Simulation of a Distributed Control of Multiagent Systems with Unknown Nonlinearities

Diego Felipe Mayorga Gomez ID: 40075927
Electrical and Computer Engineering
Concordia University
Montreal, Canada
d_mayorg@live.concordia.ca

***Abstract*—This project replicates the simulation results and provides a synthesis of the theory proposed in [1]. Where the cooperative tracker problem is simulated, by synchronizing all follower agents with the leader agent in a strongly connected digraph. All of the agents have unknown nonlinearities which are compensated by a radial basis function neural network. Simulation results are shown in the end depicting the adaptiveness of the distributed control protocol.**

I. Introduction

In recent years, control of multiagent systems has raised interest among researchers due to the many advantages of adopting a distributed approach. In applications where a functionality or a system have a high complexity and one can assign multiple vehicles to have simpler laws. These agents can work cooperatively and achieve a desired functionality. This reduction in complexity could be one of the advantages that to exploit in this approach, since a less complex system may imply lower operational costs, as well as more flexible and adaptable. [2]

One of the main topics in control of multiagents is consensus control. The control protocols designed in a consensus approach is that for all of the agents in a communication network, where the agents only have the information of its local neighbours states, have as a final goal an agreement between all of the agents with this limited information.

In this project the objective is for the follower agents to synchronize with the trajectories of the leader agent. The states of the leader agent are provided to at least one of the follower agents and the agents are all strongly connected in a directed graph topology. This document will continue with Section II, where we will formulate the problem and establish the framework where all of the agents will synchronize. Section III shows the proposed control protocols in [2], this is a decentralized approach. Section IV will show the simulation and results, it will depict two different communication topologies maintaining the dynamics of the agents.

II. Problem Formulation

As per [1] Let $G = (V, E)$ represent a directed graph with $V = \{v_1, v_2, \ldots, v_N\}$, which represent each agent, and E is the set of edges (v_i, v_j), this denotes any communication topology of interest. Since each agent can only communicate its state to its neighbours we define an adjacency matrix $A = [a_{ij}]$ where $a_{ij} > 0$ if $(v_j, v_i) \,\epsilon\, E$ and $a_{ij} = 0$, otherwise $i = 1, 2, \ldots, N$. Additionally, we define a matrix $B = diag[b_1, b_2, \ldots, b_N]$ where $b_i > 0$ if the agent i has access to the leader and $b_i = 0$ if it is not the case. For the dynamics of the agents we consider a network of N follower agents and they are structured as follows:

$$\begin{cases} \dot{x}_{i1} = x_{i2} \\ \dot{x}_{i2} = x_{i3} \\ \quad\vdots \\ \dot{x}_{in} = f_i(x_{i1}, x_{i2}, \ldots, x_{in}) + u_i + w_i \end{cases} \tag{1}$$

Where, $i = 1, 2, \ldots, N$ represents the number of agents, and $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T$ is the state vector of the follower agents given n as the number of states of each agent. Similarly, the leader agent will have the following dynamics:

$$\begin{cases} \dot{x}_{01} = x_{02} \\ \dot{x}_{02} = x_{03} \\ \quad\vdots \\ \dot{x}_{0n} = f_0(x_{01}, x_{02}, \ldots, x_{0n}) \end{cases} \tag{2}$$

The functions $f_i(x_{i1}, x_{i2}, \ldots, x_{in})$ and $f_0(x_{01}, x_{02}, \ldots, x_{0n})$ are unknown nonlinear functions, that are smooth and piece-wise continuous in time t. This assumption is important, as we wish to approximate these unknown function by means of a Radial Basis Function Neural Network (RBFNN) as shown in Fig. 1 with one hidden layer as to compensate for the nonlinearities as to have a bounded synchronization error between agents and a leader and that this bound can be reduced arbitrarily.

