International Journal of Control, Automation and Systems 15(5) (2017) 2320-2328
http://dx.doi.org/10.1007/s12555-016-0440-9

ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Analysis of the Effects of Communication Delays for Consensus of Networked Multi-agent Systems

Myrielle Allen-Prince, Christopher Thomas, and Sun Yi*

Abstract: Achieving cooperation and coordination in a network of multi-agent systems is key to solving the consensus problem. Synchronization of such systems requires consistent communication between agents to reach a consensus, which is not feasible in the presence of delays, data loss, disturbances, and other unpredictable factors. Communication delays combined with environmental uncertainties can cause adverse effects and negatively change the behavior of the networked system preventing synchronization. In this paper, the stability of the delayed networked systems is analyzed using delay differential equations. Solving these equations has not been feasible, because of the infinite number of characteristic roots. The approach based on the Lambert W function has the capability of analytically solving delay differential equations. The approach is used to quantify and analyze the stability of the delayed networked systems. The communication between the agents is modeled using the graph theory and the Laplacian matrix. The stability is analyzed by incorporating the Laplacian matrix into the Lambert W function based approach which provides the locus of the eigenvalues of the system as delay changes. Sensitivities and convergence speed with respect to delay for various topologies of the network are presented for comparison. The numerical results and implementation using MATLAB/Simulink are presented for illustration.

Keywords: Consensus, delay, multiple-agent systems, sensitivity, stability, topology.

1. INTRODUCTION

Coordinated and cooperative control has been studied in [1,2] to develop control algorithms for multi-agent systems (MAS). Compared to a single-agent system, MAS can perform multiple complex tasks in less time and encountering less setbacks or task failures [3,4]. MAS can improve security, search and rescues, environmental monitoring, and much more. The consensus problem consists of making the individual agent within a network come to an agreement on a decision and/or task. This gives these multi-agents the ability to perform cooperatively in a coordinated manner. Some applications of the consensus problem are formation flight, traffic monitoring and control, swarming/flocking, satellite formation, and many more systems that can communicate and contain cooperative control capabilities. There is a great potential but there are also many challenges that must be dealt with and overcome. Networked systems are prone to failures due to uncertainties such as disturbances, nonlinearity, information signal loss, and communication delay. This research focuses on analyzing the stability, convergence speed, and sensitivity of networks of multi-agent systems in the presence of delay.

Delays in the network can lead to instability and unpredictable behaviors of the system and its agents. These agents are expected to perform designated tasks requiring them to communicate. This communication is done wirelessly, which is not always reliable. Wireless communication has major challenges within itself such as signal fading, interference, attenuation, intermittent connectivity, or link breakage. All of which can lead to delayed communication between agents. The effects of delay are neither trivial nor predictable [5]. One stipulation of the consensus problem is synchronization between agents that can be greatly affected by the delay. Time delay and communication topology are the key factors that influence the stability of the multi-agent system [2]. To ensure the agents reach a consensus, we need to know when and how delay will affect it. Analysis of the delay's effects is challenging. Delay in the characteristic equation can be represented by exponential functions, which lead to an infinite spectrum. Thus, when modeling a delayed system using delay differential equations (DDEs) the delay term introduces an

Manuscript received July 22, 2016; revised December 19, 2016; accepted January 15, 2017. Recommended by Associate Editor Hyo-Sung Ahn under the direction of Editor Duk-Sun Shim. This material is based on research sponsored by Air Force Research Laboratory and OSD under agreement number FA8750-15-2-0116. Also, the research has been supported by DOE. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.

Myrielle Allen-Prince, Christopher Thomas, and Sun Yi are with Department of Mechanical Engineering in North Carolina A and T State University, 1601 East Market Street, Greensboro, North Carolina 27411, USA (e-mails: myrielle.allenprince@gmail.com, cothomas@aggies.ncat.edu, syi@ncat.edu).
* Corresponding author.

infinite number of characteristic roots and, thus, solving these equations has not been feasible. In this paper, the stability of the delayed networked systems is analyzed using an approach based on the Lambert W function, which has the capability of analytically solving delay differential equations. The approach is used to quantitatively analyze the stability of the delayed networked systems. The communication between the agents is modeled using the graph theory and the Laplacian matrix. The stability is analyzed by incorporating the Laplacian matrix into the Lambert W function based approach. That enables one to obtain the locus of the eigenvalues of the system as delay changes. Also, sensitivities and convergence speed with respect to delay for various topologies of the network are presented for comparison. The numerical results and application of a testbed using MATLAB/Simulink are presented for illustration.

This is organized as follows: Section 2 contains background information of this research on the consensus problem of multi-agent systems The focus, challenges to overcome, and the methods used to address these challenges are described. Section 3 provides the methodology used to model these systems and analyze their behaviors. A new method is used to analyze the systems stability and sensitivity to ensure its synchronization ability. Section 4 contains validation of the methods used in Section 3. Section 5 provides a summary of the results and recommendations for future work.

2. CONSENSUS AND DELAY

2.1. Consensus

The consensus problem has been integrated and applied in many fields of study for networks of dynamic systems. The increased attraction to the distributed coordination of these systems is partially due to the broad applications of MAS such as cooperative control of unmanned aerial vehicles (UAVs), formation control, flocking, distributed sensor networks, satellite clustering, and congestion control in communication networks [6]. In all these applications, MAS are groups of agents that need to come to an agreement on a task. There are multiple studies on the consensus problem [1, 2], most of which focused on switching topology and time delay [2, 6] and cooperative and coordinated control [1, 7–9].

