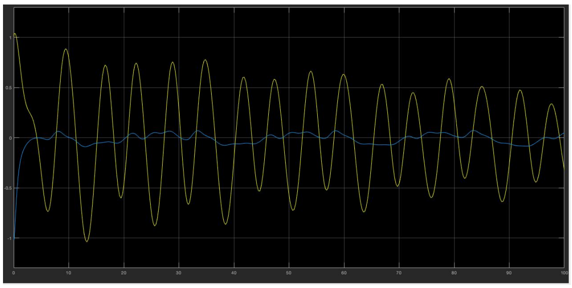
MLRC Assignment 2

ASarthak Bhagat

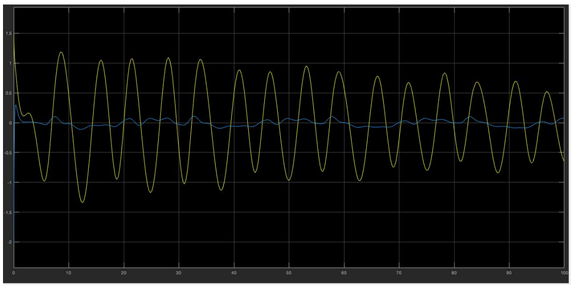
April 2019

1 Question 1 (a): L=10

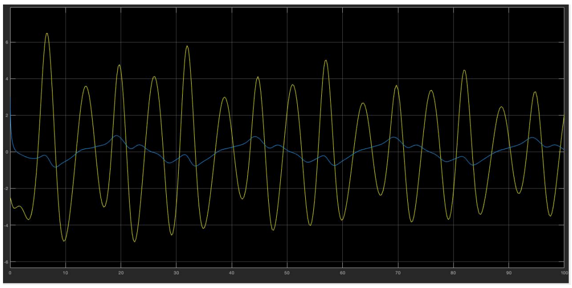
1.1 Without robust modification



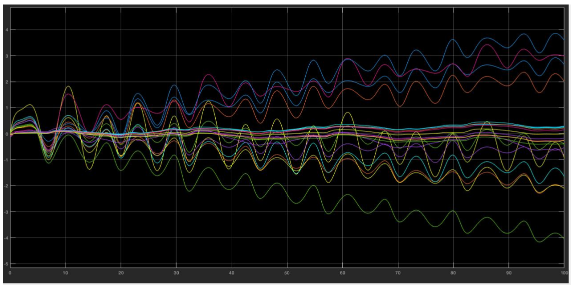
(a) e(t)



(b) r(t)

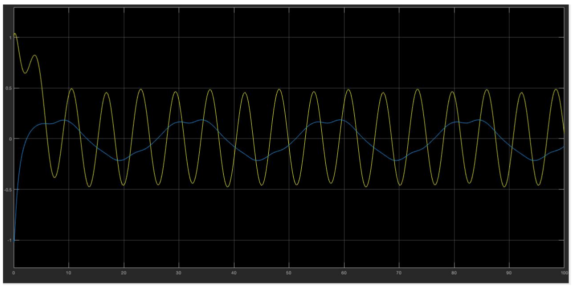


(c) τ(t)

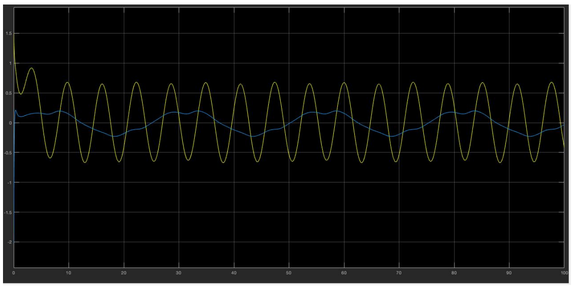


(d) $\hat{\mathbf{W}}$(t)