The control design to bound this error is base on:

Lemma 1 [1] : Let $x(t)$ be a smooth function of t and assume that $x(0), \dot{x}(0), \ldots, x^{(n-1)}(0)$ are all bounded. Define:

$$s(x,t) = \left(\lambda + \frac{d}{dt}\right)^{n-1} x(t) \tag{3}$$

where $\lambda > 0$ is a constant. If there exists a scalar $\Phi > 0$ such that $|s(x,t)| \leq \Phi$, then there exists a finite T_0 that is dependent on the values of $x(0), \dot{x}(0), \ldots, x^{(n-1)}(0)$, such that

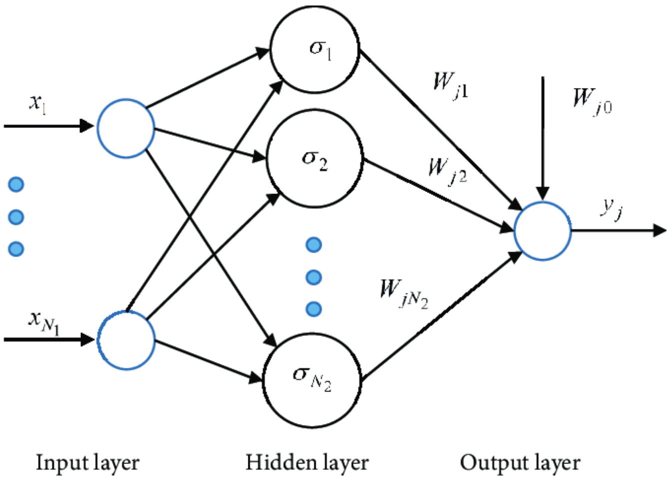

Fig. 1. Diagram of a Radial Basis Function Neural Network.

$$||x^{(i)}(t)|| \leq \frac{2^i \Phi}{\lambda^{n-i-1}}, \; t \geq T_0, \; i = 0, 1, \ldots, n-1. \tag{4}$$

This function is intended to show a bound on the synchronization errors between the agents and its neighbours, and ultimately on the synchronization errors of the follower agents with the leader agent by means of a constant parameter that is assigned. Due to the structure that is assumed in (1), by bounding the synchronization error for all the states with (4) we can achieve consensus. This is proven in [1]

III. Generalized Control Laws

It is necessary to implement a control law that bounds the error and limits that bound arbitrarily. Based on *Lemma* 1 the proposed distributed control protocol for the follower agents is as follows:

$$\begin{aligned} u_i &= cr_i - \hat{f}_i(x_i), \\ &= cr_i - \hat{W}_i^T \phi_i(x_i), i = 1, 2, \ldots, N. \end{aligned} \tag{5}$$

where, based on (3)

$$\begin{aligned} r_i &= \left(1 + \frac{d}{dt}\right)^{n-1} e_{i1} \\ &= C_{n-1}^0 e_{i1} + C_{n-1}^1 e_{i2} + \ldots + C_{n-1}^{n-1} e_{in} \end{aligned} \tag{6}$$

and,

$$\mathbf{e} = -\mathbf{L}\ \mathbf{x} - \mathbf{B}\ (\mathbf{x}_0\ -\ \mathbf{x}) \tag{7}$$

Being $\mathbf{e} = [e_{i1}, e_{i2}, \ldots, e_{in}]$ the state error between follower agents and state error between leader and followers with access to its information and $\mathbf{L}$ is the Laplacian Matrix of the communication topology of the agents. Coefficients C are the binomial expansion that depends on the number of states of the agents. $\hat{W}_i^T \phi_i(x_i)$ is the approximation of the nonlinearities of the follower agents. $\hat{W}_i^T$ and $\phi_i(x_i)$ are the weights and the activation function of the neural network, respectively.