2.2. Time delay

Time delay is present typically in systems that are required to communicate over a network or channel [10]. This is due to many factors, a few being the signal transmission speed and network congestion [11, 12]. Ignoring time delay is impossible, because it is always present and can cause instability in the system [11]. Delay in the network can lead to unpredictable behaviors of the system and its agents. The delay provides the model with an infinite number of characteristic roots creating difficulty in determining the stability or designing controllers.

Lyapunov and linear matrix inequality (a.k.a. LMI) methods, robust controller synthesis has been used to obtain a maximal allowable upper bounds for known and unknown delays [11, 13, 14]. Many types of Lyapunov functions have been used to handle delays [15, 16]. The latter has conservative results, which applies to time-varying delays without restriction but boundedness and Krasovskii-Lyapunov is a classical technique and requires a bounded derivative [5]. Refer to [5] for an article on multiple time delay techniques and open problems. The Lambert W Function-based approach provides solutions to a type of DDEs and the stability by only using its principle branch [17].

2.3. Lambert W function-based approach

The Lambert W Function is defined as

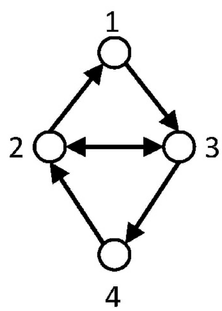
$$z = W_k(z)e^{W_k(z)}, \tag{1}$$

where z is any complex number, $W(z)$ is the Lambert W function. Also, $k = -\infty, \cdots, +\infty$ represent the branches of the W function. These branches correspond to the infinite number of solutions the Lambert W function can provide. The principal branch is $W_0(z)$ is the only required branch to determine stability in the case of infinite roots [17–19]. This function can handle the exponential term in the characteristic equation of DDEs. For detailed information on the Lambert W function, refer to [18]. MATLAB has a Lambert W function embedded, which was used for calculations within this research.

2.4. Graph theory and Laplacian matrix

Graph theory is used in mathematics and computer science, where graphs are used to show the connection or relationship between a specific group of objects. These graphs typically consist of circles, lines and arrows. The circles are called nodes or vertices, which are the objects (agents) within the group. The lines or arrows are called edges, which indicate the connected/communicating nodes. Graphs can be categorized as undirected, which are represented by lines or directed, which are represented arrows. This research uses directed graphs to indicate the information being sent or received, or the direction of communication. A directed graph also known as a topology is defined as $G = (V, E)$, where V is a set of nodes or vertices $\{v_1, v_2, ..., v_n\}$, and E is a set of edges. An edge can be defined as $e_{ij} = (v_i, v_j)$, where v_i is the node sending and v_j is receiving. Some sources define a graph as $G = (V, E, A)$, where A is the adjacency matrix $[a_{ij}]$ which, likes edges, shows which nodes are connected.

A Laplacian Matrix is developed from a graph (topology). These matrices consist of mostly 0 and -1 on the off-diagonal. The diagonal will depend on the number



$$L_{ij} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

Fig. 1. Topology and Laplacian matrix of the System A [6]: each agent must send or receive information.

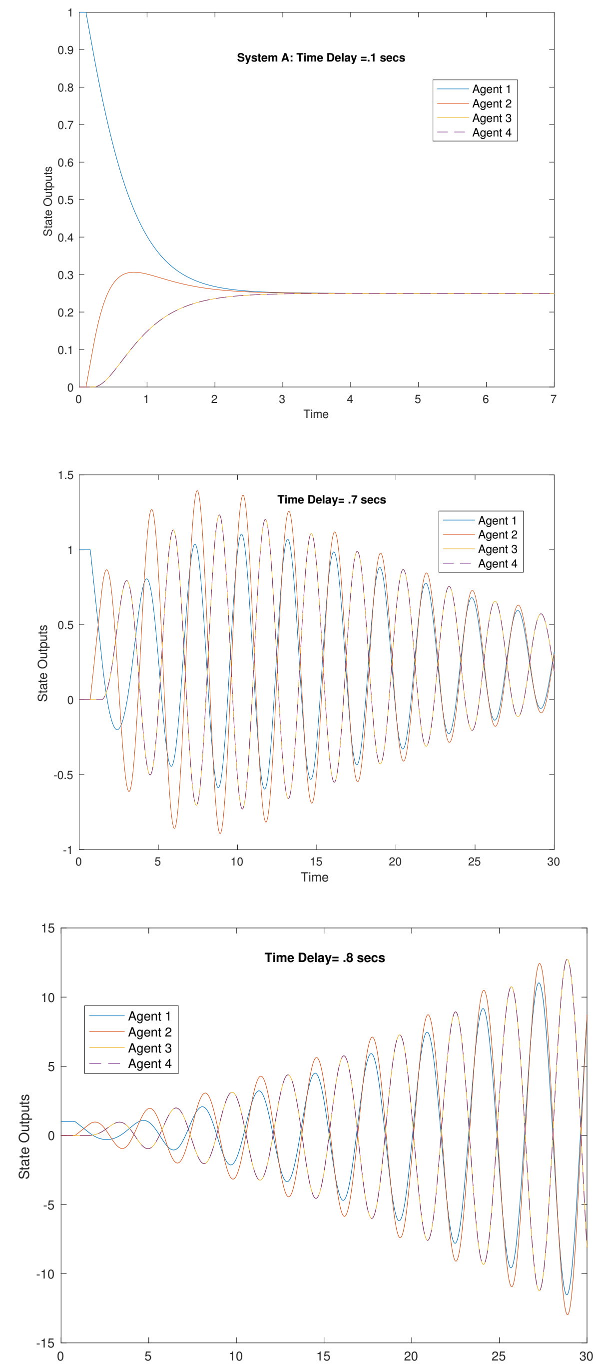


Fig. 2. Simulations of effects from time delay of .1, .7, and .8 secs on System A.

