Check for updates

Original Article

Institution of MECHANICAL ENGINEERS
Journal of AEROSPACE ENGINEERING

The consensus of non-linear agents under switching topology using dynamic inversion in the presence of communication noise and delay

Proc IMechE Part G:
J Aerospace Engineering
2022, Vol. 236(2) 352–367
© IMechE 2021



Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/09544100211012795
journals.sagepub.com/home/pig



Sabyasachi Mondal and Antonios Tsourdos

Abstract

In this study, a consensus protocol for non-linear multi-agent systems using the non-linear dynamic inversion (NDI) technique is presented. It is named as distributed NDI or DNDI. The agents are considered to be working in a randomly changing environment which is realistic. The randomness in the operating environment is introduced by random switching communication topology, delays, and noise. The consensus protocol is obtained as a closed-form expression, which is a critical requirement for real-time implementation. Also, various cases regarding the communication issues have been considered to study the performance of the DNDI controller. The simulation results are found to be satisfactory.

Keywords

Consensus, multi-agent systems, non-linear dynamic inversion, switching topology, communication delay, communication noise

Date received: 13 August 2020; accepted: 27 March 2021

Introduction

Background

The consensus of multi-agent systems (MASs) is a well-studied research topic. Multi-agent systems are used to execute critical and complex tasks which involve life risk, or urgent attention is required. Examples of these tasks include search and rescue, surveillance, delivery, firefight, and other civil or military operations. Such complex tasks would be completed more effectively or performed co-operatively, but for this, each agent must share information with its neighbour and compute the control action accordingly. A common goal can be achieved if cooperation among agents is possible or otherwise if they could reach a consensus. For consensus to be achieved, the agents must be connected in a meaningful communication topology. Therefore, the consensus is widely dependent on the communication topology.

While there exist several issues associated with communication topologies that can affect the consensus among the agents, the most critical of them is switching topology, communication noise and delay. These events are handled by modifying the consensus protocols designed considering the agent dynamics. Some of the works related to communication issues in MASs are mentioned in the following paragraphs.

Literature review

In MAS communication, switching topology is a common event that occurs primarily due to the changing surrounding, communication range limitation, or nodal link failure. Switching topology has been considered in many research studies. The distributed consensus tracking problem is presented for multi-agent systems having Lipschitz-type node dynamics.[1] In this study, the topology among followers is considered to be changing with time. Wen et al.[2] described the consensus tracking problem where the agents are having second-order non-linear dynamics tracked a time-varying reference state. They considered the time-varying communication topology. The consensus problem for heterogeneous multi-agent systems involving first- and second-order dynamics was addressed by Kim et al.[3] The communication among the agents has probabilistic link failure, which is represented using a Bernoulli probability sequence. Liu et al.[4] solved the leaderless consensus problem of multi-agent systems

Cranfield University, UK

Corresponding author:
Sabyasachi Mondal, Department of Aerospace Engineering, Cranfield University, College Road, Cranfield MK43 0AL, Bedford, UK.
Email: sabyasachi.mondal@cranfield.ac.uk

having Lipschitz non-linear dynamics. They considered communication topology to be directed and switching with time. A consensus problem of multi-agent systems having non-linear dynamics is studied by Ding et al.[5] In this work, the switching of communication topologies is considered to be Markovian in nature. Xia et al.[6] described the leader-following consensus problem considering switching topology.

Communication delay is another event that occurs during the transmission of information between two agents. Time-varying delays arise naturally due to the asymmetry of interactions, the congestion of the communication channels, and the finite transmission speed. An average consensus problem of multi-agent systems is discussed. The communication topology among the agents is considered to be undirected and connected. Also, the communication channel is considered to have time delays. Li et al.[7] discussed the consensus problem considering communication delay. In this study, a class of agents having general third-order dynamics is considered. Zhan et al.[8] focussed on the asynchronous consensus of multiple agents having double-integrator dynamics. The authors solved the consensus problem assuming arbitrary sampling intervals and communication delays. Xu et al. discussed the consensus problem of the agents having single-integrator dynamics with fixed communication topology. One crucial point is that unbounded time-varying transmission delays are considered in this research work. Cui et al.[10] presented a leader-following multi-agent system having non-linear dynamics. The work finds time-varying communication delays among the agents. Jia et al.[11] described the synchronization problem of multi-agent systems with non-linear dynamics. The authors evaluated a time-varying control that is designed, assuming transmission delay over a communication network.

Another practical issue is communication noise which is stochastic and depends on transmission medium characteristics. Cheng et al.[12] described a mean square consensus protocol for multi-agent systems having linear dynamics. They considered channel noise in communication among the agents. Wang et al.[13] presented the consensus of linear multi-agent systems with communication noises. A consensus algorithm for multi-agent with noisy communication is described by Morita et al.[14] Liu et al.[15] discussed consensus with communication noise of MASs having stochastic dynamics.

The studies that consider the combination of switching topology and communication noise are discussed here. Markovian switching topologies, along with communication noises, are regarded by Wang et al.[16] They solved the consensus of multi-agent systems with linear dynamics and multi-input multi-output (MIMO) systems. Kar et al.[17] described the consensus in MASs considering communication link failures and channel noise among agents. Tariverdi et al.[18] presented a fault-tolerant consensus of MASs having non-linear dynamics. This study considered directed communication topologies with link failures and communication noise. Actuator faults were discussed as well. Li et al.[19] considered communication among the agents to be switching and noisy. The switching of the topologies is deemed to be random. Li et al.[20] discussed the consensus considering Markovian switching topologies and communication noises. Moreover, necessary and sufficient conditions for consensus of continuous-time multi-agent systems are derived. Ming et al.[21] also presented the communication channel as Markovian switching topologies along with stochastic communication noise.

Next, the works having the combination of switching topology and communication delay are discussed. Leader-following consensus stability of multi-agent systems is discussed by Park et al.[22] Also, the stabilization is studied with interval time-varying delays. The communication among the agents is Markovian switching and contains delay. Pei et al.[23] considered switching topologies and time delay to solve the consensus of multi-agent systems having discrete-time linear dynamics. Ding et al.[24] described sampled-data leader-following consensus of multi-agent systems having non-linear dynamics. Communication is characterized by random switching network topologies and communication delay. Shang et al.[25] discussed the consensus problem of MASs considering Markovian switching network with time-varying delays. The topologies are also uncertain in nature. Jiang et al.[26] also studied a consensus control problem for multiple robotic manipulators under directed switching topologies with and without time-varying delays.

A few works have been reported where communication delay and noise have been considered to study the consensus among multi-agent systems. Liu et al.[27] showed that uncertainties and communication delays affect the consensus. Zong et al.[28] investigated stochastic consensus conditions of multi-agent systems. In this research article, both time delays and measurement noises are considered to achieve consensus among MASs.

All of the studies discussed so far consider one or two events among switching delay, communication delay, and noise. It can be mentioned that there exists a research study that was conducted by Ming et al.[29] (to the best of the author's knowledge) in which all the issues related to communication among the agents are discussed. However, this work assumed the agent dynamics to be linear time-invariant (LTI), and the consensus protocol was designed accordingly. Therefore, the consensus protocol does not apply to non-linear systems. However, in reality, the agents have non-linear dynamics. Therefore, a protocol is required, which can be applied to the non-linear systems and can handle the communication issues. The scope of finding the consensus protocol of multi-agent systems having non-linear dynamics is explored in this study. Communication among the agents is considered to have all the issues related to communication, that is, switching topology, communication delay, and noise.

The consensus of agents is achieved when the error among similar states converges to zero or the consensus error zero. Therefore, the consensus controller aims to drive error zero. In this context, it can be mentioned that, along with other robust controllers, there exists one

control strategy which enforces the error dynamics and includes plant non-linearities in the control expression. This controller is based on the philosophy of non-linear dynamic inversion (NDI), which uses feedback linearization theory to eliminate the plant's nonlinearities. Moreover, the closed-loop plant responds similarly to a stable linear system. Non-linear dynamic inversion has many advantages, like simple and closed-form control expression, easily implementable, global exponential stability of the tracking error, use of non-linear kinematics in the plant inversion, minimize the need for individual gain tuning or gain scheduling. Non-linear dynamic inversion has been used to design the controller for various purposes. The theory of NDI is used by Enns et al.[30] to design flight controller. Singh et al.[31] used NDI to develop a controller for autonomous landing of unmanned aerial vehicle (UAV). A partial integrated guidance and control (PIGC) is designed for reactive obstacle avoidance of UAVs using neuro-adaptive augmented dynamic inversion by Padhi et al.[32] Another application of NDI is presented by Mondal et al.,[33] where a formation flying scheme is proposed. In this research study, the NDI is used to track the leader's desired position, velocity and orientation. Caverly et al.[34] described the use of NDI for the attitude control of a flexible aircraft. Horn et al.[35] described control design using dynamic inversion. Lombaerts et al.[36] showed NDI-based attitude control of a hovering quad tiltrotor electric vertical take-off and landing vehicle.

The robust non-linear NDI has been widely used for designing controllers for many applications, but for the first time, in this study, we have reformulated NDI to solve the consensus problem of non-linear agents and named it as distributed NDI (DNDI). It inherits all the advantages of NDI discussed in the last paragraph. It is tested in the presence of communication issues like switching topology, delay, and noise. The switching topology is random and changes at each time step. Moreover, the number of switching topologies is not limited, but it can be scaled as the user wants, as it is often in real cases. Also, not all the topologies may have a spanning tree, and thus they may be uncertain when the stability is concerned. The delay is considered in various scenarios like random delays for randomly selected agents. The performance of DNDI to handle the communication issues is studied and discussed in detail in the following sections.