The updating law of the weights are as follows:

$$\dot{\hat{W}}_i^T = \begin{cases} -\chi_i \phi_i r_i + \chi_i \frac{r_i \hat{W}_i^T \phi_i}{Tr[\hat{W}_i^T \hat{W}_i]} \hat{W}_i, & \text{if } -r_i \hat{W}_i^T \phi_i > 0 \text{ and} \\ & Tr[\hat{W}_i^T \hat{W}_i] = W_{max\ i} \\ -\chi_i \phi_i r_i, & \text{otherwise} \end{cases} \tag{8}$$

Where $W_{max\ i}$ is a constant that limits the value of the neural network weights, andχ_i is the adaptation rate. Given a strongly connected directed graph with N followers this distributed adaptive control protocol shall achieve synchronization of all agents to the leader.

IV. Simulation and Results

The simulation test case was taken from [1] and the dynamics for the agents were chosen as follows:

$$\begin{cases} \dot{x}_{01} = x_{02} \\ \dot{x}_{02} = x_{03} \\ \dot{x}_{in} = -3x_{01} - 3\ x_{02} - 4\ x_{03} + 11\ cos(3t) + 15\ sin(3t) \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = x_{13} \\ \dot{x}_{in} = -2x_{13} - 3\ sin(x_{12}) + 4\ cos(x_{11}) + u_1 + w_1 \\ \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = x_{23} \\ \dot{x}_{in} = -2x_{23} - 3\ sin(x_{22}) + 4\ cos(x_{21}) + u_2 + w_2 \\ \dot{x}_{31} = x_{32} \\ \dot{x}_{32} = x_{33} \\ \dot{x}_{in} = -2x_{33} - 3\ sin(x_{32}) + 4\ cos(x_{31}) + u_3 + w_3 \\ \dot{x}_{41} = x_{42} \\ \dot{x}_{42} = x_{43} \\ \dot{x}_{in} = -2x_{43} - 3\ sin(x_{42}) + 4\ cos(x_{41}) + u_4 + w_4 \\ \dot{x}_{51} = x_{52} \\ \dot{x}_{52} = x_{53} \\ \dot{x}_{in} = -2x_{53} - 3\ sin(x_{52}) + 4\ cos(x_{51}) + u_5 + w_5 \end{cases} \tag{9}$$

The disturbance are $w_i = 2\ sin(t + i\pi/8)$. Two network topologies were used for this simulation.

A. Test Results of Simulation with Topology 1

Topology 1 is depicted in Fig 2. With its Laplacian matrix as follows:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & -1 \\ -1 & -1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \tag{10}$$

And communication between Leader and followers is:

$$B = diag[0\ 0\ 0\ 1\ 0] \tag{11}$$

With matrices (10) and (11) we have a representation of the communication between agents, encoding the adjacency

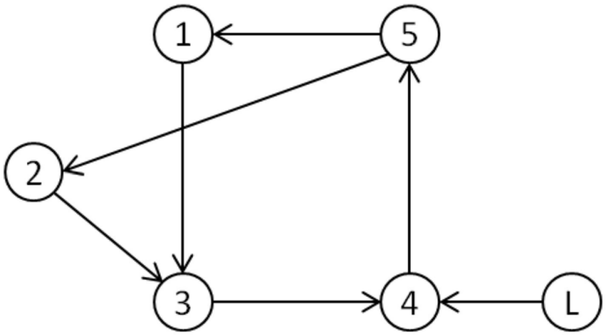

Fig. 2. Topology 1 Strongly Connected Directed Graph.

matrix to find the state errors between follower agents and the leader agent.

