MTRN4010-Lab Exercise (weeks7 - 10) Control of mobile robots – simulation studies

Runkai Zhao, z5146927

1. Discuss the effect of using different numbers of hidden layer neurons on the model accuracy.

Answer:

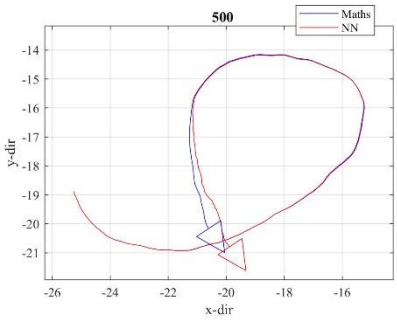
With the same parameters of neural networking,

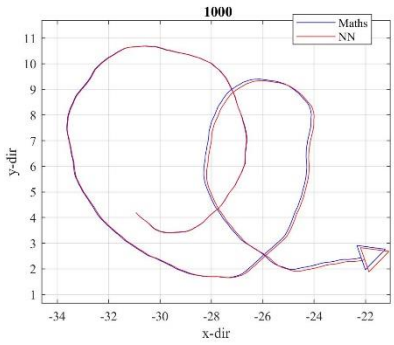
Number of hidden layer neurons	Order of trials	Operating time (seconds)	Best performance (MSE)
10	1	3	2.7783e-6
	2	<1	1.8917e-5
	3	1	1.2124e-5
20	1	5	7.46e-9
	2	3	4.18e-7
	3	5	2.03e-8
30	1	2	2.51e-7
	2	2	8.39e-8
	3	8	8.89e-9
40	1	7	2.88e-9
	2	6	4.73e-9
	3	8	3.34e-9
50	1	14	6.67e-9
	2	11	2.26e-9
	3	17	3.15e-9

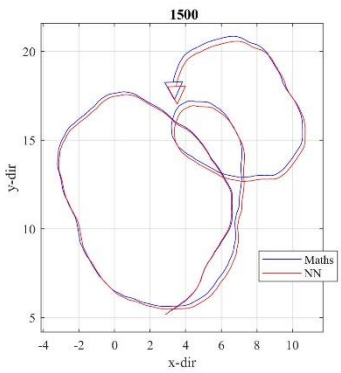
As the number of hidden layers neurons raising up, it is clear to see that the running time will be increased. The mean squared error will be minimized and the better performance of validation and training will be reached. However, this could not ensure to acquire a best neural networking model, because the training process may be terminated by the threshold or the minimum error gradient satisfied. The trajectory created by trained neural networking gets more consistent with the one created by the math model. Additionally, the number of hidden layer neurons is proportional to the complexity of the networking, so the test set error will be increased as the number of hidden layer neurons increases.

2. What is the effect of varying the number of training samples on the model accuracy?

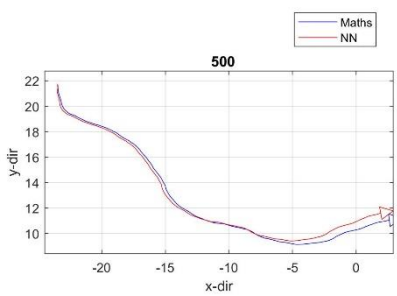
Answer: For testing and validation, the number of training samples is increased from 500 to 1500 by the step of 500, the trajectory provided by trained neural networking highly matches up to the one provided by the math model as the training sample size going up.