The contribution of this work is summarized as follows.

1. We have addressed the communication issues together, which are rare. Moreover, all issues with non-linear dynamics are not reported. We wanted to explore the scope and found that NDI is an appropriate control strategy that could be beneficial in this context.
2. The application of NDI to the consensus problem is new. We have presented the new derivation in the study and named it distributed NDI or DNDI. We have highlighted the features of NDI along with applications in the introduction. We wanted to study the performance of DNDI in the consensus problem.
3. To highlight the effect of the issue 'switching topology', we have considered unlimited topologies in contrast to other studies where limited topologies are considered. Moreover, we considered the switching of the topologies in a completely random manner to make the simulation scenario more realistic. Communication noise is also random. The communication delay is considered in four cases which are realistic.

The rest of the study is organized as follows. The section 'Preliminaries' gives a brief description of the consensus of MASs, graph theory, switching topology, communication noise, delay, and NDI. The section 'Problem Description' gives a formal problem statement. The next section, 'NDI Controller for Consensus of MASs', presents the mathematical details of the DNDI. The 'Result' section presents the simulation results and their analysis. Finally, the conclusion is given.

Preliminaries

The topics which are relevant to the problem considered in this study are given in this section.

Consensus of multiple agents

The consensus of MASs on communication network is discussed in this section. The definition of the consensus is given as follows.

Definition 1. Let us consider a MAS with N agents, where X_i, ($i = 1, 2, 3, \ldots, N$) denotes the states of the ith agent. The MASs will achieve the consensus if $\|X_i - X_j\| \rightarrow 0$, $\forall i \neq j$ as $t \rightarrow +\infty$.

The primary goal of designing a consensus protocol is to minimize the error in similar states of each individual agent with its neighbour by exchanging information among them through the communication network, which is designed and explained by graph theory.

Graph theory

The networked MAS is represented by a weighted directed graph written by $G = \{V, E\}$. The vertices $V = \{1, 2, \ldots, N\}$ of the graph denote the agents, and the set of edges is denoted by $E \subseteq V \times V$ represents the communication among the agents. e_{ij} denotes the information flow along the edge from j to i. The neighbour of agent i is denoted by $N_i = \{j \in V: (i, j) \in E\}$. The Adjacency matrix $A = [a_{ij}] \in \Re^{N \times N}$ which denotes the connectivity among the nodes or agents. a_{ij} denotes the elements of A. There is no self-loop in the graph. This fact is expressed by selecting the diagonal elements of the adjacency matrix A as zero, that is, $i \in V$, $a_{ii} = 0$. The off-diagonal elements, that is, $\forall\ i \neq j$, $e_{ij} \in E$, $a_{ij} > 0$ represents the weight associated to edge e_{ij}, while $a_{ij} = 0$ otherwise. The degree matrix is denoted by $D \in \Re^{N \times N} = \text{diag}\{d_1 d_2 \ldots d_N\}$, where $d_i = \sum_{j \in N_i} a_{ij}$. The Laplacian

matrix is written as $L = D - A$. All the matrices describe the connections and properties of the consensus among MASs.

There exist several issues with communication among MASs, which affect the consensus. The issues like switching topology/link failure, communication noise and delay are common in practice. They are discussed in the following sections.

Time delay in communication channel

The communication delay is a practical event in a MAS platform. The MASs exchange information among themselves to achieve the consensus. The agents in the synchronous network receive information from the neighbours at the same time instant; that is, they share the same clock frequency. But in a practical scenario, the situation arises where some of the agents or all of the agents may have delay. The delay may be counted by time stamping the incoming data from the neighbours. The ith agent updates its own and neighbour's states at time instants $t_0^i, t_1^i, \ldots, t_k^i, \ldots$. The agent i receives the state information of jth neighbour $j \in N_i$ at t_k^i with a delay of τ_{ij}^k, that is, $X_j(t_k^i - \tau_{ij}^k)$ where $0 < \tau_{ij}^k < \eta(t_{k+1}^i - t_k^i)$ and $\eta = 0, 1, 2, \ldots$. If a neighbour has $\eta = 1$, then the information of jth agent at t_{k-1}, that is, $X_j(t_{k-1})$ is received by agent i at t_k^i since $\tau_{ij}^k < (t_k^i - t_{k-1}^i)$ or $t_{k-1}^i < t_{k-1}^i + \tau_{ij}^k < t_k^i$. Pictorial representation of the delay is given in Figure 1.

Noise in communication channel

In this section, the effect of communication noise on the consensus of agents is highlighted. The noise is added to state information transmitted by the agents. Therefore, the noisy information is received by ith agent from its neighbours. The noisy information received by ith agent from jth neighbour $j \in N_i$ can be modelled as $\overline{X}_{ji} = X_{ji} + \sigma_{ji}\omega_{ji}$, where $X_i, X_j \in \Re^n$ are states, ω_{ji}; $i, j \in 1, 2, \ldots, N$ are independent standard white noises and σ_{ji} is the noise intensity.

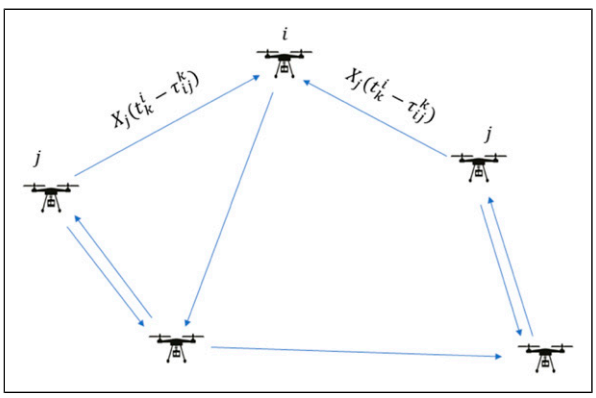


Figure 1. Description of communication delay in multi-agent systems network.

Switching topology

In this study, communication topologies are considered to be changing in a random manner. This scenario is presented by generating many topologies, and one of them is selected at each random time interval. The number of switching topologies can be selected by the user; that is, topologies are not limited. The topologies are defined by the adjacency matrices. The procedure for generating random graphs (G_i) or adjacency matrix is given in Algorithm 1.[37] A_k defines the kth topology, $k = 1, 2, \ldots, N_p$, and N_p is the number of topologies. i, j defines the position of an element in the adjacency matrix A_k. The algorithm generates N_p adjacency matrices which are square in dimension. The value at each i, j position is dependent on a random variable x. If the value of x is greater than 0.5, then $A_k(i, j) = 1$, otherwise $A_k(i, j) = 0$. The threshold value 0.5 is considered to obtain 0 and 1 with equal probability, but the value can be changed. It is important to note that the graphs should contain a spanning tree to achieve consensus. But the graphs generated may not contain a spanning tree always. Therefore, the performance of an NDI-based controller is evaluated considering this factor.

Algorithm 1 Generation of Random Graphs

```
for k = 1 to N_p do
  for i = 1 to N do
    for j = 1 to N do
      x ← random number x ∈ (0, 1)
      if x > 0.5 then
        A_k(i, j) ← 1
      else
        A_k(i, j) ← 0
      end if
      if i = j then
        A_k(i, j) ← 0
      end if
    end for
  end for
end for
```

The selection of a topology among N_p topologies at each time instant is also random, as mentioned earlier. The selection procedure is shown in Algorithm 2. A_S denotes the selected topology. The variable i denotes the time instants. The switching depends on the value of x. The variable x takes a random value between 0 and 1. If the value is greater than 0.5, the topology will change. The function random_integer ([1 N_p], 1) generates a random integer ind, in the range [1 N_p]. The selected topology A_S is the topology corresponding to the integer ind, that is, $A_S = A_{\text{ind}}$. If the value is less than 0.5, that is, $x \leq 0.5$, the topology does not change.

Algorithm 2 Random Selection of Topology

```
for i = 1 to n do
  x ← random number x ∈ (0, 1)
  if x > 0.5 then
    ind ← random_integer([1 N_p], 1)
    A_S ← A_ind
  else
    A_S remains same
  end if
end for
```

Non-linear dynamic inversion

Non-linear dynamic inversion is a potential candidate for designing the state feedback controller for the non-linear systems. A brief description of NDI is presented here. Let us consider a class of affine non-linear dynamics as follows.

$$\dot{X} = f(X) + g(X)U \tag{1}$$

$$Y = X \tag{2}$$

where $X \in \mathfrak{R}^n$, $U \in \mathfrak{R}^n$ and $Y \in \mathfrak{R}^n$ are state, control and the output, respectively. f is a continuously differentiable vector-valued function representing the intrinsic non-linear dynamics.

Remark 1. It can be noted that the system in equations (1) and (2) represents a specific class of non-linear systems. However, they are used in many practical applications. A few examples include the control design of decoupled three-loop guidance and control design of autonomous vehicles. In these cases, the number of states, output and control inputs is same for each loop.[38,39]

Assumption 1. *The system is perfectly known, and the matrix $g(X)$ is invertible for all time.*

The objective of NDI is to design a state feedback control U such that the output of system Y to track a bounded and smooth reference signal Y^* with time t. The tracking error is given by

$$e = Y - Y^* \tag{3}$$

The first-order error dynamics is written as

$$\dot{e} + Ke = 0 \tag{4}$$

It can be mentioned that the gain is considered to be positive definite, that is, $K > 0$. The expression of the tracking error, that is, e, is substituted in equation (4) to obtain the control expression as follows

$$U = (g(X))^{-1}\left[-f(X) - K(Y - Y^*) + \dot{Y}^*\right] \tag{5}$$

Assumption 2. *$g(X)$ is non-singular for all the time.*

Problem description

The problem objective in this study is to design a consensus control for MASs using NDI. The MASs consist of N agents. They are connected by the network topology. All the agents are homogeneous, that is, they have similar dynamics. In this study, the dynamics of each agent is the same as given in equations (6) and (7). The dynamics for i^{th} agent is given by

$$\dot{X}_i = f(X_i) + g(X_i)U_i \tag{6}$$

$$Y_i = X_i \tag{7}$$

The state vector of ith agent is given by $X_i \in \mathfrak{R}^n$. The output of ith agent is given by

$$Y_i = X_i \in \mathfrak{R}^n \tag{8}$$

The agents are operating in a randomly changing environment. The communication among the agents faces issues like switching topology, noise and delay, which are described in the preliminaries section.