As in [1] an RBFNN of 27 neurons per agent was implemented with $\chi_i = 1$ and $W_{max\ i}$ =20 000. All the weights of the neural networks will be initialized to 0 and the activation functions have the following form:

$$\phi_{ij} = 30\epsilon^{-\frac{||x_i\ c_j||^2}{1000}} \tag{12}$$

with i being the number of agents and j, and c_j correspond to the centers of the neural network activation function. These centers are evenly distributed in a cube of [-1, 1] x [-1, 1] x [-1, 1]. For this test the initial conditions of the states are $x_1^{(0)} = [15\ 1\ 10]^T$, $x_2^{(0)} = [20\ 20\ 2]^T$, $x_3^{(0)} = [3\ 12.5\ 1.5]^T$, $x_4^{(0)} = [4\ 32\ 3]^T$, $x_5^{(0)} = [-5\ 40\ 4]^T$, and $x_0^{(0)} = [10\ 20\ 8]^T$. The control gain c is chosen as 60 from [1]. Initially, a simulation is performed without the RBFNN activated to see the behaviour of the synchronization control without the adaptiveness of the neural network. In Figures 3 to 5 we can observe the run time of 20 seconds and the systems has not converged, the error states with respect to the leader are oscillating.

When the RBFNN is not actively compensating the nonlinearities of the follower agents even though the error is bounded it is not equal to 0. As a final objective it is desired for the relative errors to the leader converge to zero to ensure there is a synchronization between follower agents and the leader. The next step in our simulation was to activate the RBFNN and see the results of compensation of the Neural Network. Figures 6 to 14 depict the behaviour of the agents with their nonlinearities compensated, it can be observed the error is smaller and the convergence is faster.