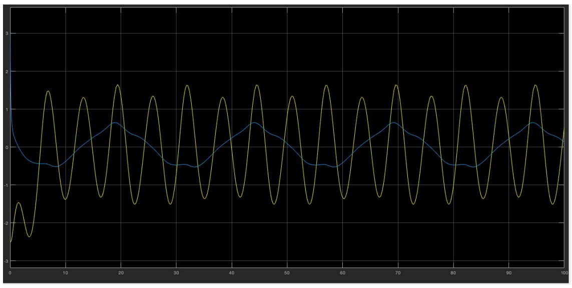
1.2 σ-mod



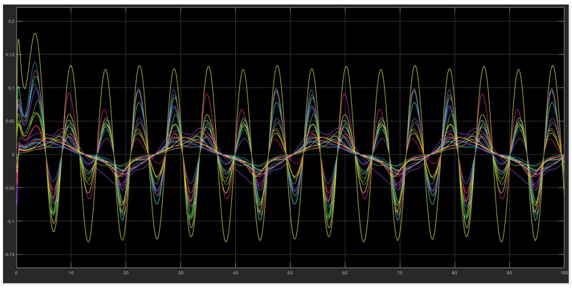
(a) e(t)



(b) r(t)

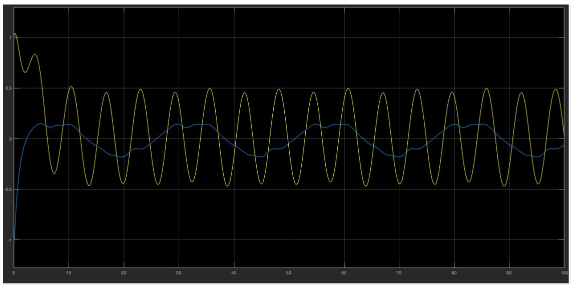


(c) τ(t)

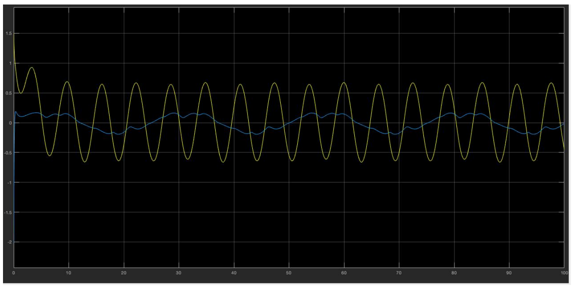


(d) $\hat{\mathbf{W}}$(t)

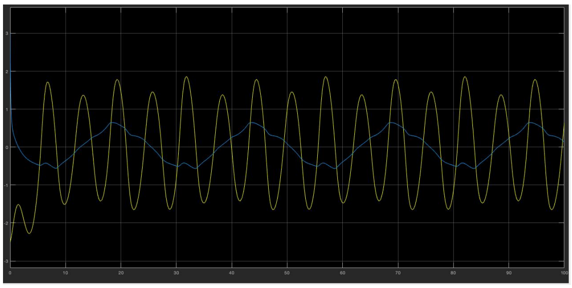
1.3 e-mod



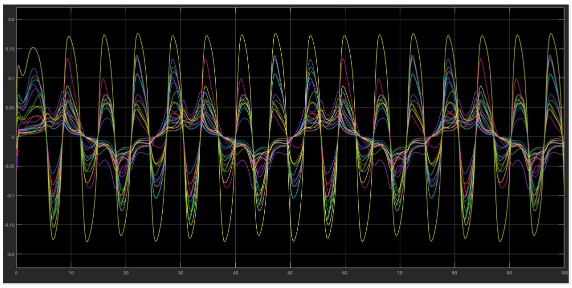
(a) e(t)



(b) r(t)