of sources and targets. The Laplacian matrix is developed from the difference in the degree matrix and the adjacency matrix ($L_{ij} = D_{ij} - A_{ij}$). These matrices provide system information necessary for modeling and simulations. Graph Laplacians are used for a group of agents given the task of formation stability [6]. The eigenvalues from the Laplacian matrices determines the stability of the system and assists in sensitivity analysis. For background information on graph theory, Laplacian matrices, and matrix theory and their application to the consensus problem refer to [2,6,7,9].

3. EIGENVALUES AND ANALYSIS

This section presents the modeling and analysis of MAS with time delay. The system is modeled, and MATLAB and Simulink are used to perform the analysis. Throughout this research, we focused on systems of four agents. These four agents send and/or receive information from one or more agents within the system based on their topology. We determine which topology is more effective for the consensus problem by modeling the systems and analyzing the delay effects on the stability and other characteristics of each system.

3.1. Modeling

Graph theory is used to rep resent the communication between agents. Each node is an agent and the arrows are the sent or received information, as seen in Fig. 1 along with its Laplacian matrix. Fig. 1 shows a multi-agent system made up of four agents. Since each agent must send or receive information, the performance of each agent has an effect on the one agent receiving the information. Due to signal transmission between agents, time delay is present in such system but the magnitude and its affects varies from system to system, and is dependent on the environment and communication types. To understand how time delay can affect a system's performance a Simulink model was built using the matrix from Fig. 1. From the simulations, the system's response and behavior are analyzed as the time delay increases. Fig. 2 shows the systems response with delay of .1, .7, and .8 seconds. As the delay increases, the system begins to oscillate. The critical point for this system is between .7 and .8 seconds seen in the last two plots in Fig. 2. After .8 seconds, the system remains unstable as the delay increases. The system's eigenvalues,

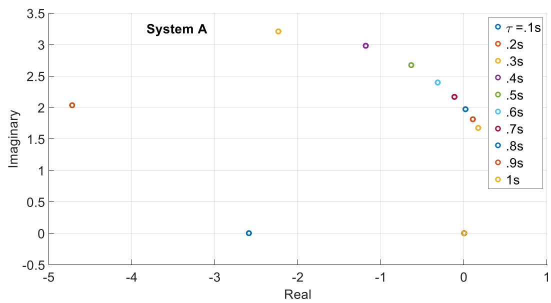


Fig. 3. Characteristic roots for the system with time delays .1 secs to 1 sec. If delay is longer than .7 seconds, the system becomes unstable.

also known as characteristic roots or poles, allow the system to be analyzed further providing important information on the states of the system. To obtain the eigenvalues of a system one should first have its mathematical representation.

Delay differential equations are used to mathematically represent the system, where A is the system matrix, A_d is the delay matrix, and the time delay. For the above case, $A = 0$ and $A_d = -L$.

$$\dot{x} = -Ax - A_d x(t-\tau). \tag{2}$$

The difficulty in analyzing DDEs lies in the transcendental characteristic equation, which leads to an infinite number of complex roots [17]. This is due to the delay operator $e^{-\tau S}$, which has an infinite number of roots [19]. The transcendental characteristic equation of (2) is represented by (3). The S matrix in (3) provides the eigenvalues of the DDE.

$$S + A + A_d e^{-\tau S} = 0. \tag{3}$$

Solving for S leads to [17]

$$S = \frac{1}{\tau} W(-A_d \tau e^{A\tau}) - A. \tag{4}$$

Equation (4) is the standard form when the coefficients, A and A_d, commute [19]. Because the matrix S represents the four agents in the system, the system must be decoupled to allow the freedom of changing one system without affecting another. To decouple the system, we begin by taking the Jordan form of $-A_d \tau e^{A} \tau$, which is denoted as $J(D)$. Taking the Jordan form handles the effects of repeated eigenvalues and does not affect the stability for systems with discrete eigenvalues. Next, apply the Lambert W function to the Jordan matrix, $W(J)$. For any off-diagonal values changed when taking $J(D)$, the Lambert W function cannot be applied directly and the derivative of the Lambert W function is applied to that value instead. Finally, a new matrix, $W(H)$, is found using eigenvalue

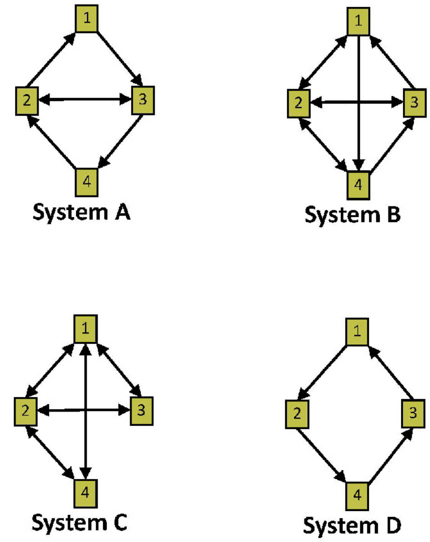


Fig. 4. Communication topologies of four multiple-agent systems.

decomposition as

$$W(H) = V * W(J) * V^{-1}, \tag{5}$$

where

$$S = \frac{1}{\tau} W(H) - A, \tag{6}$$

where V is the eigenvector matrix of $J(D)$. The matrix S can now be found by replacing (4) with (6). The eigenvalues of S are used to quantify and analyze the stability. These values can be seen in the complex plane (s-plane) in Fig. 2 where $A = 0$ and $A_d = -L$. Refer to [17–19] for more information on the application of the Lambert W function on systems of DDE's.