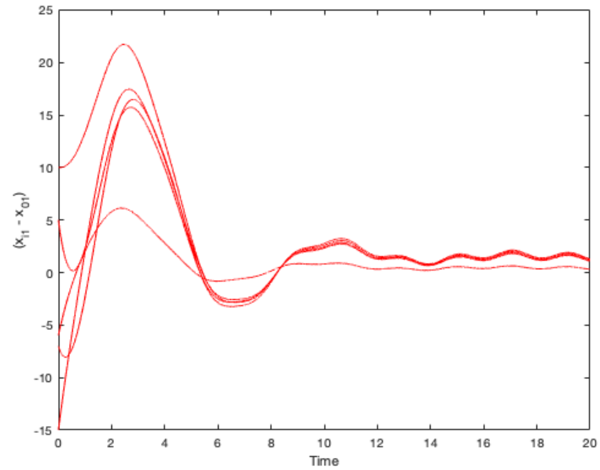

Fig. 3. State Error to leader $x_{i1} - x_{01}$

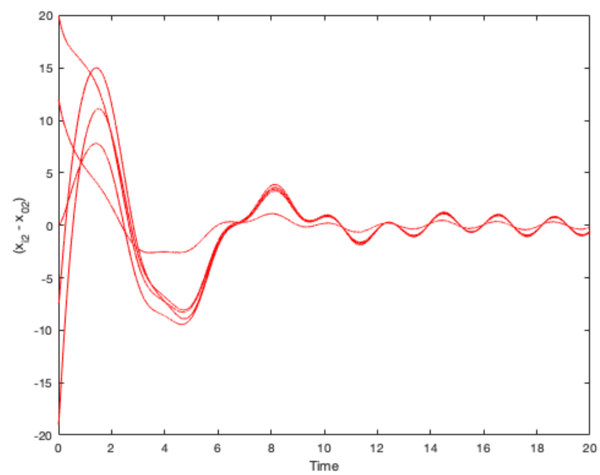

Fig. 4. State Error to leader $x_{i2} - x_{02}$

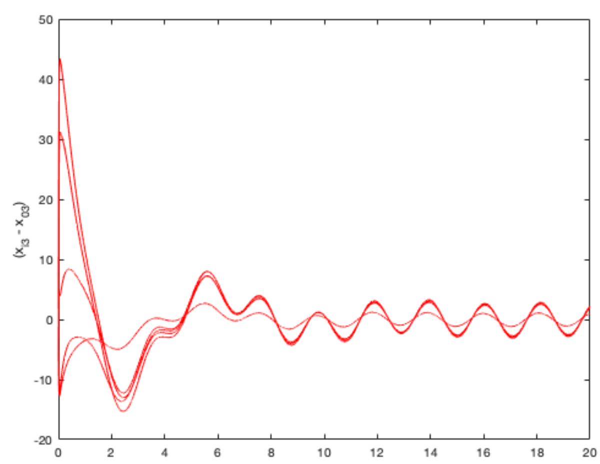

Fig. 5. State Error to leader $x_{i3} - x_{03}$

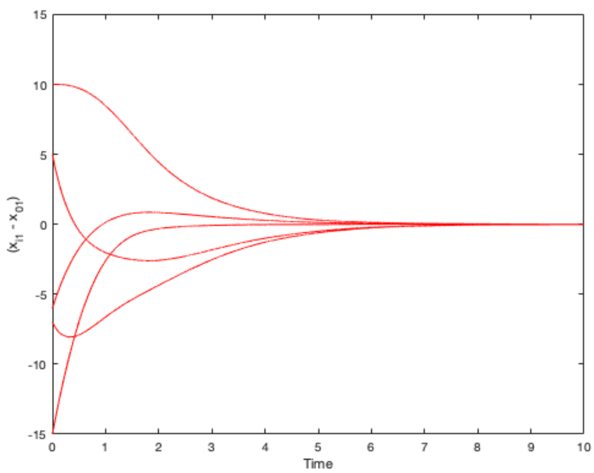

Fig. 6. State Error to leader $x_{i1} - x_{01}$

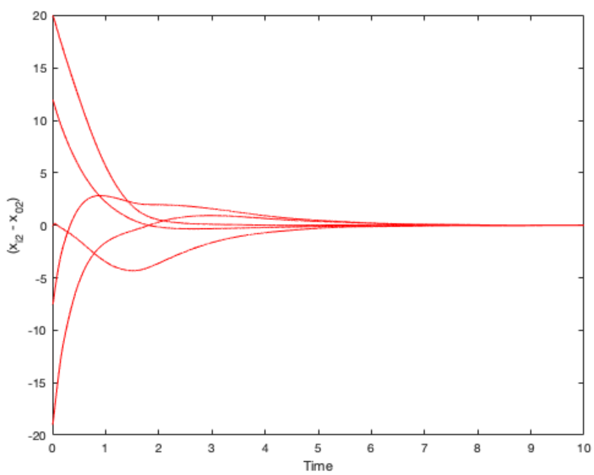

Fig. 7. State Error to leader $x_{i2} - x_{02}$

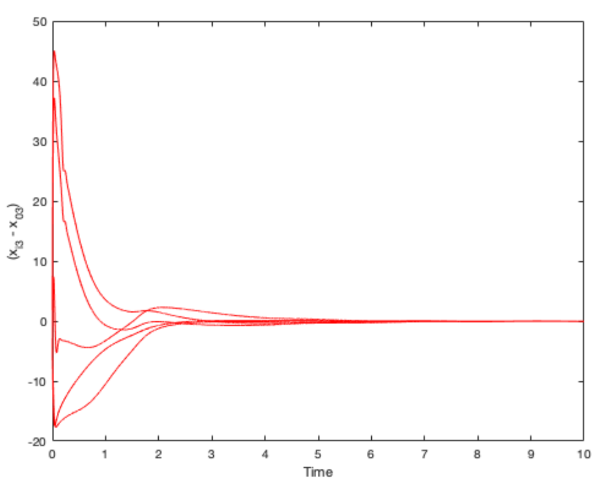

Fig. 8. State Error to leader $x_{i3} - x_{03}$

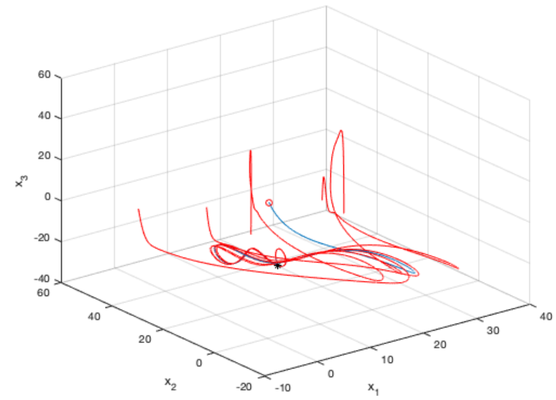

Fig. 9. Example of a figure caption.