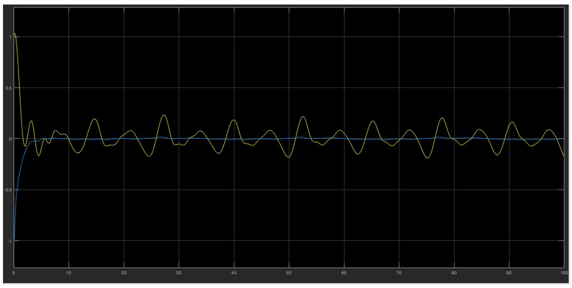
(c) τ(t)



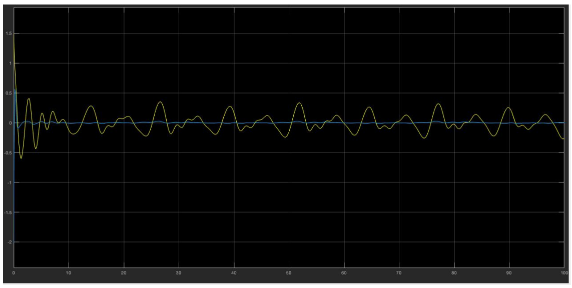
(d) $\hat{\mathbf{W}}$(t)

2 Question 1 (b): L=100

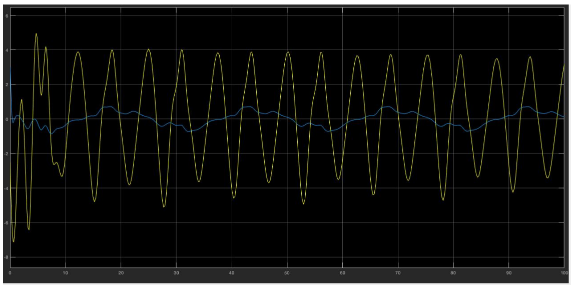
2.1 Without robust modification



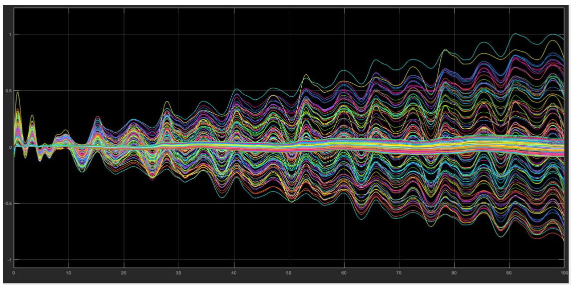
(a) e(t)



(b) r(t)

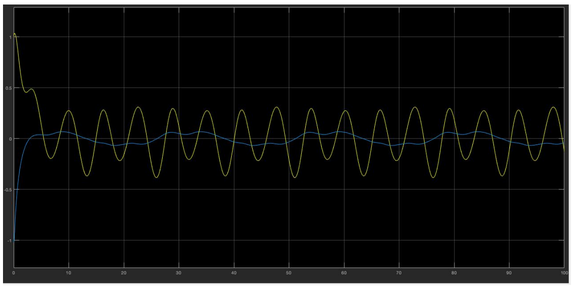


(c) τ(t)

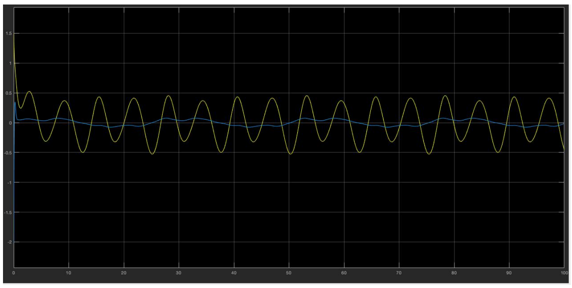


(d) $\hat{\mathbf{W}}$(t)

