2}$

$V = wV + c_1 r_1 (g - x) + c_2 r_2 (L - X)$

$X = V + X$

end

end

3. Calculate mobile robot error to actual target
4. Check for PSO gbest and pbest parameters (display the gbest value on screen as an indication of objective function improvements)
5. Update PSO velocity, position
6. Repair out-of-range particles

ii. End for particle

b. End for generation

5. Use the optimal virtual target distance to show the result (use a single simulation run with the optimal virtual target distance).

Note: because of the large number of simulation loops, the program may take a longer time to finish. You are advised to attempt this question as your homework.

Demonstrate your result, in particular the time history of fitness value (similar to Fig. 3), to the tutor.

Assessment:

This exercise carries 30 out of 100 of the total laboratory exercise marks for the course. Your programs will be marked during your laboratory class in week 13. You have to show to the tutor that your programs are functioning. Also provide short answers to the following questions (you have to try out modifying your program in order to gain an insight of the question/answer):

1. The advantages in using fuzzy control.
2. What is the effect of varying the number of membership functions
3. What is the effect of using different de-fuzzification methods?
4. In the path planning problem, how the number of positions to be visited affects the quality of the solution?
5. Comment, the use of GA crossover and mutation procedures, on the efficiency in producing improved solutions.
6. What is the advantage in using the PSO algorithm over using the GA with regard to programming complexity?

Marks:

Preparation: 5 marks

Part A: 5 marks

Part B: 5 marks

Part C: 10 marks

Short questions: 1 each, maximum 5 marks.