Fig. 3 shows that the system is stable for time delays less than .8 seconds with a critical point a .7 seconds of delay. Note that all of the systems has one eigenvalue at zero. Comparing the mathematical solution in Fig. 3 to the simulation results in Fig. 2, the distance from the imaginary axis is proportional to the convergence speed of oscillations. Thus, this mathematical approach provided an accurate way to analyze the time delay effects on this system by quantifying the stability. Stability of delayed networked systems have been studied using a bifurcation analysis, which replacing s by a purely imaginary number, and Nyquist plots [1, 6]. The bifurcation method determines whether a system is stable or not without information about how stable a system is. Nyquist plots provide stability margins but for a different system one should draw a new plot. Compared to the methods, the method using the Lambert W function provides the analytical form in terms of the parameters as shown in

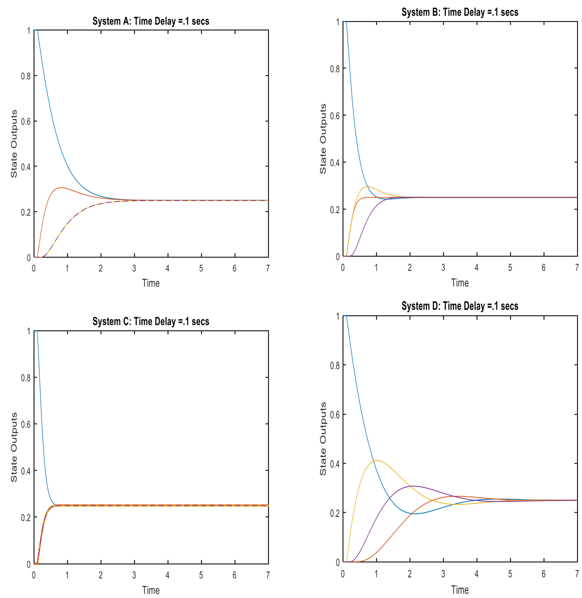


Fig. 5. Simulations for all systems A-D for .1 secs delay: Systems B and C converge faster and, thus, are more stable.

(4). The form shows how each parameter including delay in models affects stability. Also, the distance from the imaginary axis shows how stable the system is (see Fig. 3 and more discussions in the subsequent subsection). For general comparison of existing methods for stability with examples, refer to [19].

3.2. Extending to more topologies

Three more topologies are modeled and analyzed using the same approach shown in Section 3.1. Different topologies are compared because it is mentioned in [1] that the eigenvalues of the Laplacian matrix are directly affected when changing the topology of a network. Fig. 4 shows the previous topology (A) and the additional topologies (B, C, and D). These topologies will be referred to as System A, B, C, and D, respectively, throughout the rest of this paper.

The system's convergence speeds are first analyzed by their simulated responses shown by Fig. 5 with .1 second delay. Systems B and C appear to have one or more agents that converge faster than those in A and D and are therefore more stable systems. By obtaining the eigenvalues, we can observe how much faster Systems B and C are compared to Systems A and D. The convergence speed of a system is determined by the systems time constant, which is the inverse of the absolute real part of the eigenvalue, $1/|-\lambda_i|$. The farther left the eigenvalue, the faster the system converges. Each system's eigenvalues were plotted on the complex plane (s-plane) in Fig. 6. Analyzing their speeds for $\tau = .1$ seconds, Systems B and C both has an eigenvalue with a magnitude of approximately

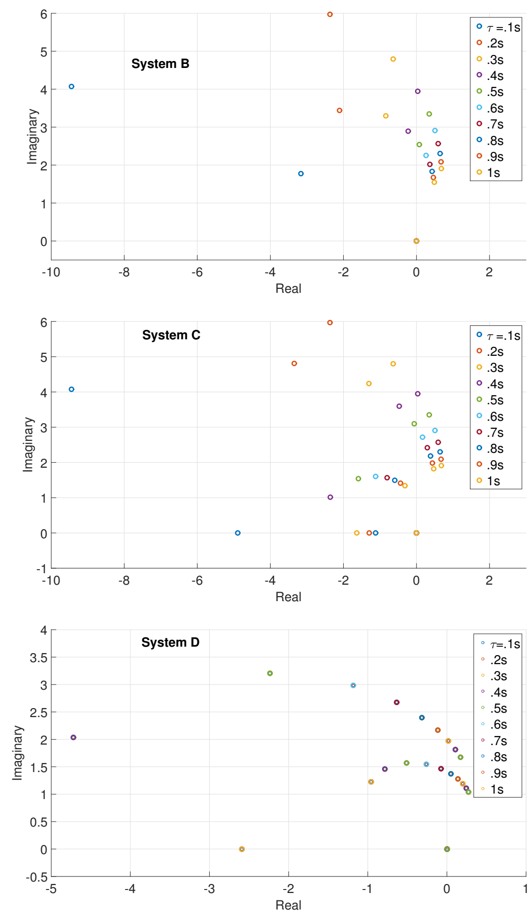


Fig. 6. Eigenvalues in the complex plane for each system.