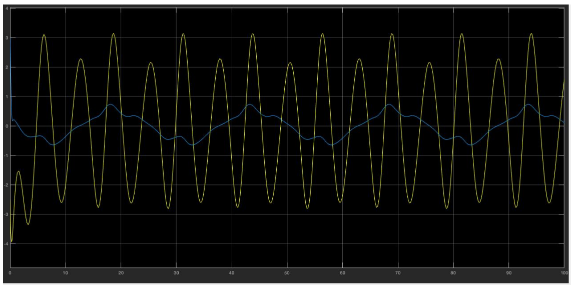
2.2 σ-mod



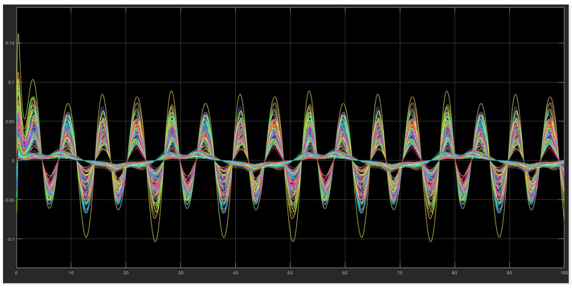
(a) e(t)



(b) r(t)

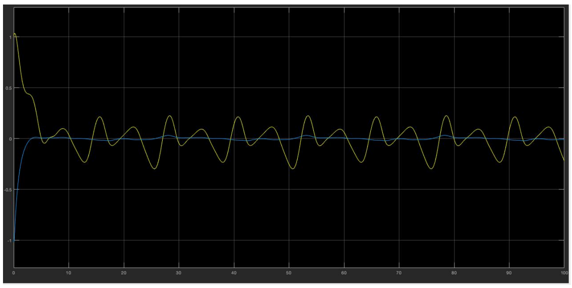


(c) τ(t)

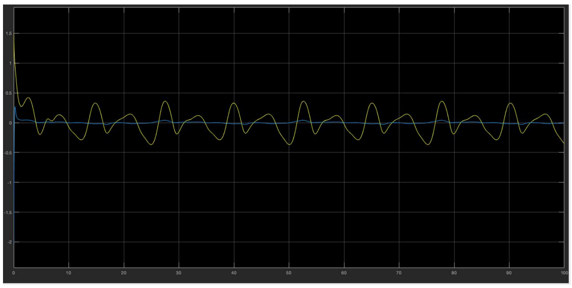


(d) $\hat{\mathbf{W}}$(t)

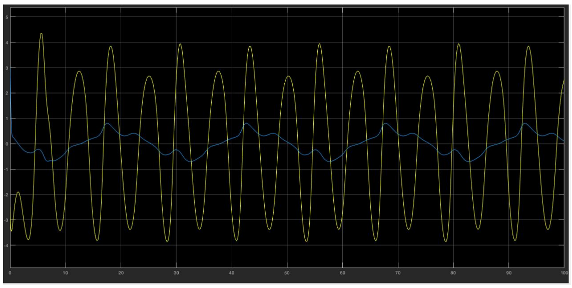
2.3 e-mod



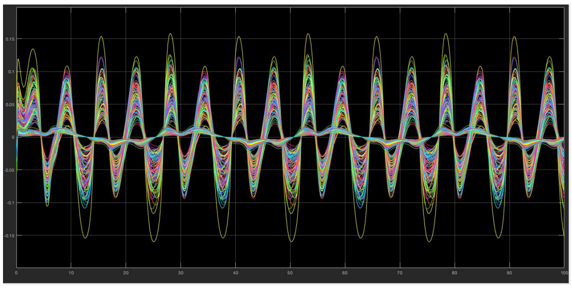
(a) e(t)



(b) r(t)



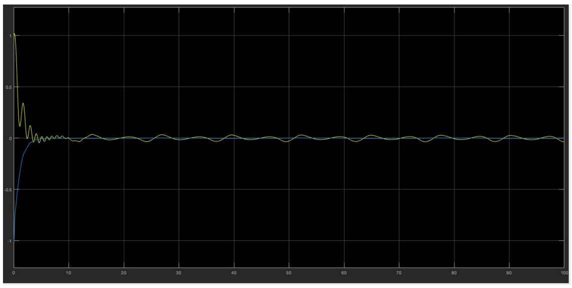
(c) τ(t)