Non-linear dynamic inversion controller for consensus of MASs

The derivation of NDI-based control for MASs is presented in this section. In case of consensus of MASs, the reference signal or desired output of each agent is the outputs of its neighbours. First, the error for scalar output is obtained. The error in output of ith agent can be written as

$$e_i = \sum_{j \in N_i} a_{ij}\left(Y_i - Y_j\right) \tag{9}$$

where j denotes the jth agent of ith agent's neighbourhood N_i. Equation (9) can be simplified as follows

$$e_i = d_i Y_i - a_i Y \tag{10}$$

where

$$d_i = \sum_j a_{ij} \in \mathfrak{R}, \quad a_i = \begin{bmatrix} a_{i1} & a_{i2} & \dots & a_{iN} \end{bmatrix} \in \mathfrak{R}^N$$

and

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix} \in \mathfrak{R}^N$$

The error in equation (10) is written for vector output of ith agent, that is, $Y_i \in \mathfrak{R}^n; n > 1$ as

$$\mathbf{e}_i = \overline{d} Y_i - \overline{a}\mathbf{Y} \tag{11}$$

where $\overline{d} = (d_i \otimes \mathbb{I}_n) \in \mathfrak{R}^{n \times n}$, $\overline{a} = (a_i \otimes \mathbb{I}_n) \in \mathfrak{R}^{n \times nN}$ and $\mathbf{Y} \in \mathfrak{R}^{nN}$. $\mathbb{I}_n$ is $n \times n$ identity matrix. '$\otimes$' denotes the Kronecker product. The Kronecker product of $A = [a_{ij}]$ and B is given by

$$A \otimes B = \begin{bmatrix} a_{ij}B \end{bmatrix} \quad \forall i,j$$

We define a Lyapunov function V_i as follows

$$V_i = \frac{1}{2}\mathbf{e}_i^T\mathbf{e}_i \tag{12}$$

Differentiating equation (12) yields

$$\dot{V}_i = \mathbf{e}_i^T\dot{\mathbf{e}}_i \tag{13}$$

According to the Lyapunov stability theory, let the time derivative of the Lyapunov function should be

$$\dot{V}_i = -\mathbf{e}_i^T K\mathbf{e}_i \tag{14}$$

where $K \in \Re^{n\times n}$ is a positive definite diagonal matrix. The expression of $\dot{V}_i$ in equations (13) and (14) is equated to obtain

$$\mathbf{e}_i^T\dot{\mathbf{e}}_i = -\mathbf{e}_i^T K\mathbf{e}_i \tag{15}$$

Equation (15) is simplified as follows

$$\dot{\mathbf{e}}_i + K\mathbf{e}_i = 0 \tag{16}$$

Differentiation of equation (11) yields

$$\dot{\mathbf{e}}_i = \overline{d}\dot{\mathbf{Y}}_i - \overline{a}\dot{\mathbf{Y}} = \overline{d}[f(X_i) + g(X_i)U_i] - \overline{a}\dot{\mathbf{Y}} \tag{17}$$

Substitution of the expressions for $\mathbf{e}_i$ and $\dot{\mathbf{e}}_i$ in equation (16) gives

$$\overline{d}[f(X_i) + g(X_i)U_i] - \overline{a}\dot{\mathbf{Y}} + K\left(\overline{d}Y_i - \overline{a}\mathbf{Y}\right) = 0 \tag{18}$$

The expression of control U_i for ith agent is obtained by simplifying equation (18) as follows

$$U_i = (g(X_i))^{-1}\left[-f(X_i) + \overline{d}^{-1}\left(\overline{a}\dot{\mathbf{Y}} - K\left(\overline{d}Y_i - \overline{a}\mathbf{Y}\right)\right)\right] \tag{19}$$

It is clear that the control expressions of conventional NDI are different from what is obtained for consensus of MASs. This control expression in equation (19) is used to generate results which shown in the following section.

Results

The simulation study is performed to evaluate the performance of the NDI-based controller in achieving consensus in the presence of different issues related to communication discussed before. The performance is evaluated under randomly changing behaviour of communication. Five cases are considered as follows.

Case 1: Consensus using NDI
Case 2: Consensus using NDI with Switching Topology
Case 3: Consensus using NDI with Communication Noise
Case 4: Consensus using NDI with Communication Delay
Case 5: Consensus using NDI with Switching Topology, Communication Delay and Communication Noise

These cases are simulated with the same initial conditions and for the same simulation duration. The simulation study is performed on PC with AMD Ryzen five processor and 8 Gb RAM. The gain of each agent K_i is same, that is

$$K_i = \begin{bmatrix} 8 & 0 \\ 0 & 10 \end{bmatrix}$$

The gain is selected in a random manner. The user can change the gain and obtain the response accordingly. Any specific method is not used to find the gain because the aim of this work is to focus on how the NDI is used for the consensus of agents having non-linear dynamics. The dynamics are given in the following section.

Agent dynamics considered for simulation study

The non-linear dynamics considered for the ith agent is given by

$$\dot{X}_{i_1} = X_{i_2}\sin(2X_{i_1}) + U_{i_1} \tag{20}$$

$$\dot{X}_{i_2} = X_{i_1}\cos(3X_{i_2}) + U_{i_2} \tag{21}$$

where $X_i = [X_{i_1} \quad X_{i_2}]^T$. Putting the dynamics of equations (20) and (21) in the form given in equation (6) gives

$$f(X_i) = \begin{bmatrix} X_{i_2}\sin(2X_{i_1}) \\ X_{i_1}\cos(3X_{i_2}) \end{bmatrix} \tag{22}$$

and

$$g(X_i) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{23}$$

and

$$U_i = \begin{bmatrix} U_{i_1} \\ U_{i_2} \end{bmatrix} \tag{24}$$

The output Y_i is chosen as follows

$$Y_i = X_i \tag{25}$$

where $Y_i \in \Re^2$. In this simulation study, 10 agents are considered. The global state vector of the agents is given by

$$X_1 = [X_{1_1} \quad X_{2_1} \quad \ldots \quad X_{10_1}] \tag{26}$$

and

$$X_2 = [X_{1_2} \quad X_{2_2} \quad \ldots \quad X_{10_2}] \tag{27}$$

Also, the global control vector is given by

$$U_1 = [U_{1_1} \quad U_{2_1} \quad \ldots \quad U_{10_1}] \tag{28}$$

and

$$U_2 = [U_{1_2} \quad U_{2_2} \quad \ldots \quad U_{10_2}] \tag{29}$$

The initial conditions of the states X_1 and X_2 are generated using MATLAB command 'randi' in the range $[-10, 10]$.

Case 1. Consensus using NDI controller

In this section, the results obtained using NDI are shown. In this case, communication delay, noise, and switching topologies are not considered. The states X_1 and X_2 achieved a common value or consensus in a few seconds, as shown in Figures 2 and 3, respectively. It can be mentioned that the time of convergence of the states depends on the gain K. The trajectories of the agents are shown in different colours. The controls for the consensus of ith agent, that is, U_1 and U_2, are calculated using DNDI. Both control signals of all the agents are shown in Figures 4 and 5, respectively. The controls have driven the states of all agents to the consensus, which can be understood from the error plot. The consensus error e_i in states X_1 and X_2 of all the agents is shown in Figures 6 and 7, respectively. It is clear that the error converges to zero in few seconds.

Case 2. Consensus using NDI with switching topology

The consensus of the agents is discussed in this section considering switching topology. Other events, like communication delay and noise, are not included. It is important to note that in this work, random switching topologies are considered, as discussed in section. The number of switching topologies considered in this work is

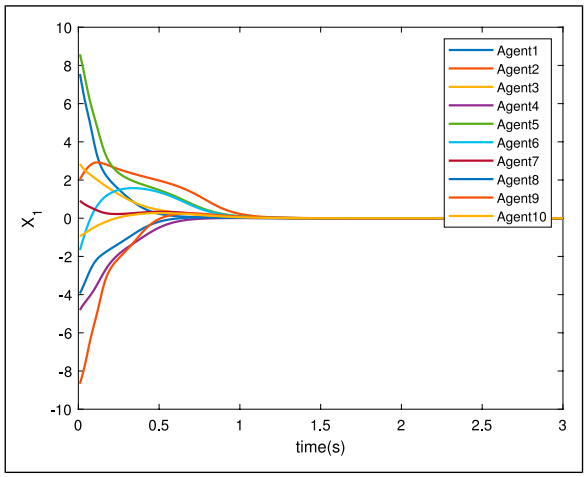


Figure 2. Consensus of state X_1 (Case 1).