B. Test Results of Simulation with Topology 2

The second simulation in this project changes the topology of communication between the followers and Leader as depicted in Figure 10, taken from the base work of [1]. The Laplacian Matrix of this topology :

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad (13)$$

And communication between Leader and followers is:

$$B = diag[0\ 0\ 1\ 0\ 0] \quad (14)$$

Once we have obtained the topology of the communication network, we can implement the proposed control protocol for the agents. The initial states for the agents are: $x_1^{(0)} = [-25\ 1\ -15]^T$, $x_2^{(0)} = [20\ -20\ -4]^T$, $x_3^{(0)} = [3\ -14\ -3.5]^T$, $x_4^{(0)} = [9\ -30\ -5.5]^T$, $x_5^{(0)} = [5\ 45\ 12]^T$, and

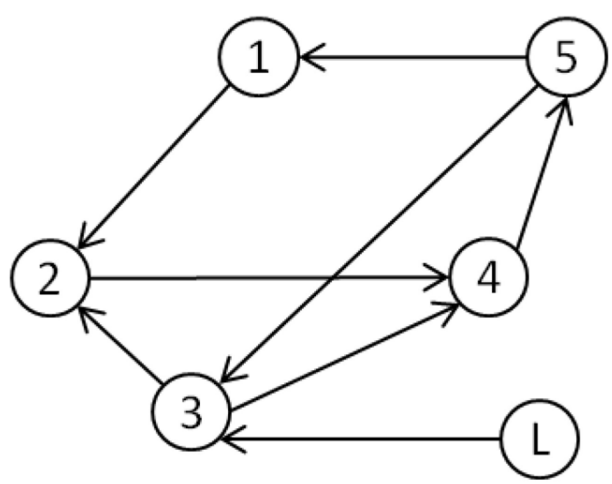

Fig. 10. Topology 2 Strongly Connected digraph

$x_0^{(0)} = [10\ 20\ 8]^T$. In this simulation the control gain will remain the same as in the previous test.
The objective of this second test is to achieve a synchronization with different initial conditions for the follower agents as well as for the Leader within a different communication network topology. This was done to show the adaptiveness of the control protocols and how an agreement is still possible for a wide range of cases. Figures. 11 to 13 show the state error of each agent with the leader and it can be observed that the error is bounded and all the agents reach consensus.

V. Conclusion

The simulation results in [1] have been successfully replicated and it is important to notice that the simulation of these dynamics requires a very small sampling time, in this case a 0.001 step per frame. When creating an application for agents that require real-time nonlinear compensation of the RBFNN, one may encounter problems since it is not practical to have a sampling time for the resolution required. Nevertheless, the RBFNN is an important tool to add adaptivity to the control protocols where the dynamics of some agents are not know but need to be compensated.