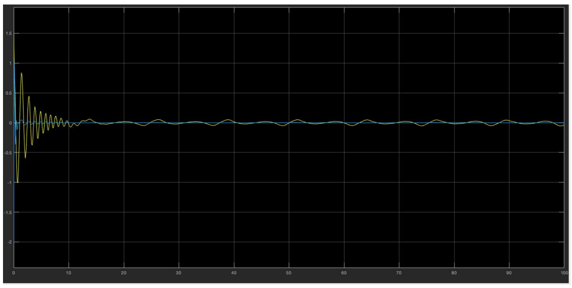
(d) $\hat{\mathbf{W}}$(t)

3 Question 1 (b): L=500

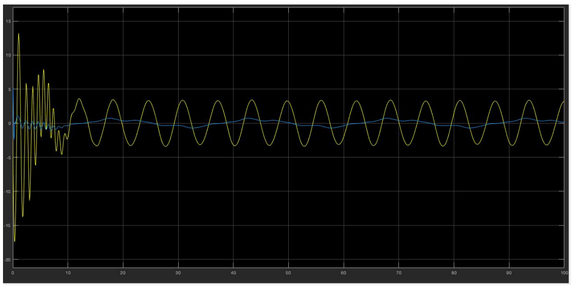
3.1 Without robust modification



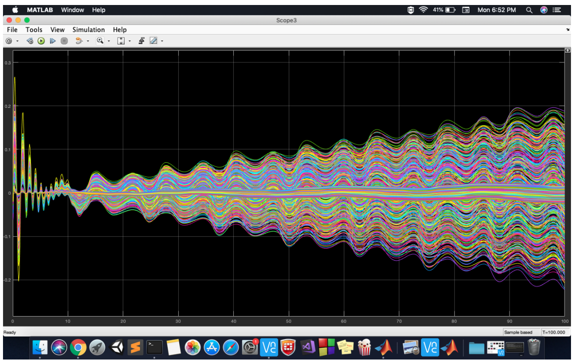
(a) e(t)



(b) r(t)

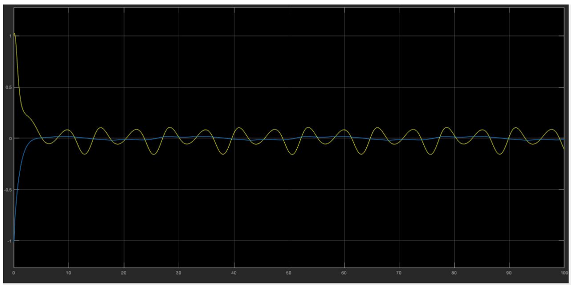


(c) τ(t)

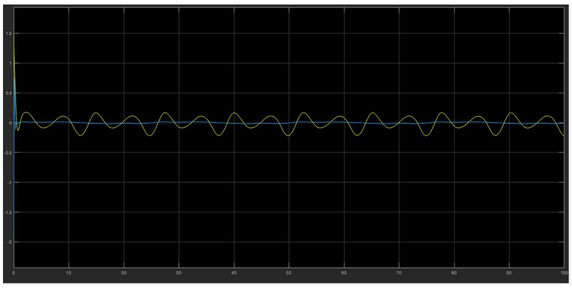


(d) $\hat{\mathbf{W}}$(t)