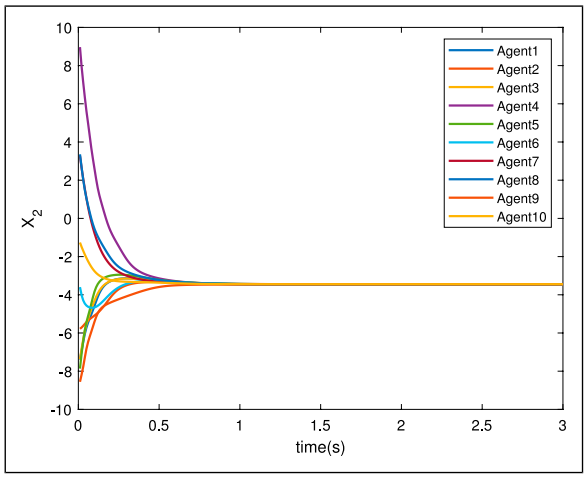


Figure 3. Consensus of state X_2 (Case 1).

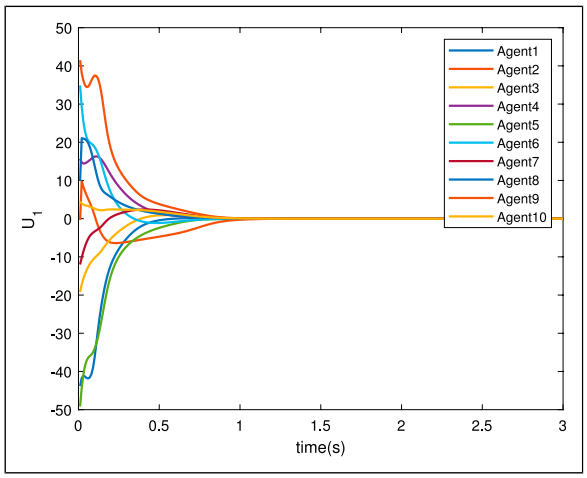


Figure 4. Control U_1 calculated using non-linear dynamic inversion (Case 1).

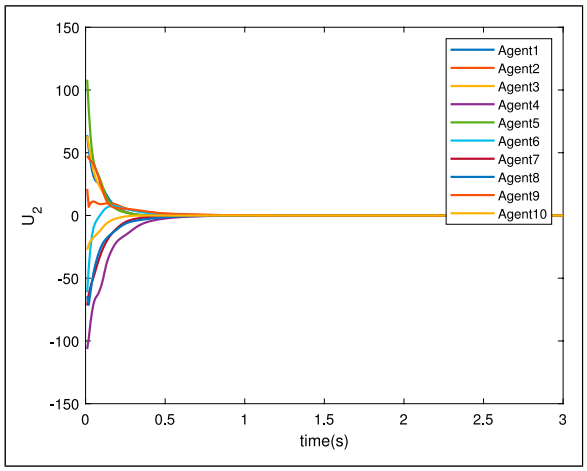


Figure 5. Control U_2 calculated using non-linear dynamic inversion (Case 1).

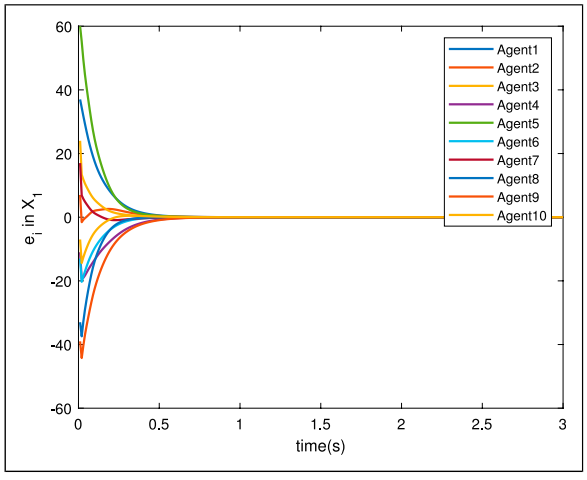


Figure 6. Error in X_1 calculated using non-linear dynamic inversion (Case 1).

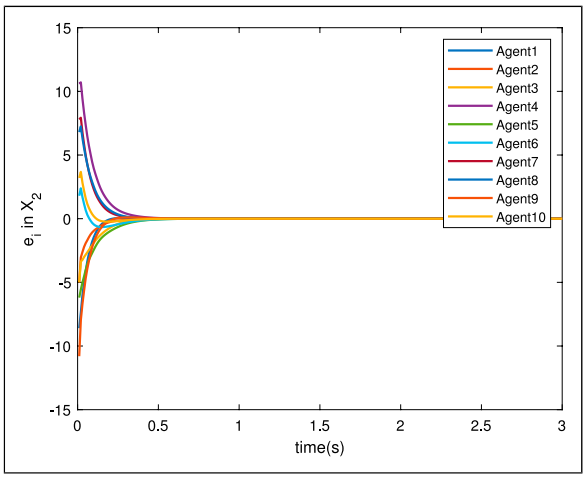


Figure 7. Error in X_2 calculated using non-linear dynamic inversion (Case 1).

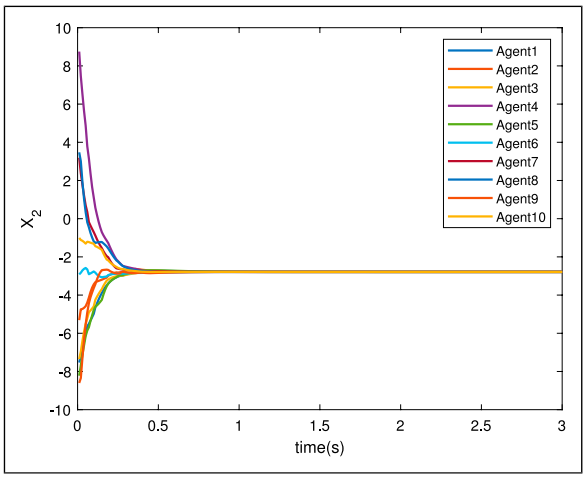


Figure 9. Consensus of states X_2 under switching topology (Case 2).

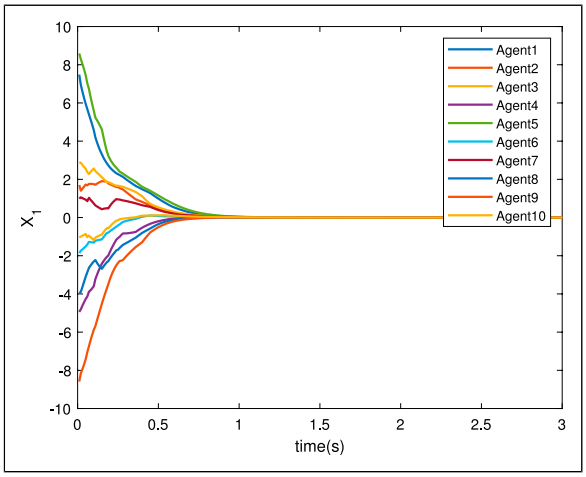


Figure 8. Consensus of states X_1 under switching topology (Case 2).

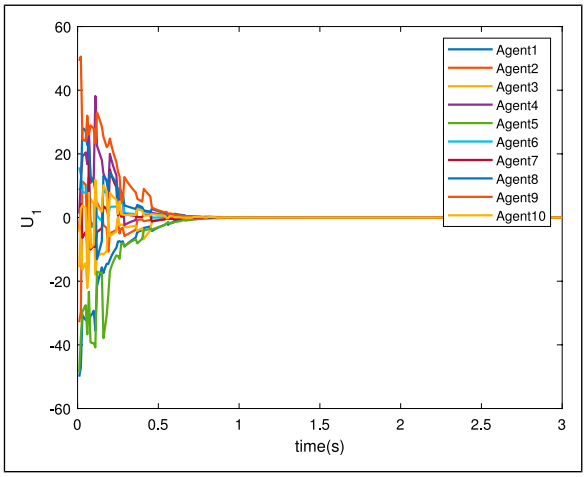


Figure 10. Control U_1 calculated using non-linear dynamic inversion under switching topology (Case 2).

100, that is, $N_p = 100$. At each time instant, one of these topologies is selected.

The state trajectories of all the agents, that is, X_1 and X_2, are shown in Figures 8 and 9, respectively. The DNDI is able to make the states converge to a common value successfully in the presence of the effect of switching topology. The difference between the state trajectories for Case 1 and Case 2 can be observed. The state trajectories in Case 2 are not as smooth as those obtained in Case 1. The corresponding control signals U_1 and U_2 are shown in Figures 10 and 11, respectively. It is clear that the control signals have the effect of switching topology. The switching signal is shown in Figure 12. The 'y' axis shows the topology number selected at each time instant, and the 'x' axis shows the time in second. The topologies are selected according to Algorithm 2. It can be observed that the topologies toggle randomly over time. The control signals are calculated considering these switching topologies. Therefore, the difference in control signals between Case

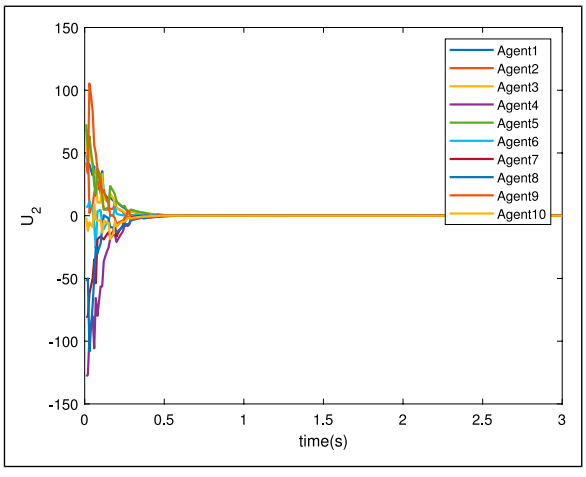


Figure 11. Control U_2 calculated using non-linear dynamic inversion under switching topology (Case 2).