9.8. Systems A and D eigenvalues has a magnitude of approximately 2.6 at $\tau = .1$ seconds. Therefore, Systems A and D are 3-4 times slower than B and C. To see if their speed has any relation to their stability, the critical points are examined. System A was previously analyzed and has a critical point at $\tau = .7$ seconds. System's B and C are both critically stable at $\tau = .3$ seconds and system D at $\tau = .5$ seconds. That is, System A is stable for the widest range of delays followed by system D making them more robust than systems B and C.

This shows there is an inverse relationship to the speed of the system and the system's robustness, agreeing with [1, 6] stating there is a tradeoff between performance and robustness of a protocol to time delays. There is also an assumption made from this relation between performance and robustness, that faster systems are more sensitive to change in information flow. A sensitivity analysis will be performed to see if that assumption holds.

3.3. Sensitivity analysis

The simulations show that Systems A and D show slower responses than B and C but are more tolerant to the change in time delay. This leads to the assumption that faster systems are more sensitive to changes in topology. A sensitivity analysis shows how the eigenvalues are affected. To do this we start by rewriting (2) as

$$\dot{x} = Lx(t-\tau). \tag{7}$$

Use the eigenvalue decomposition of L to get

$$\dot{x} = V\Lambda V^{-1}x(t-\tau), \tag{8}$$

where V is the eigenvector matrix of and is the eigenvalue matrix of L. Multiplying both sides by V^{-1}

$$V^{-1}\dot{x} = \Lambda V^{-1}x(t-\tau) \tag{9}$$

creating a new variable y, and setting it equal to $V^{-1}x$ we get an equation for the eigenvalues as

$$\dot{y} = \Lambda y(t-\tau). \tag{10}$$

Since Λ is diagonalizable the system can be decoupled, changing (10) to (11) where the equation is represented by its components $i = 1,\ldots,4$.

$$\dot{y}_i = \lambda_i y_i(t-\tau). \tag{11}$$

The same steps from (2) to (6) were applied to (11) as

$$s = \frac{1}{\tau}W(\lambda_i\tau). \tag{12}$$

The derivative of the solution in (12) with respect to τ can be written as

$$\frac{ds}{d\tau} = -\frac{W(z)}{\tau^2} + \frac{\lambda_i}{\tau[e^W(z)+z]}, \tag{13}$$

where $z = \lambda_i\tau$. The derivative of Lambert W function given in [17, 18]. The eigenvalues sensitivity parameters are obtained from (13) and are listed below in Table 1. Comparison of these results help determine how sensitive a matrix/system is to perturbations. This is necessary because the change in the topologies is affecting the eigenvalues. In Table 1, the sign and magnitude are used to analyze the sensitivities. If the sign is negative, then change in information flow stabilized the system more shifting the eigenvalue further left of its original location. For positive values, the eigenvalue crossed the imaginary axis in the s-plane as the information flow changed. The higher the magnitude of the positive real component of the eigenvalue the more sensitive the system is to perturbations.

Comparing Systems A and D to Systems B and C, Systems B and C are more sensitive than A and D (larger values). For systems A and D, the highest change in the eigenvalues was approximately 40 for λ_{2-4} in A and λ_2 in D for $\tau = .2$ seconds. Systems B and C has a higher magnitude of approximately 161 for $\tau = .1$ seconds for λ_4. Systems A and D have been proven to be able to withstand the changes of time delay much longer than systems B and C. The sensitivity analysis confirms the assumption about the direct relationship between speed and sensitivity.

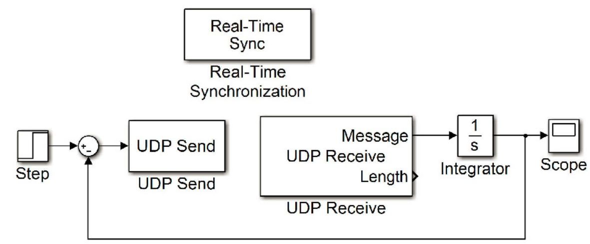


Fig. 7. Computer 1 Model for simulations in real time to realize the systems with Internet delays.

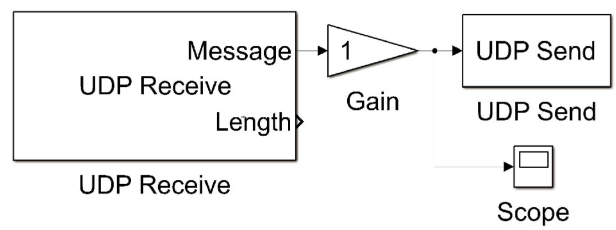


Fig. 8. Computer 2 Model for simulations in real time to realize the systems with Internet delays.

4. EXPERIMENTS THROUGH INTERNET

This section presents the results of the MATLAB/Simulink simulations run in real time to realize the systems in Fig. 4 with Internet delays. Computers can communicate in two ways, either using Transmission Control Protocol over Internet Protocol (TCP/IP) or User Datagram Protocol (UDP). UDP is used here since this research requires fast communication while running in real-time. Testing the communication between two computers required using Simulink's UDP Send and Receive blocks. Below are the models for communication from computer one to computer two then back to computer one.