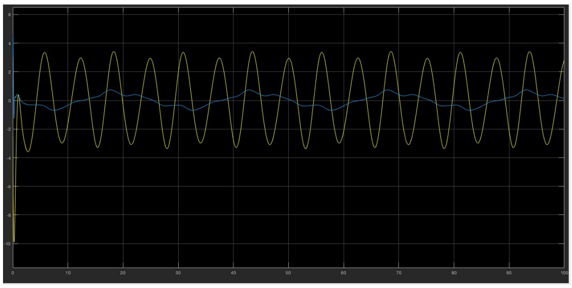
3.2 σ-mod



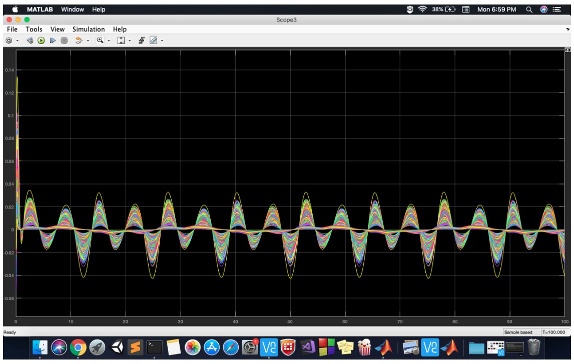
(a) e(t)



(b) r(t)

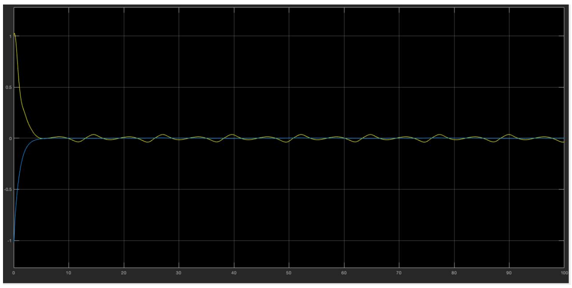


(c) τ(t)

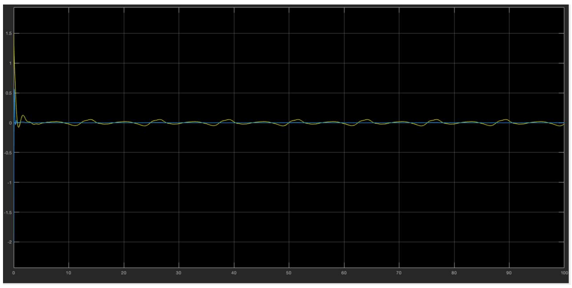


(d) $\hat{\mathbf{W}}$(t)

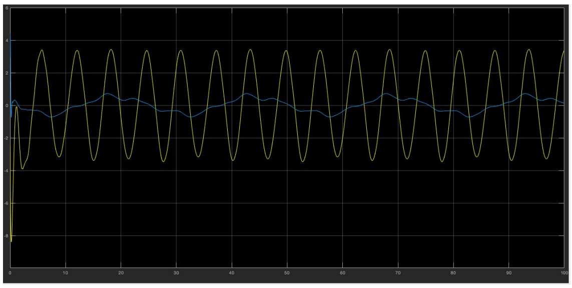
3.3 e-mod



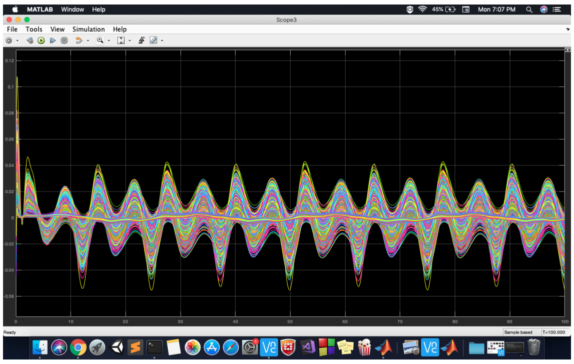
(a) e(t)



(b) r(t)



(c) τ(t)



(d) $\hat{\mathbf{W}}$(t)

4 Question 1 (c)

Yes, Parameter drift is clearly visible in figures without robust modification. Certain values in the vector $\hat{\mathbf{W}}$(t) can be observed to move away from zero towards infinity. This is because as the term $\hat{\mathbf{W}}$(t) does not occur in our final derivative of Lyapunov equation. Hence, we do not constraint them and so they can go to infinity.