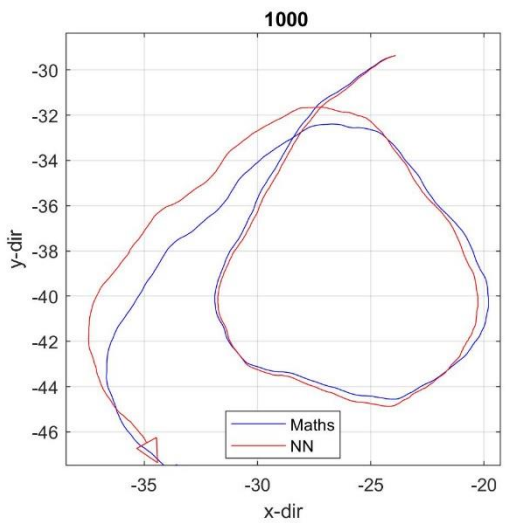


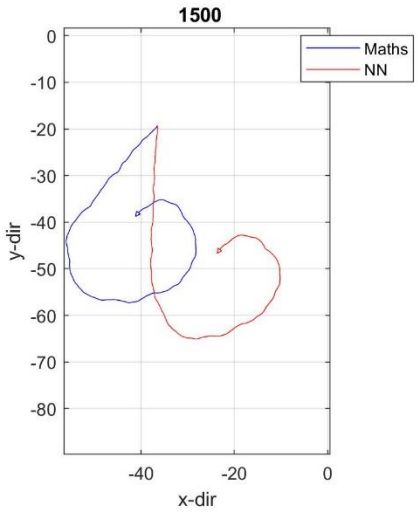




From the followed three output figures with different number of training samples, but the velocity and angular rate will be randomly generated within a different range from the training samples. For this situation, it is clear that increasing the number of training samples is unable to ensure the high quality of the neural networking model. The number of training samples is increased from 500 to 1500 by the step of 500, the trajectory provided by trained neural networking does not effectively match up to the one provided by the math model as the training sample as going up.







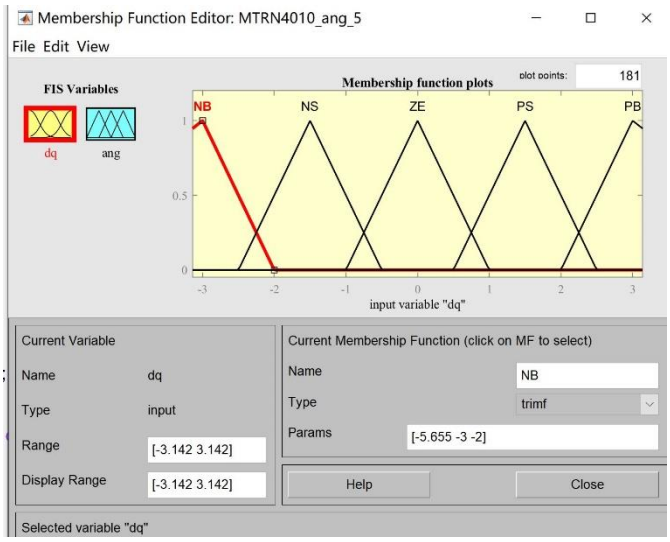
3. The advantages in using fuzzy control.

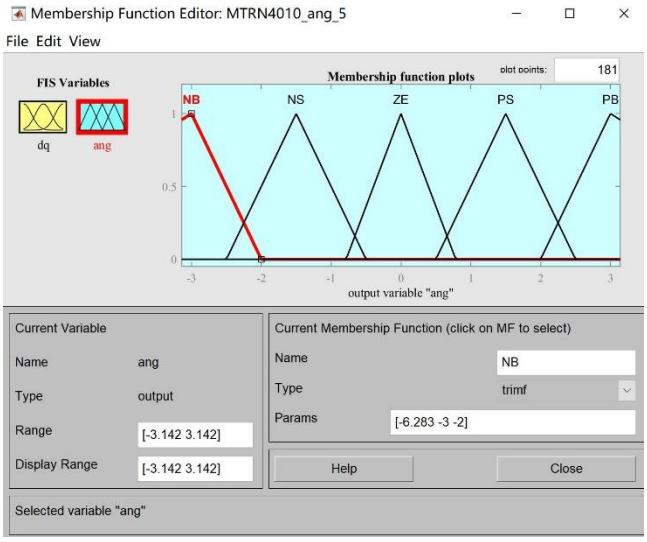
Answer: The advantages of fuzzy control are listed as below:

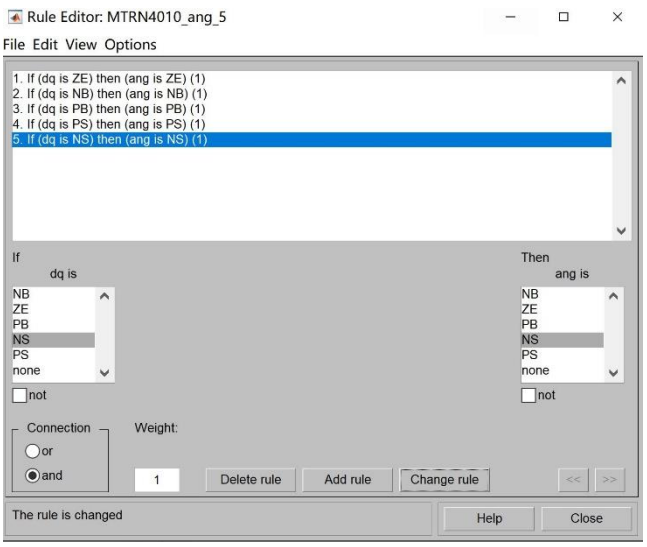
1) The fuzzy logic control is simple and flexible to change and develop for many applications.
2) It can solve the system with the inputs polluted by noise or uncertainties. The controller doesn't require accurate input data.
3) It can also handle multiple different scenarios.

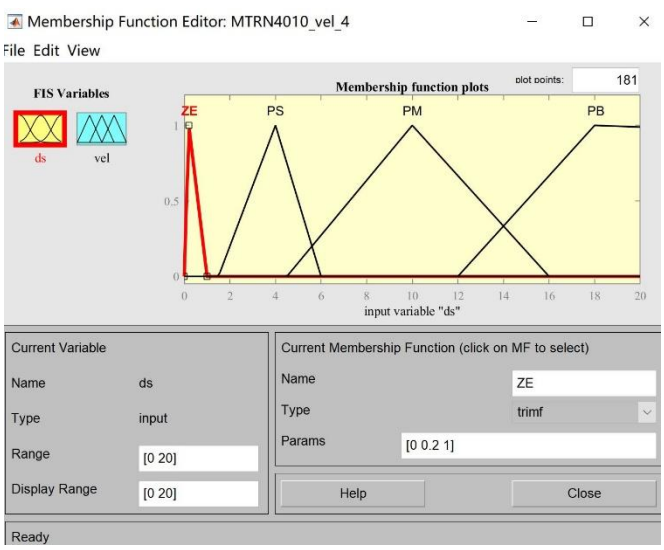
4. What is the effect of varying the number of membership functions?

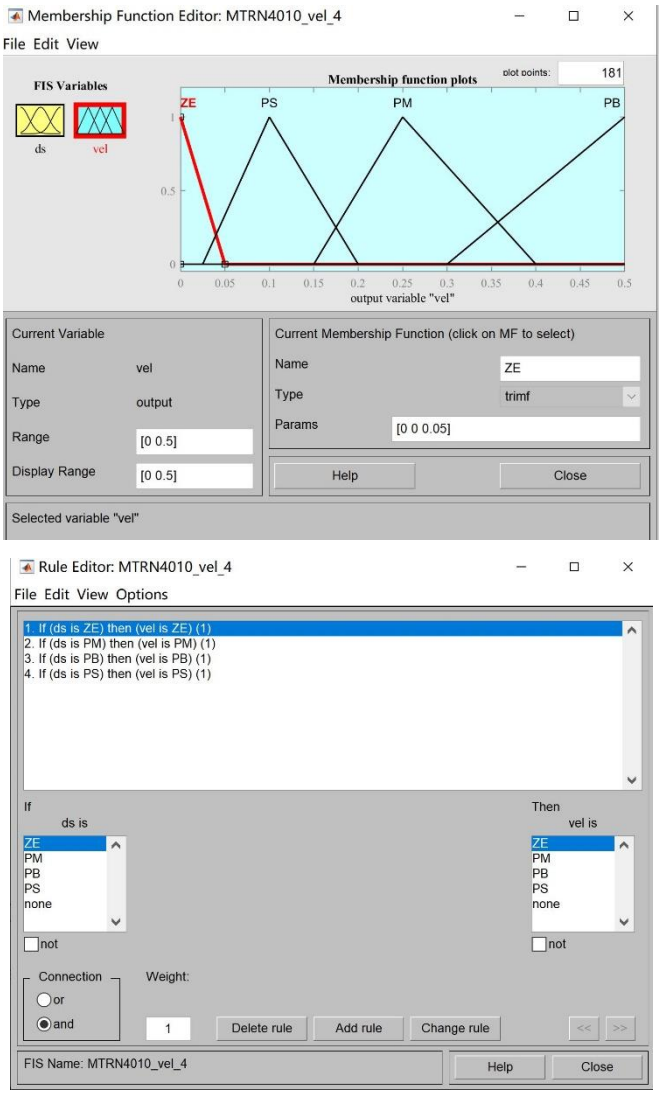
Answer: Increasing the number of member functions and rules in FLC will effectively enhance the quality of the system, because the system is able to handle multiple different situations or classifications. But the number of member functions depends on how we solve the problem and "IF…, THEN…". The changed fuzzy control designer for velocity and angular rate has been shown below.