The Real-Time Synchronization block captures the communication between the computers over the number of seconds for which the systems are communicating; in synchronization with the computer's clock. This set-up allows for the testing of the control gain until the model became unstable. To incorporate this into our research the models above had to be modified to reflect their mathematical models. For simplicity, two computers are used for this experiment and each computer represents an agent in a network. All but the gain block from Fig. 8 are used and the zero-order hold block was added to ensure the discrete information received is sent as a continuous signal. The transport delay is artificial delay, added to the system to allow the comparison of the previous states in this agent to the delayed states received from the other agent(s). The delay between two computers was approximated as .1 seconds and is used as the artificial delay time. The algorithm for modeling a delayed networked system was found in [6]

Table 1. Sensitivity Analysis or Systems A, B, C, and D using the result in (13).

System A							
Delay	.1	.2	.3	.4	.5	.6	.7 Critical pt.
λ_1	0.0000	$0.0000+0.0000i$	$0.0000+0.0000i$	$0.0000+0.0000i$	$0.0000+0.0000i$	$-0.0000+0.0000i$	$-0.0000+0.0000i$
λ_2	−9.0668	$40.3312+50.0675i$	$15.0194-0.3624i$	$7.2709-3.0772i$	$4.0620-2.9750i$	$2.4821-2.5317i$	$1.6116-2.1114i$
λ_3	−9.0668	$40.3312+50.0675i$	$15.0194-0.3624i$	$7.2709-3.0772i$	$4.0620-2.9750i$	$2.4821-2.5317i$	$1.6116-2.1114i$
λ_4	−9.0668	$40.3312+50.067i$	$15.0194-0.3624i$	$7.2709-3.0772i$	$4.0620-2.9750i$	$2.4821-2.5317i$	$1.6116-2.1114i$
System B				System C			
Delay	.1	.2	.3 Critical pt.	Delay	.1	.2	.3 Critical pt.
λ_1	$0.0000+0.0000i$	$-0.0000+0.0000i$	$0.0000+0.0000i$	λ_1	$0.0000+0.0000i$	$0.0000+0.0000i$	$0.0000+0.0000i$
λ_2	$-5.42+17.88i$	$17.6204+4.0936i$	$8.4906-3.8467i$	λ_2	$-1.41+0.0000i$	$-2.2667+0.0000i$	$-5.2121+0.0000i$
λ_3	$-5.42-17.88i$	$17.6204-4.0936i$	$8.4906+3.8467i$	λ_3	$-46.91+0.0000i$	$33.7937-0.8153i$	$12.0611-7.0302i$
λ_4	$161.32+200.27i$	$29.0836-12.3087i$	$9.9286-10.1267i$	λ_4	$161.32+200.27i$	$29.0836-12.3087i$	$9.9286-10.1267i$

System D					
Delay	.1	.2	.3	.4	.5 Critical pt.
λ_1	$0.0000+0.0000i$	$0.0000+0.0000i$	$0.0000+0.0000i$	$-0.0000+0.0000i$	$-0.0000+0.0000i$
λ_2	$-9.0668+0.0000i$	$40.3312+50.0675i$	$15.01940.3624i$	$7.27093.0772i$	$4.06202.9750i$
λ_3	$0.98512.4647i$	$2.43281.8704i$	$2.78480.3456i$	$2.1780+0.6068i$	$1.5230+0.9292i$
λ_4	$0.9851+2.4647i$	$2.4328+1.8704i$	$2.7848+0.3456i$	$2.17800.6068i$	$1.52300.9292i$

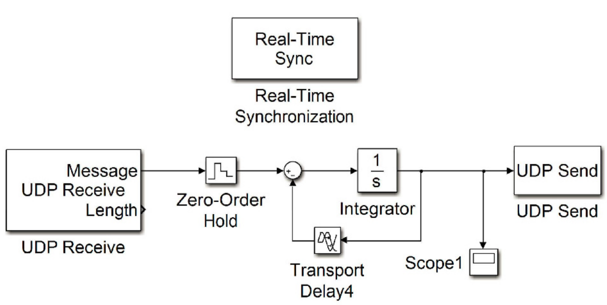


Fig. 9. Simulink for each agent to communicate with an additional transport delay.

as

$$\dot{x}_i = \sum_{v_j \in N_i} \alpha_{ij}[x_j(t-\tau_{ij}) - x_i(t)], \tag{14}$$

where v_j is an agent, N_i is a set of neighbors of v_i. This equation shows the consensus algorithm. The algorithm does not take into account the artificial delay presented in Fig. 9. The block diagram in Fig. 9 has the additional transport delay to the system changing (14) to

$$\dot{x}_i = \sum_{v_j \in N_i} \alpha_{ij}[x_j(t-\tau) - x_i(t-\tau)], \tag{15}$$

where τ is the delay in the received and sent information. This allows fair comparison of the delayed information in both systems. The Simulink blocks were set-up accordingly for each computer. The agents should reach a consensus if set up correctly. Agent 1 is given an initial condition to ensure information is being sent when the system starts. Agent 2 has no initial condition and is started before Agent 1. By doing this, we can see when Agent 2 begins to receive information from Agent 1; being the delay in communication. Fig. 10 shows the response of the network from the start of Agent 1. As hoped each computer came to an agreement on the final state of the system between .2 and .25 seconds. The delay in Agent 2 can be seen as well and is less than 1 second. Agent 2 starts receiving information approximately .5 seconds after Agent 1. When Agent 2 is shutdown, the communication terminates. Due to delay, Agent 1 is still receiving information even though Agent 2 has shutdown. Agent 1 continues to receive this delayed information until it reaches zero meaning there is no more information to collect. The two-agent real-time communicating system is extended to four-agent systems. System D is the simpliest topology because it is a cyclic graph and, therefore, is used for comparison with the numerical data. Fig. 12 is the system D's response, which includes information flow between four agents. From the topology of system D and as seen in Fig. 11, the agents are communicating as $1(host) \to 2 \to 4 \to 3 \to 1$ and the response is represented in the same manner. To catch the delay in the