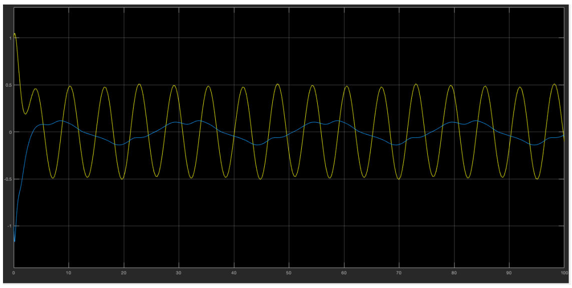
Yes, this parameter drift is greatly reduced when we use robust modifications. Clearly, the term $\hat{\mathbf{W}}$(t) does not move too much away from zero on adding the feedback term to the updation strategy. This is because on adding this term to the update strategy, we get a term $\hat{\mathbf{W}}$(t) also in the derivative of Lyapunov function and this term needs to be constraint (below a certain bound) for the system to be stable.

Yes, we can clearly observe unlearning effect in case of σ-mod which is later reduced in e-mod. This can be observed from the curve of $\hat{\mathbf{W}}(t)$, where the learned weights are unlearned by some amount when the updation condition is not satisfied. The effect reduces in e-mod as the magnitude of the unlearning effect is reduced by multiplying another term which is the magnitude of error to the feedback term. This term is really small when the error is small and hence, unlearning magnitude is also reduced when the model has learned significantly and reduced the errors to a low value.

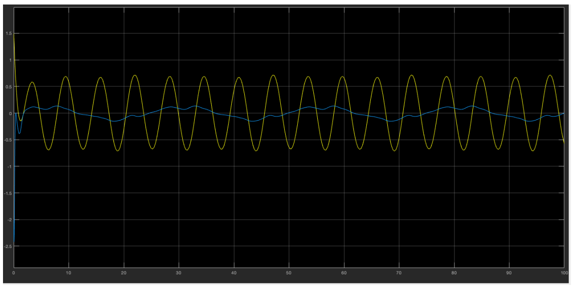
We observe that the performance is of the order given by
L=500 > L=100 > L=10.
This can be observed from figures of errors for each of cases. As the number of neurons increases, the time required for reducing loss close to zero and the error at steady state reduces. Neural networks (here one layered) are used to approximate functions and hence, using more number of neurons for our model helps us approximate the required functions more accurately and hence, a increase in performance is observed.

5 Question 2

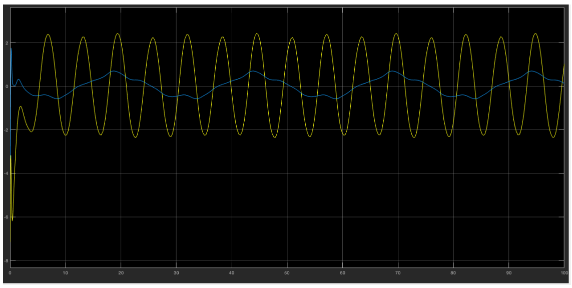
5.1 L=10



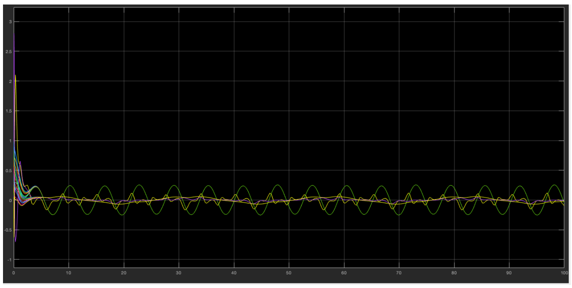
(a) e(t)