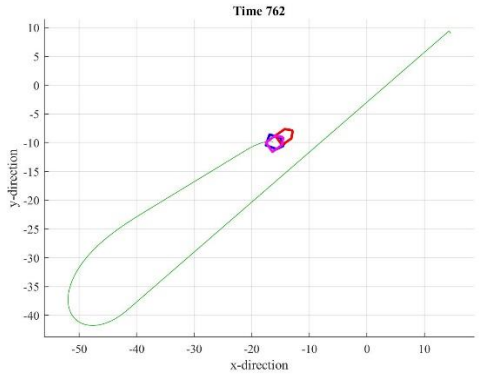

Membership Function Editor: MTRN4010_ang_5
File Edit View
FIS Variables
dq
ang
Membership function plots
plot points:
181
NB
NS
ZE
PS
PB
output variable "ang"
Current Variable
Name
ang
Type
output
Range
[-3.142 3.142]
Display Range
[-3.142 3.142]
Current Membership Function (click on MF to select)
Name
NB
Type
trimf
Params
[-6.283 -3 -2]
Help
Close
Selected variable "ang"


Rule Editor: MTRN4010_ang_5
File Edit View Options
1. If (dq is ZE) then (ang is ZE) (1)
2. If (dq is NB) then (ang is NB) (1)
3. If (dq is PB) then (ang is PB) (1)
4. If (dq is PS) then (ang is PS) (1)
5. If (dq is NS) then (ang is NS) (1)
If
dq is
NB
ZE
PB
NS
PS
none
not
Then
ang is
NB
ZE
PB
NS
PS
none
not
Connection
or
and
Weight:
1
Delete rule
Add rule
Change rule
<<
>>
The rule is changed
Help
Close

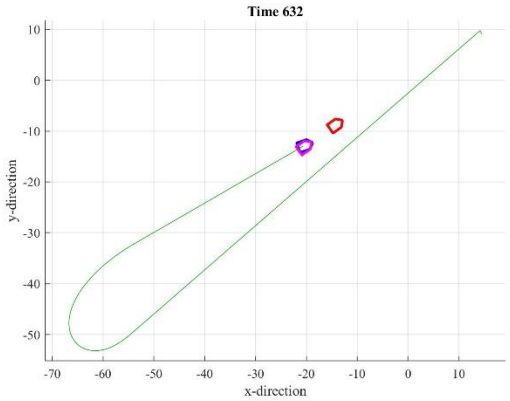

Membership Function Editor: MTRN4010_vel_4
File Edit View
FIS Variables
ds
vel
Membership function plots
plot points:
181
ZE
PS
PM
PB
input variable "ds"
Current Variable
Name
ds
Type
input
Range
[0 20]
Display Range
[0 20]
Current Membership Function (click on MF to select)
Name
ZE
Type
trimf
Params
[0 0.2 1]
Help
Close
Ready



The following figure has shown the path that is worked out by the fuzzy logic controller with three membership functions.