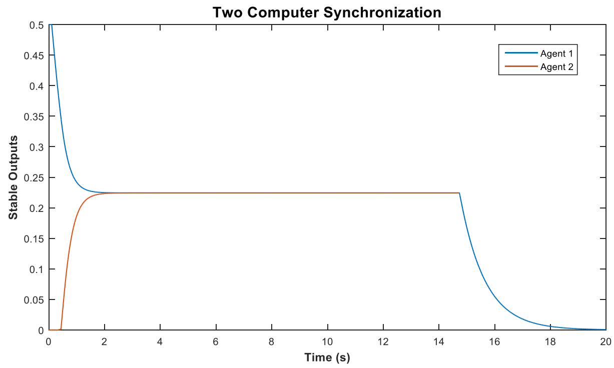


Fig. 10. Two computer communication synchronization in real-time.

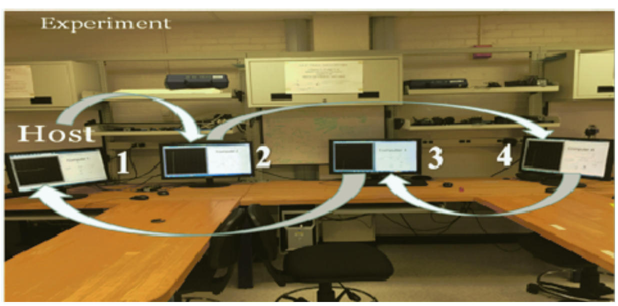


Fig. 11. Four Computers are used for real- time experiment considering delay effects.

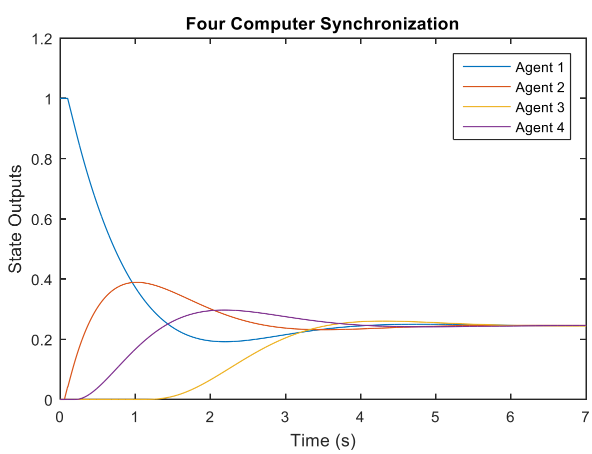


Fig. 12. System D communication synchronization in real-time.

communication, the individual agents start in the reverse order in which they are expected to operate. Agent 3 will begin first, then Agent 4, Agent 2, and lastly Agent 1. Like with the two Agents above, this method shows the delayed start for each Agent in the network. The delay between these agents are much more obvious than the system with two agents. By analyzing Fig. 12, the biggest delay is in Agent 3 at approximately 1.1 seconds, which is .88 seconds after Agent 4 began to receive information. The smallest delayed start was in Agent 2 at .1 seconds. The experimental responses for System's A-C were obtained but omitted due to limited space. Each system is stable and reaches a consensus like their simulated responses. Similar to the response of System D, there is obviously some delay between the communicating agents. Compared to System D, these systems have a faster response and less delay in their agent's communication. Systems B and C are the faster converging systems, which can also be seen in their simulated responses. The size of the delay seems to be dependent on the amount of agents communicating and the flow of information.

5. CONCLUSION

The consensus problem and the system's stability in the presence of communication delay was studied. The analytical solutions are derived in terms of the Lambert W function. Then, the stability of the systems is determined from the characteristic roots (or eigenvalues). Also, using the derivative of the function sensitivity with respect to changes in the delay are quantified and analyzed. By simulations alone, one could conclude Systems B and C consisted of agents with response speeds much faster than those in Systems A and D. Fast system responses are ideal but not practical in reality for most systems. The time delay from .1 seconds to 1 second caused each system to become unstable. Systems B and C became unstable for shorter time delays compared to Systems A and D. It was mentioned by Saber and Murray that there is a tradeoff between performance and robustness, also found in [1, 6]. Applying the Lambert W function to these simulated systems confirmed this approach is able determine the eigenvalues of the system. Although the Lambert W function-based approach was applied to 4 topologies, it can be applied to general topologies for quantitative study of stability and sensitivity. This is the unique and novel contribution of the presented work. Compared to the methods in [1, 6, 7], the presented approach make it possible to analyze and quantify the stability, sensitivity, and convergence speed of delayed systems. These simulated systems were implemented in Simulink for real-time experiments. The Lambert W function assisted in the stability analysis of real systems. The testbed was created for the analysis of delay in real-time systems. Future research can implement the work presented here into the development of a control system for cooperative and coordinated control of MAS in the consensus problem. The rightmost eigenvalues could be used as a key point for eigenvalue assignment or pole placement within a control loop to ensure the system remains stable especially while switching topology. This should guarantee an optimal performance/topology. This work can also be implemented on real network systems, such as drones, DC motors, and ground robots.