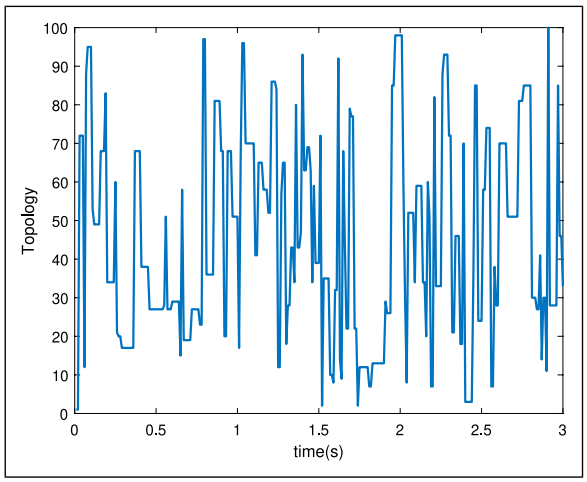


Figure 12. Random switching topology (Case 2).

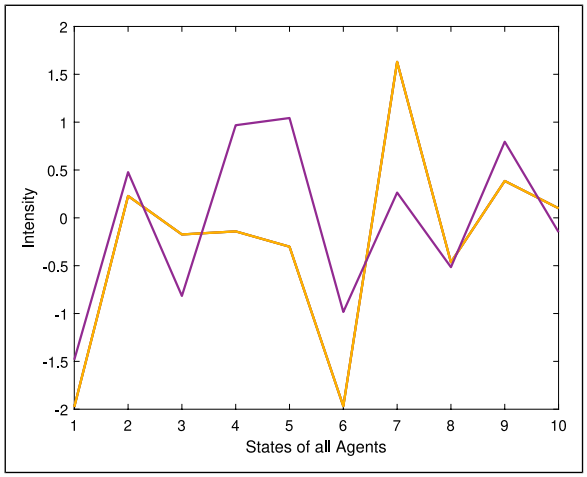


Figure 13. White Gaussian noise added to the agent states at a particular time instant (Case 3). X_1 and X_2 states are shown in purple and yellow colour.

1 and 2 is obvious. It can be noted that each of the selected topologies may not contain a spanning tree, but NDI can achieve consensus among the agents.

Case 3. Consensus using NDI with communication noise

In this section, the effect of communication noise on the consensus of agents is studied; other effects are not considered. The noise considered in this study is assumed to be white Gaussian. The noise is generated using the MATLAB function 'WGN'. The intensity is selected as $\sigma_{ij} = 0.015$. The random noise added to both the states of all the agents at a time instant is shown in Figure 13. The effect of the noise can be observed in the state trajectories shown in Figures 14 and 15. The trajectories are affected by the noise, but the NDI controller is able to achieve the consensus in finite time. The control signals corresponding to the state trajectories are shown in Figures 16 and 17.

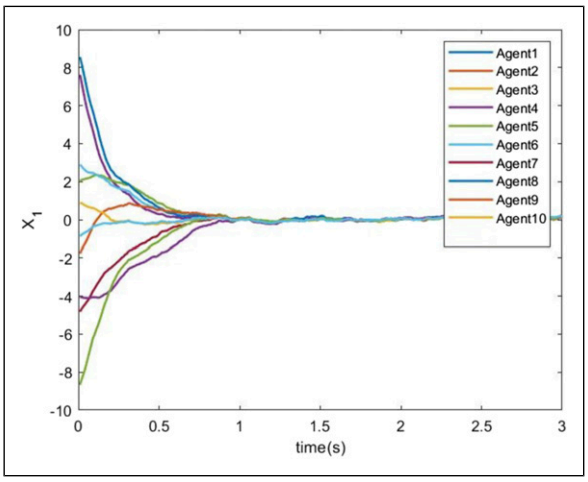


Figure 14. Consensus of states X_1 by non-linear dynamic inversion with communication noise (Case 3).

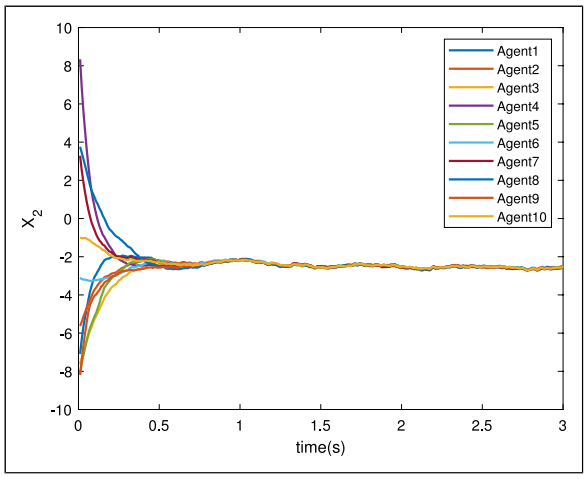


Figure 15. Consensus of states X_2 by non-linear dynamic inversion with communication noise (Case 3).

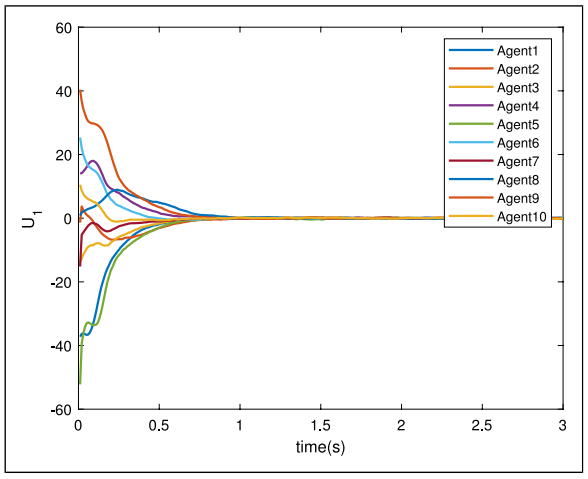


Figure 16. Control U_1 calculated using non-linear dynamic inversion with communication noise (Case 3).

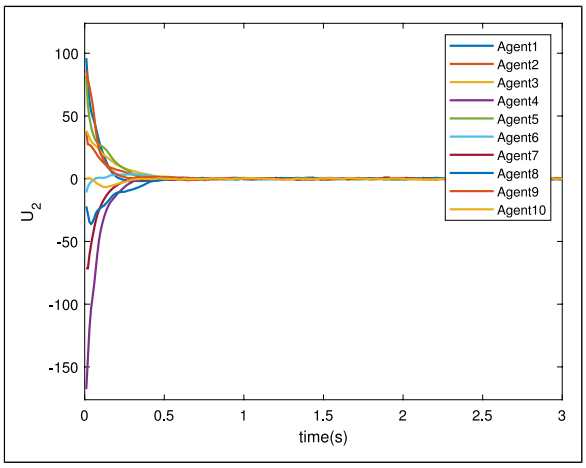


Figure 17. Control U_2 calculated using non-linear dynamic inversion with communication noise (Case 3).

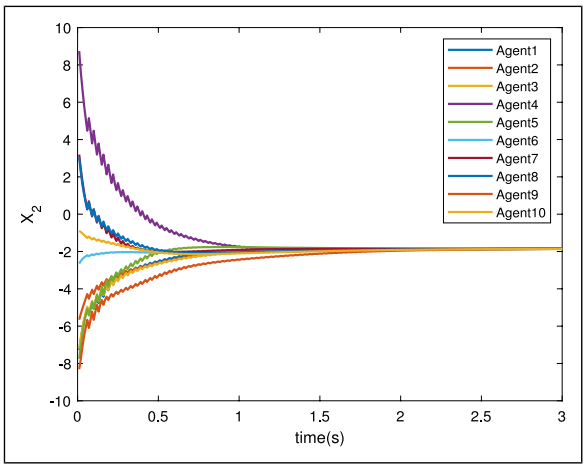


Figure 19. Consensus of state X_2 with communication delay (Case 4.1).

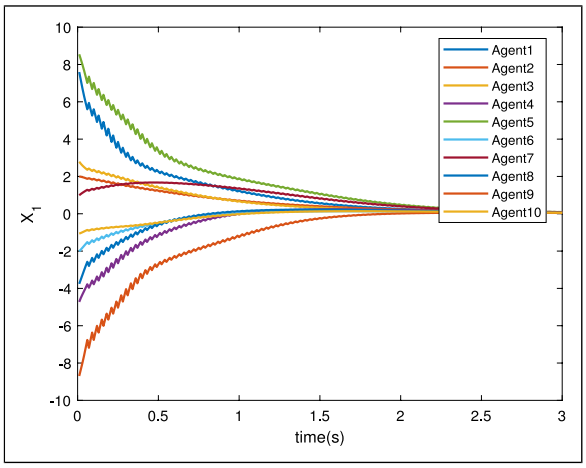


Figure 18. Consensus of state X_1 with communication delay (Case 4.1).

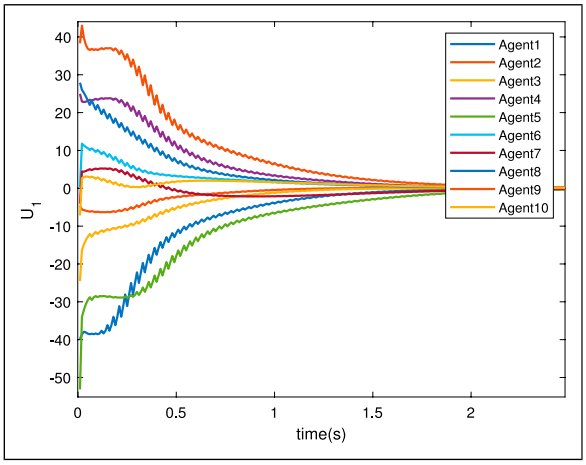


Figure 20. Control U_1 calculated using non-linear dynamic inversion with communication delay (Case 4.1).