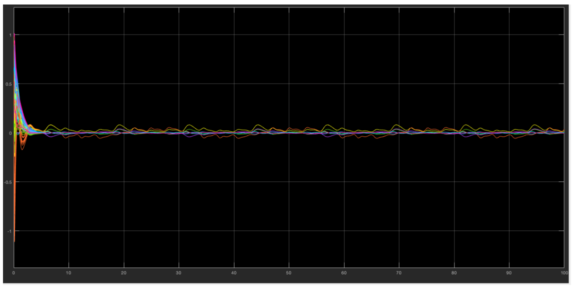
(b) r(t)



(c) τ(t)

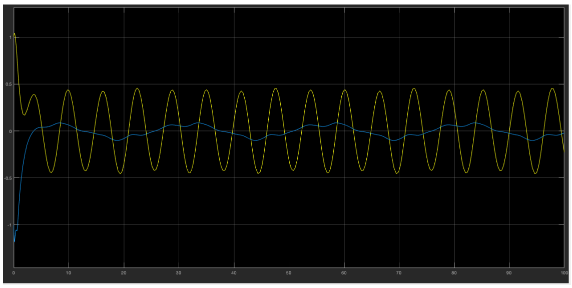


(d) $\hat{\mathbf{W}}$(t)

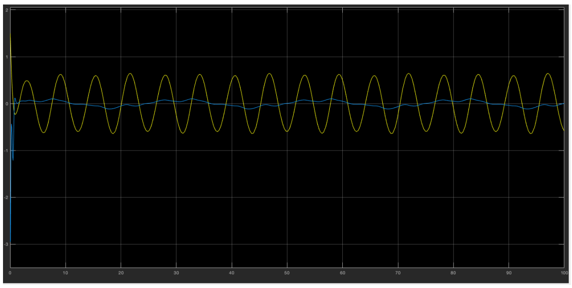


(e) $\hat{\mathbf{V}}$(t)

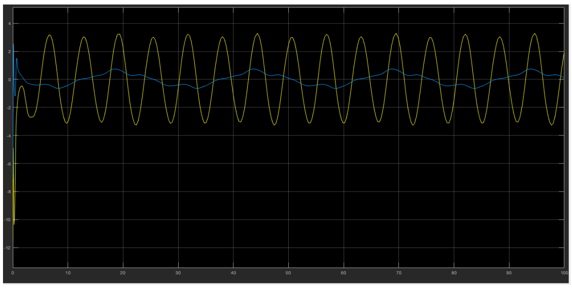
5.2 L=20



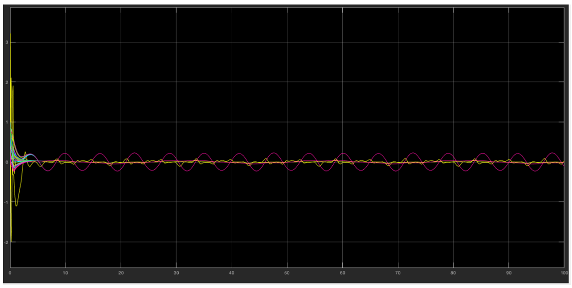
(a) e(t)



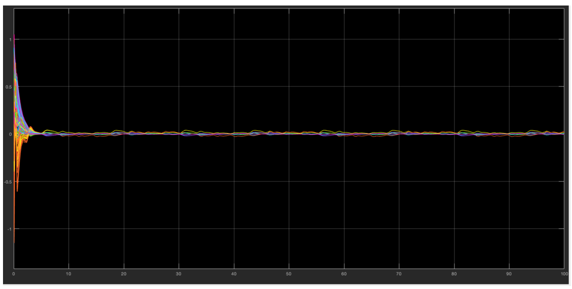
(b) r(t)



(c) τ(t)



(d) $\hat{\mathbf{W}}$(t)



(e) $\hat{\mathbf{V}}$(t)

We can clearly observe that the performance in terms of the convergence rate of the weights and errors increases as we increase the number of neurons in layers.