REFERENCES

[1] R. Olfati-Saber, A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *Proceedings of the IEEE*, vol. 95, pp. 215-233, 2007. [click]

[2] P. Lin and Y. Jia, “Average consensus in networks of multi-agents with both switching topology and coupling time-delay,” *Physica A: Statistical Mechanics and its Applications*, vol. 387, no.1, pp.303-313, 2008.

[3] P. Yang, M. Liu, X. Lei, and C. Song, “A novel control algorithm for the self-organized fission behavior of flocking system with time delay,” *International Journal of Control, Automation and Systems*, vol. 14, pp 986-997, 2016. [click]

[4] S. Djaidja and Q. Wu, “Leader-following consensus of single-integrator multi-agent systems under noisy and delayed communication,” *International Journal of Control, Automation and Systems*, Volume 14, no. 2, pp 357-366, 2016. [click]

[5] J. P. Richard, “Time-delay systems: an overview of some recent advances and open problems,” *Automatica*, vol. 39, pp. 1667-1694, 2003. [click]

[6] R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” *IEEE Transactions on Automatic Control*, vol. 49, pp. 1520-1533, 2004. [click]

[7] J. A. Fax and R. M. Murray, “Information flow and cooperative control of vehicle formations,” *IEEE Transactions on Automatic Control*, vol. 49, pp. 1465-1476, 2004. [click]

[8] J. Lin, A. Morse, and B. Anderson, “The multi-agent rendezvous problem. An extended summary,” in *Cooperative Control*, ed: Springer, 2005, pp. 257-289. [click]

[9] R. Wei, R. W. Beard, and E. M. Atkins, “A survey of consensus problems in multi-agent coordination,” *Proceedings of American Control Conference*, 2005, pp. 1859-1864. [click]

[10] S. Armah, S. Yi, and W. Choi, "Design of feedback control for quadrotors considering signal transmission delays,” *International Journal of Control, Automation and Systems*, vol. 14, pp. 1395-1403, 2016. [click]

[11] D. Yan, W. Chen, and B. Shuyu, "Decentralized formation control for multiple UAVs based on leader-following consensus with time-varying delays,” *Proc. of Chinese Automation Congress (CAC)*, pp. 426-431, 2013.

[12] L. Huayi, Z. Yingchun, Q. Wenyi, and L. Baohua, “Relative attitude control in satellite formation flying with information delay,” *Proc. of the Sixth World Congress on Intelligent Control and Automation*, pp. 8453-8456, 2006.

[13] R. Saravanakumar, M. S. Ali, J. Cao, and H. Huang, “H∞ state estimation of generalised neural networks with interval time-varying delays,” *International Journal of Systems Science*, vol. 47, no. 16, pp. 3888-3899, 2016.

[14] J. Yu, W. Sun, and H. Liu, “Improved H∞ analysis of Markovian jumping stochastic systems with time-varying delays,” *International Journal of Systems Science*, vol. 44, pp. 521-532, 2013. [click]

[15] Y. Li, S. Tong, and T. Li, “Hybrid fuzzy adaptive output feedback control design for uncertain MIMO nonlinear systems with time-varying delays and input saturation,” *IEEE Transactions on Fuzzy Systems*, vol. 24, pp. 841-853, 2016. [click]

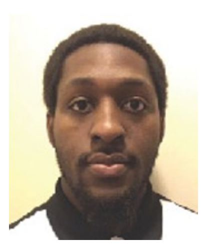
[16] Y. Li and S. Tong, “Adaptive neural networks decentralized FTC design for nonstrict-feedback nonlinear interconnected large-scale systems against actuator faults,” *IEEE Transactions on Neural Networks and Learning Systems*, vo. 99, pp. 1-14, 2017.

[17] F. M. Asl and A. G. Ulsoy, “Analysis of a system of linear delay differential equations,” *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 125, pp. 215-223, 2003. [click]

[18] H. Shinozaki, *Lambert W Function Approach to Stability and Stabilization Problems for Linear Time-delay Systems*, Kyoto Institute of Technology, Thesis, 2007.

[19] S. Yi, *Time-delay Systems: Analysis and Control using the Lambert W function*, World Scientific, 2010.

Myrielle Allen-Prince received her B.S. in Mathematics from Bennett College for Women in 2013. She later attended North Carolina Agricultural & Technical State University, where she received her M.S. Mechanical Engineering in 2016. She is currently a mechanical engineer at Boeing specializing in system modeling and controls.



Christopher Thomas received his B.S. degree for Mechanical Engineering from North Carolina Agricultural & Technical State University in 2016. Since 2016, he has been pursuing his Ph.D. in Mechanical Engineering at North Carolina Agricultural & Technical State University. His main research interests are in the areas of control systems, time delayed systems, teleoperation, Internet communication. His goal is to reach stable teleoperation of a system through a dynamic environment, such as the Internet.

Sun Yi received the B.Sc degree in Mechanical and Aerospace Engineering from Seoul National University, Republic of Korea, in 2004, and the M.Sc. and Ph.D degrees in Mechanical Engineering from University of Michigan, Ann Arbor, in 2006 and 2009, respectively. He is currently an Associate Professor in Department of Mechanical Engineering at the North Carolina A &T State University (NCAT). His current research interests include time-delay systems, and analysis and control of dynamic systems with application to mobile robots and satellites.