Case 4. Consensus using NDI with communication delay

The effect of communication delay on consensus is studied considering different scenarios which are described in following sections.

Case 4.1. All agents with same constant delay

In this scenario, all the agents are assumed to have the same constant delays, that is, each agent receives information about states of its neighbours not in real time but with a delay of two time steps, that is, at t_k^i, agent i receives the state information $X_j(t_k^i - \tau_{ij}^k)$ with $\eta = 2$. The state trajectories X_1 and X_2 are shown in Figures 18 and 19, respectively. The effect of the delay can be observed in the state trajectories as ripples appear. The control signals corresponding to the state trajectories are shown in Figures 20 and 21. Although the delay is present, the NDI controller is successful in achieving the consensus.

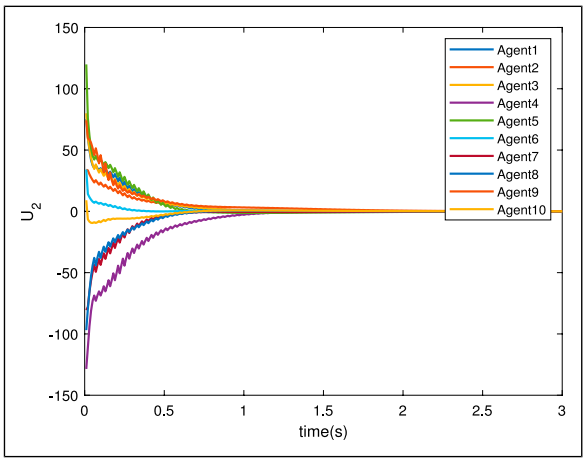


Figure 21. Control U_2 calculated using non-linear dynamic inversion with communication delay (Case 4.1).

Case 4.2. Fixed agents with different constant delay

In this case, some agents have a constant delay. But the values of delay are different and remain constant for the simulation duration. One example of the agents and their corresponding delays is shown in Table 1. The simulation results are presented considering the values in the table. The delays can vary between one and two-unit time steps; that is, some of them have a delay of one unit time step,

Table 1. Fixed agents and their delay.

Agent	Delay(time step)
1	1
4	2
5	1
9	2
10	1

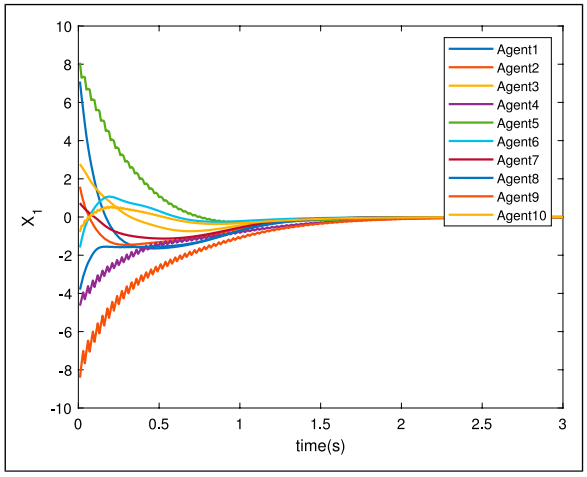


Figure 22. Consensus of state X_1 for fixed agents with constant delay (Case 4.2).

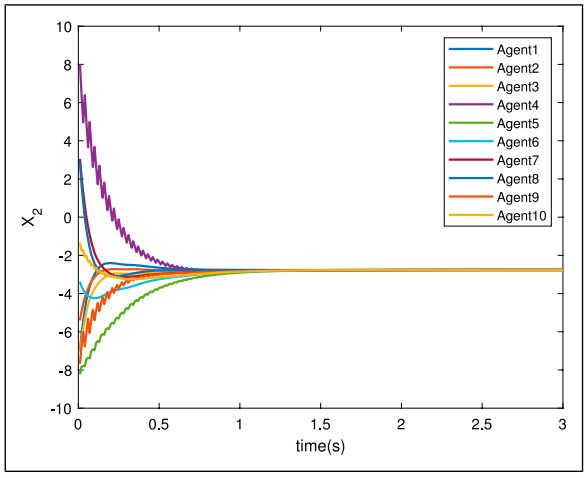


Figure 23. Consensus of state X_2 for fixed agents with constant delay (Case 4.2).

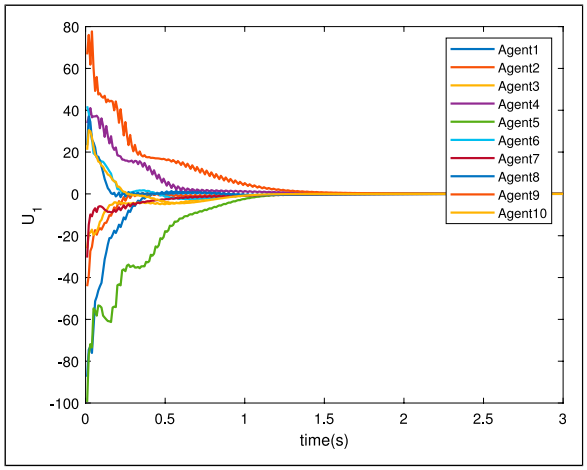


Figure 24. Consensus control U_1 for fixed agents with constant delay (Case 4.2).

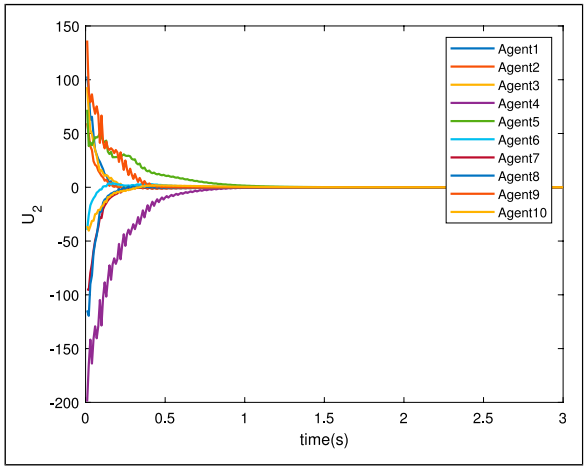


Figure 25. Consensus control U_2 for fixed agents with constant delay (Case 4.2).

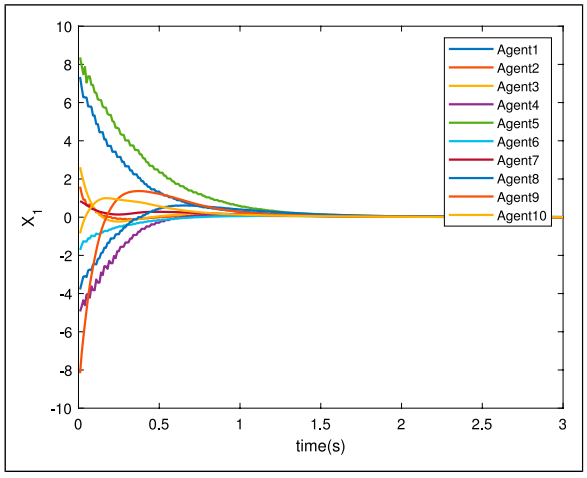


Figure 26. Consensus of state X_1 for fixed agents with variable delay (Case 4.3).

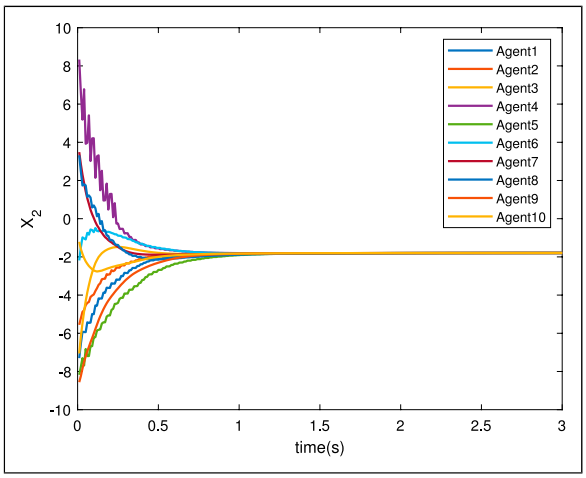


Figure 27. Consensus of state X_2 for fixed agents with variable delay (Case 4.3).

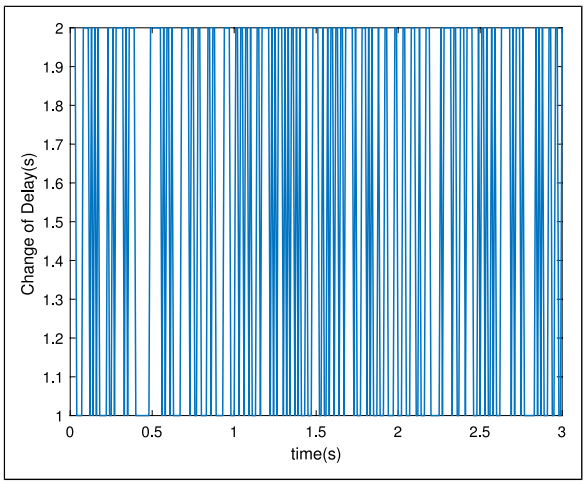


Figure 28. Toggling of delay for agent 4 (Case 4.3).

and the rest of them have two, as shown in the table. The state trajectories are shown in Figures 22 and 23. It can be observed in both the figures that ripples in the trajectories corresponding to the agents in the table are different. This is obvious because they have different delays, as shown in the table. The control signals corresponding to the trajectories are shown in Figures 24 and 25, receptively. The control signals have ripples as expected.