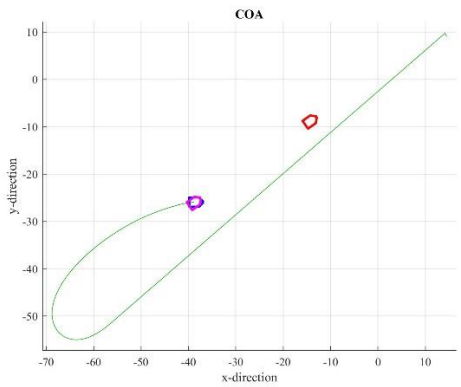
The following figure has shown the path that is worked out by the fuzzy logic controller with four membership functions. Apparently, in this situation, the car could catch up the virtual target quicker than the above one.

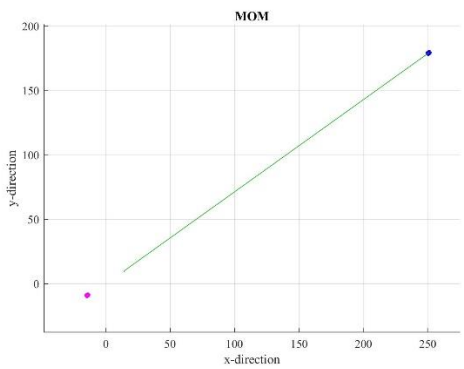


5. What is the effect of using different de-fuzzification methods?

Answer: COA and MOM are the most common methods for defuzzification. In order to compare the effect of using different defuzzification methods of COA and MOM, the defuzzification methods of fuzzy control designer for angular rate could separately be set to COA and MOM. With a same fuzzy controller of velocity, the plots of the car trajectory are shown below.

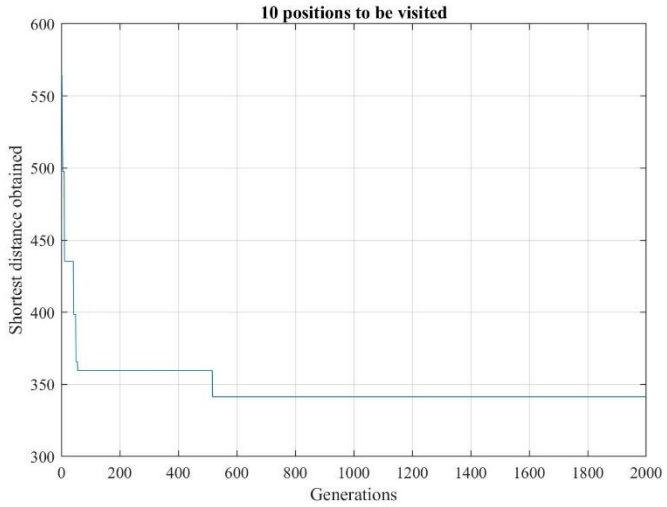
It can be seen that the path by MOM method is far away from the target and the path by COA can catch up the virtual target. An FLC based on the COA produces a better steady-state performance and a lower mean square error than that based on the MOM. COA is also a common method for defuzzification.