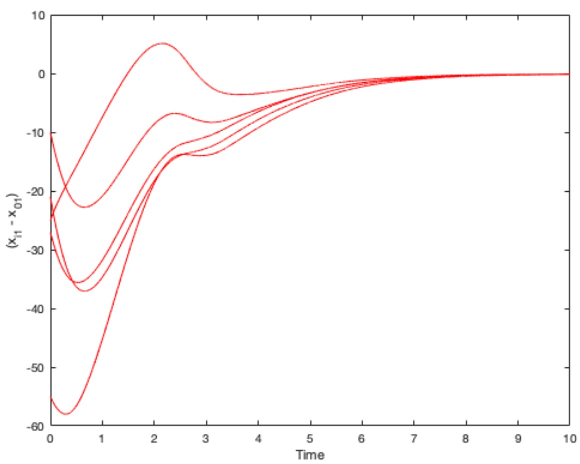

Fig. 11. State Error to leader $x_{i1} - x_{01}$

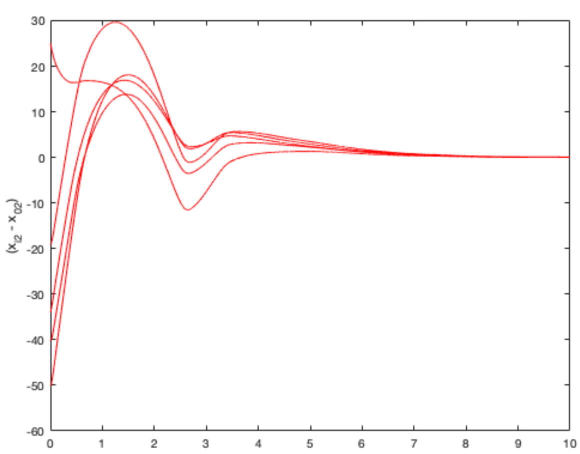

Fig. 12. State Error to leader $x_{i2} - x_{02}$

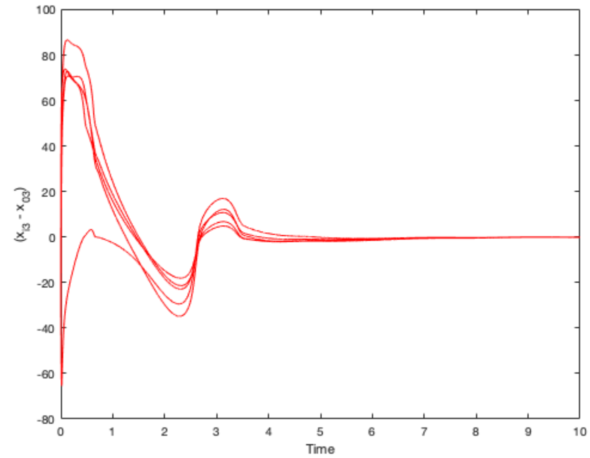

Fig. 13. State Error to leader $x_{i3} - x_{03}$

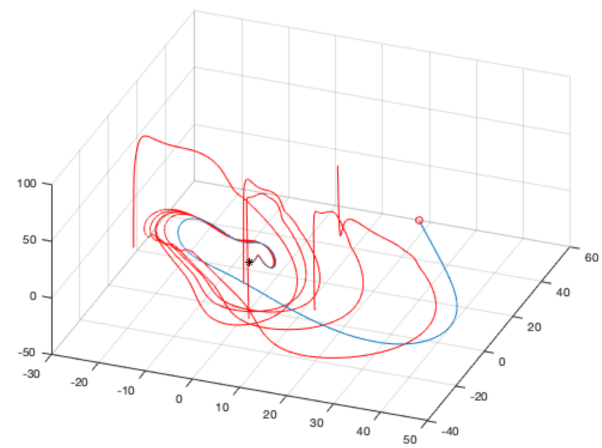

Fig. 14. State Error to leader $x_{i3} - x_{03}$

With the two test cases simulated we noticed that the system, at least for these two cases, can reach synchronization when presented with a different topology and different initial conditions. It would have been interesting to test with higher order dynamics and a greater number of agents the adaptability of these control protocols, but due to time constraints it was not able to be completed. This may be for future work.

References

[1] S. Su, Z. Lin, and A. Garcia, "Distributed synchronization control of multiagent systems with unknown nonlinearities," IEEE Trans. Cybern., vol. 46, no. 1, pp. 325-338, 2016.

[2] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," IEEE Trans. Ind. Informatics, vol. 9, no. 1, pp. 427-438, 2013.