Case 4.3. Fixed agents with variable delay

In this case, only fixed agents have delays, and they are varying at each time instant. The delays toggle between one and two time steps. It can be mentioned that the toggling is random, and they are generated by the MATLAB function 'randi'. The agents 1, 2, 4, 5, 6, and 8 agents have delays. The state trajectories of the agents are shown in Figures 26 and 27. An example of toggling of delays (of agent 4) is shown in Figure 28. It shows the agent delay toggles between one and two-unit time steps.

The control signals of the agents are shown in Figures 29 and 30.

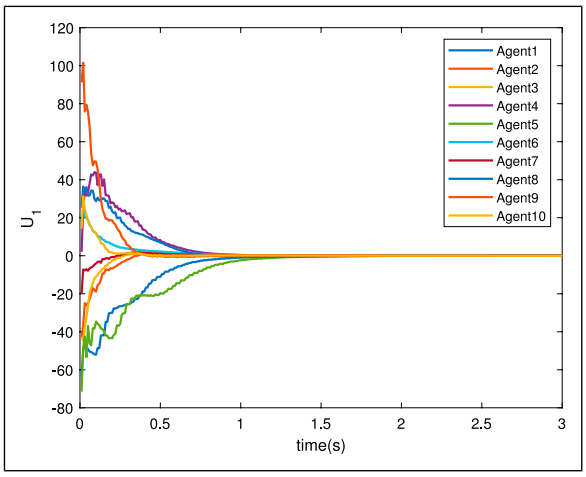


Figure 29. Consensus control U_1 for fixed agents with variable delay (Case 4.3).

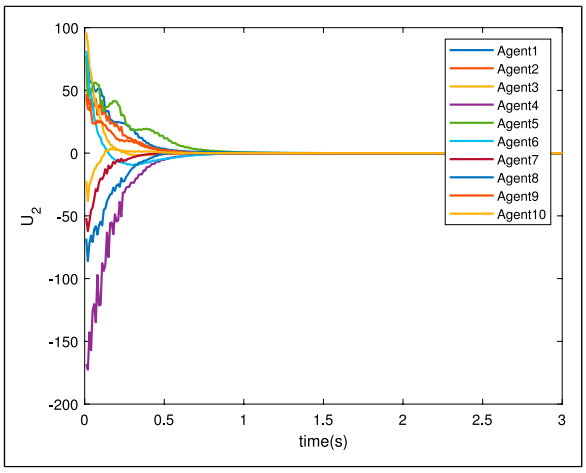


Figure 30. Consensus control U_2 for fixed agents with variable delay (Case 4.3).

Case 4.4. Variable agents with variable delay

In this case, some agents are selected randomly at each time instant with a random delay ranging between one and two unit time steps. At time instant t_k, a set of agents are selected in a random manner, and a set of delays are considered. In the next time instant t_{k+1}, another set of agents are selected randomly with a different set of delays. It can be mentioned that the number of agents that would be selected at each time instant is also random. This number is generated by using the MATLAB function 'randi'. Also, the delays are selected in a similar way. For example, let us assume that at a time instant t_k^i, the selected agents are 2^{nd} and 4^{th}, with a time delay of one and two unit time steps, respectively. In the next instant, the agents selected are 3^{rd}, 4^{th} and 7^{th} with the delays of 1, 2 and 1 unit time steps, respectively. The state trajectories are shown in Figures 31 and 32.

The control signals are shown in Figures 33 and 34. The agents selected at each time instant is shown in Figure 35. The number of agents is allowed to vary between two and eight.

In general, the state and control history of the agents looks different from each other. This is due to the different cases that we have considered.

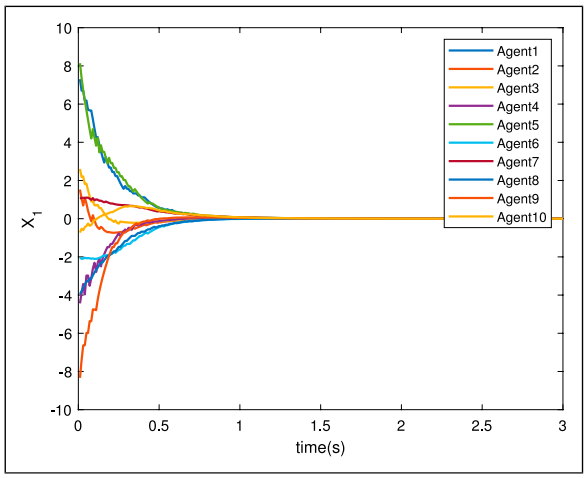


Figure 31. Consensus of state X_1 for variable agents with variable delay (Case 4.4).

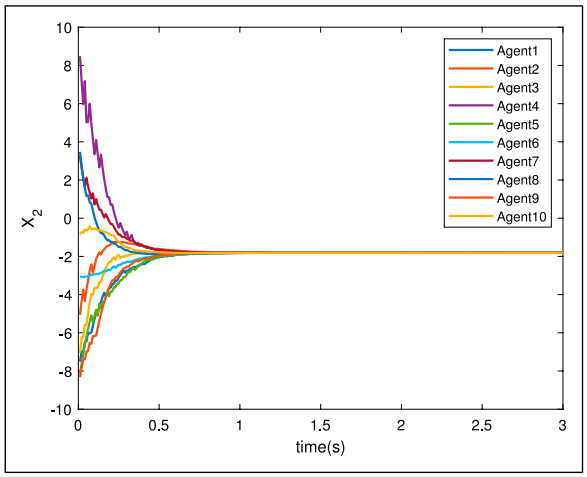


Figure 32. Consensus of state X_2 for variable agents with variable delay (Case 4.4).

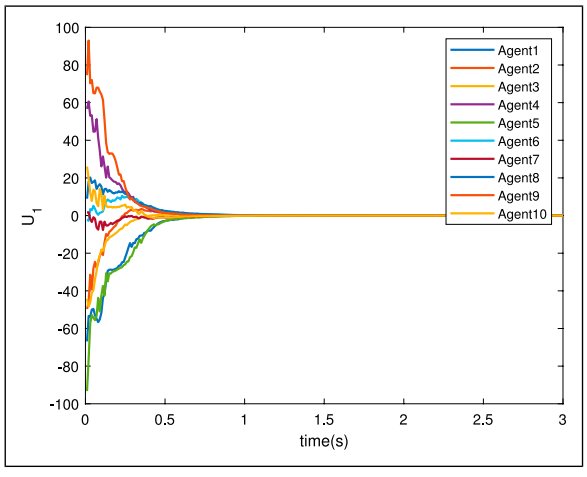


Figure 33. Consensus control U_1 for variable agents with variable delay (Case 4.4).

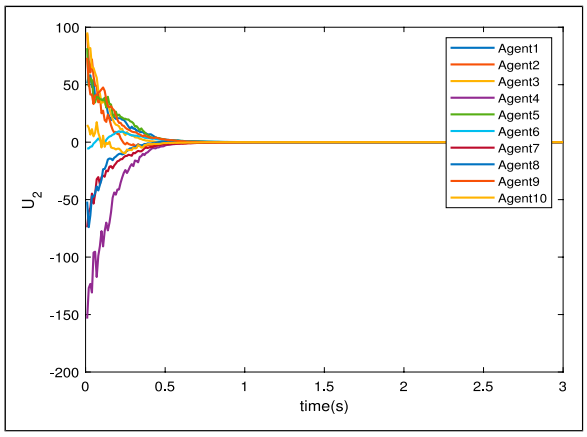


Figure 34. Consensus control U_2 for variable agents with variable delay (Case 4.4).

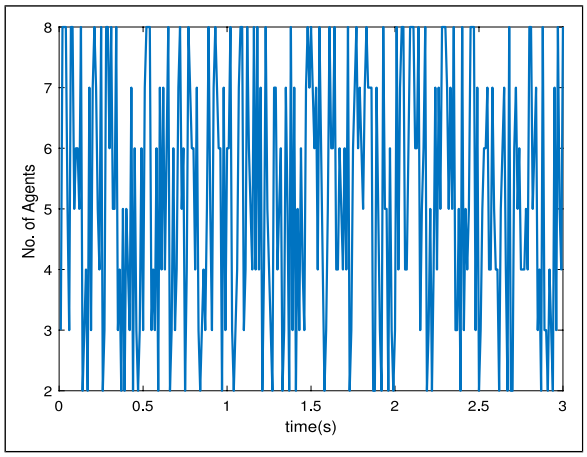


Figure 35. Number of agents selected at each time instant (Case 4.4).

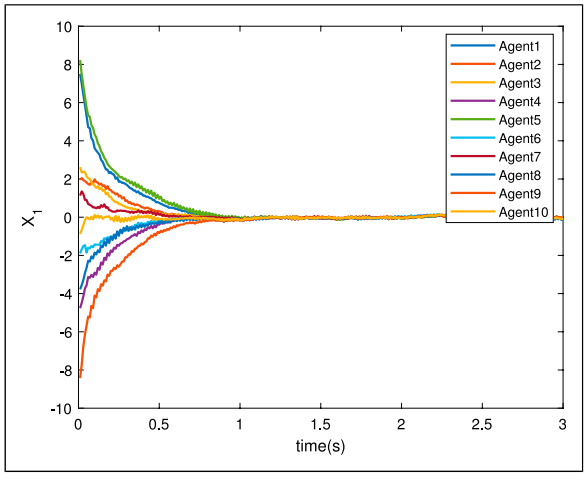


Figure 36. Consensus of state X_1 under switching topology, in the presence of communication noise and delay (Case 5).