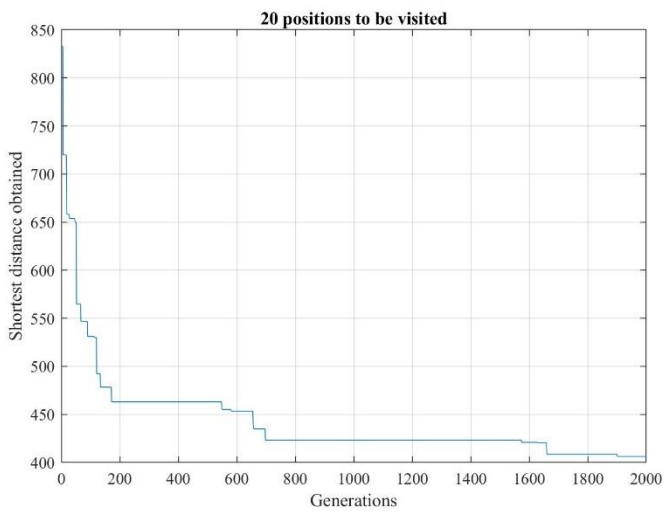


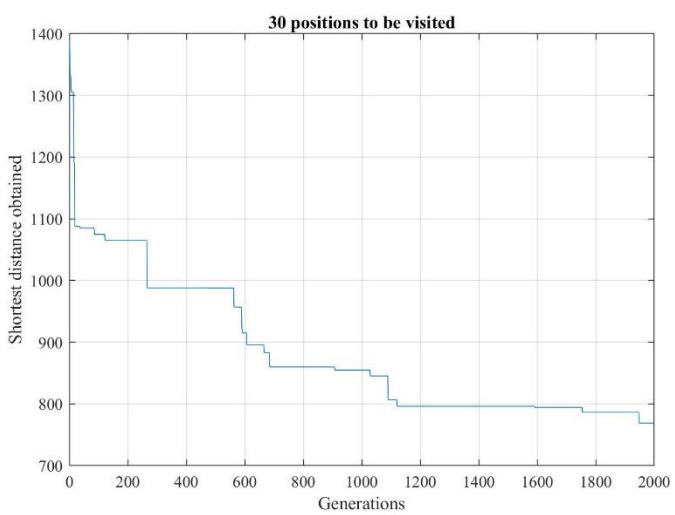


6. In the path planning problem, how the number of positions to be visited affects the quality of the solution?

Answer: Genetic algorithm with the abilities of selection, crossover and mutation offers a fast method to find the shortest distance for a car within the limited generations, but the greater number of positions to be visited requires more generations to approach the desired optimal. This will increase the complexity of process and running time. Therefore, under the same conditions, like the number of generations or other parameters, increasing the number of positions will decrease the quality of the GA solution. The validation of this result is shown as following.







7. Comment, the use of GA crossover and mutation procedures, on the efficiency in producing improved solutions.

Answer: In path planning problem, the most direct way to get the shortest distance from the start point to the end point is going through each possible path and finding out the minimization of these. The times we

have to calculate the distances is the factorial of the number of positions, which is a quite large number if we have many positions need to pass. Obviously, this method is time and computation consuming. The genetic algorithm is a nature-inspired metaheuristic. Although it could not promise that the optimal can be figured out, it provides a more efficient method to approach the optimization problem than iterative methods.

8. What is the advantage in using the PSO algorithm over using the GA with regard to programming complexity?

Answer: Referred by the MTRN4030 lecture material, the GA and PSO programming procedures have been demonstrated as the below. Beside of selection process, PSO does not have the genetic operation like GA, such as cross-over and mutation. The next position of particle is determined by the velocity calculated from the current position, local best and global best shared from others, which can be implemented through one mathematic equation with random gain factors. Compared with GA, the PSO is easier to implement and there are less parameters need to be changed.

Algorithm 1 Genetic Algorithm

Objective function $f(\mathbf{x})$, $\mathbf{x} = [x_1, \cdots, x_n]^\top$,
Encode the solution into chromosomes (binary strings)
Define fitness F (e.g., $F \propto f(x)$ for maximization)
Generate the initial population
Initial probabilities of crossover (p_c) and mutation (p_m)
while $t <$ Max number of generations **do**
  Generate new solution by crossover and mutation
  if $p_c > rand$ **then**
    Crossover
  end if
  if $p_m > rand$ **then**
    Mutate
  end if
  Accept the new solutions if their fitness increase
  Select the current best for new generation (elitism)
end while
Decode the results and visualization

Algorithm 2 Particle Swarm Optimization

Objective function $f(\mathbf{x})$, $\mathbf{x} = [x_1, \cdots, x_n]^\top$,
Initialize locations $\mathbf{x}_i$ and velocity $\mathbf{v}_i$ of n particles.
Find $f(\mathbf{g}^*)$ from $\min\{f(\mathbf{x}_1, \cdots, f(\mathbf{x}_n)\}$ (at $t = 0$)
while stopping criterion not met **do**
  $t = t + 1$ (pseudo time or iteration counter)
  for loop over all n particles and all p dimensions **do**
    Generate new velocity $\mathbf{v}_i^{t+1}$
    Calculate new locations $\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^{t+1}$
    Evaluate objective functions at new locations $\mathbf{x}_i^{t+1}$
    Find the current best $\mathbf{x}_i^*$ for each particle $\mathbf{x}_i^{t+1}$
  end for
  Find the current global best $\mathbf{g}^*$
end while
Output the final results $\mathbf{g}^*$ and $f(\mathbf{x}^*)$