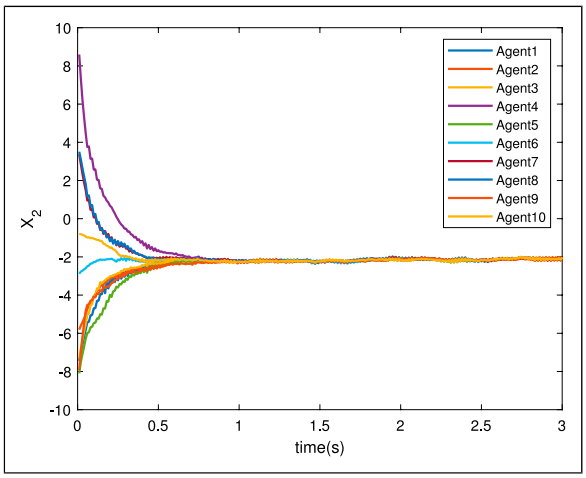


Figure 37. Consensus of state X_2 under switching topology, in the presence of communication noise and delay (Case 5).

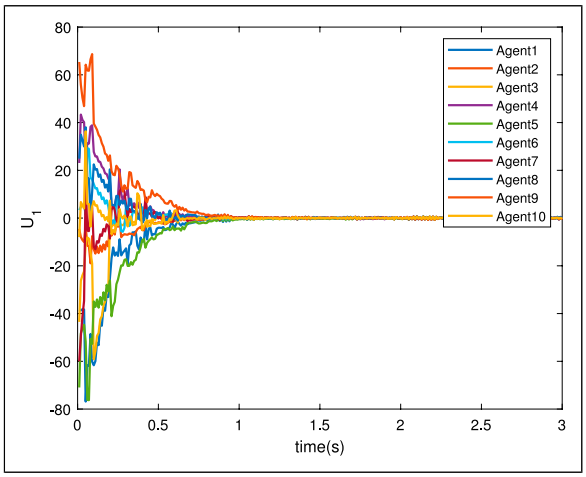


Figure 38. Control U_1 calculated using non-linear dynamic inversion under switching topology, in the presence of communication noise and delay (Case 5).

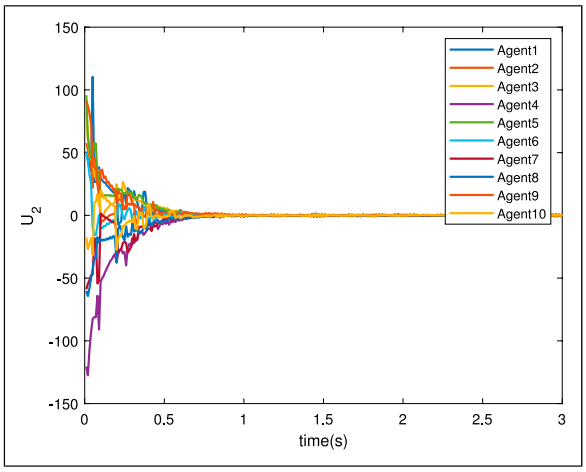


Figure 39. Control U_2 calculated using non-linear dynamic inversion under switching topology, in the presence of communication noise and delay (Case 5).

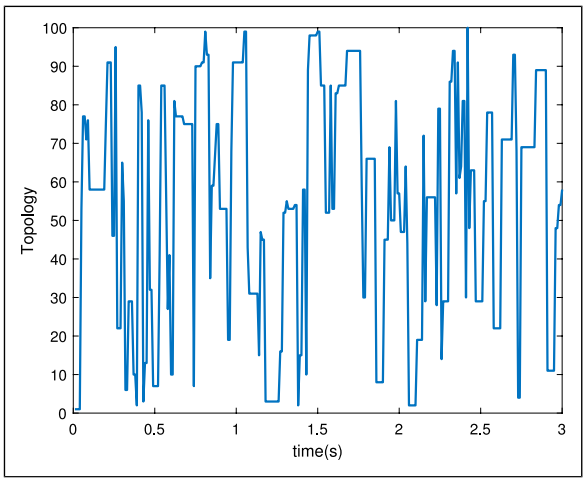


Figure 40. Switching signal (Case 5).

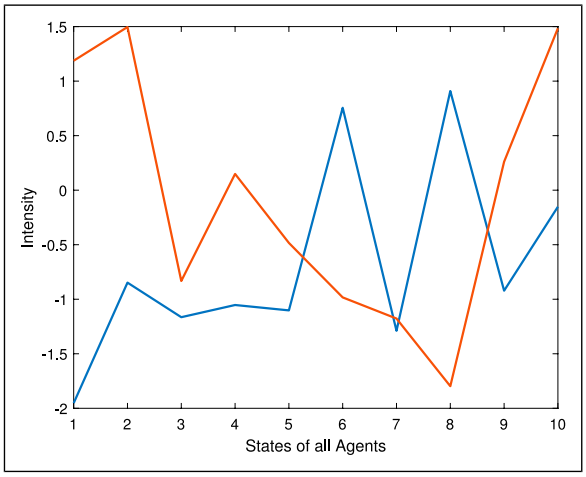


Figure 41. White Gaussian noise added to the states of all agents at a particular time instant (Case 5).

Case 5. Consensus using NDI with switching topology, communication noise and communication delay

The performance of NDI considering the combined effect of switching topology, communication noise, and delay is discussed in this section. The switching topology is considered the same as Case 2. Noise is the same, as shown in Case 3. Also, the delay is considered the same as in Case 4.1. The consensus of the states is shown in Figures 36 and 37.

The consensus of the states is reached by the controls U_1 and U_2 which are shown in Figures 38 and 39, respectively.

The controller designed using NDI is capable of handling all the events such as switching topology, communication noise, and delay. The switching signal is shown in Figure 40.

The communication noise added to the two states of the agents is shown in Figure 41.

The communication issues mainly appear in the outputs of the agents. They do not add to the system dynamics. The DNDI contains the expression of the states of

neighbouring agents, and therefore, it can handle the communication issues.

Conclusion

The proposed distributed non-linear dynamic inversion controller (DNDI) is designed with a robust non-linear control theory dynamic inversion. This is for the first time dynamic inversion is applied to design the distributed consensus protocol among MASs. The agents' communication is subjected to practical issues like switching topology, communication delay, and noise. To make these issues more realistic, randomness is mixed with them. The results generated prove that the controller efficiently handles the communication issues among MASs. Moreover, the expression of consensus protocol comes in closed form, making it possible to implement in real-time. Therefore, DNDI is a potential candidate for designing consensus protocol. We consider applying DNDI in consensus problem to handle faults in actuators as our future work and compare the result with other existing controllers.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was partially funded by an Engineering and Physical Sciences Research Council (EPSRC) project CASCADE (EP/R009953/1).

ORCID iDs

Sabyasachi Mondal https://orcid.org/0000-0003-1109-5730
Antonios Tsourdos https://orcid.org/0000-0002-3966-7633

References

1. Wen G, Duan Z, Chen G, et al. Consensus tracking of multi-agent systems with lipschitz-type node dynamics and switching topologies. *IEEE Trans Circuits Syst Regular Pap* 2013; 61(2): 499–511.
2. Wen G, Yu Y, Peng Z, et al. Consensus tracking for second-order nonlinear multi-agent systems with switching topologies and a time-varying reference state. *Int J Control* 2016; 89(10): 2096–2106.
3. Kim JM, Park JB and Choi YH. Leaderless and leader-following consensus for heterogeneous multi-agent systems with random link failures. *IET Control Theor Appl* 2014; 8(1): 51–60.
4. Liu W, Zhou S, Qi Y, et al. Leaderless consensus of multi-agent systems with lipschitz nonlinear dynamics and switching topologies. *Neurocomputing* 2016; 173: 1322–1329.
5. Ding L, Han QL, Guo G, et al. Network-based consensus of nonlinear multi-agent systems with markovian switching topologies. In: IEEE International Conference on Industrial Technology (ICIT), Busan, Korean, 26 February–1 March 2014, pp. 828–833.
6. Xia H and Dong Q. Dynamic leader-following consensus for asynchronous sampled-data multi-agent systems under switching topology. *Inf Sci* 2020; 514: 499–511.
7. Li Y, Yan F and Liu W. Distributed consensus protocol for general third-order multi-agent systems with communication delay. In: Chinese Control and Decision Conference (CCDC), Nanchang, China, 3–5 June 2019, pp. 3436–3441.
8. Zhan J and Li X. Asynchronous consensus of multiple double-integrator agents with arbitrary sampling intervals and communication delays. *IEEE Trans Circuits Syst Regular Pap* 2015; 62(9): 2301–2311.
9. Xu X, Liu L and Feng G. Consensus of single integrator multi-agent systems with unbounded transmission delays. *J Syst Sci Complexity* 2019; 32(3): 778–788.
10. Cui B, Zhao C, Ma T, et al. Leader-following consensus of nonlinear multi-agent systems with switching topologies and unreliable communications. *Neural Comput Appl* 2016; 27(4): 909–915.
11. Jia Q, Han Z and Tang WKS. Synchronization of multi-agent systems with time-varying control and delayed communications. *IEEE Trans Circuits Syst Regular Pap* 2019; 66(11): 4429–4438.
12. Cheng L, Hou ZG and Tan M. A mean square consensus protocol for linear multi-agent systems with communication noises and fixed topologies. *IEEE Trans Automatic Control* 2013; 59(1): 261–267.
13. Wang Y, Cheng L, Hou Z-G, et al. Consensus seeking in a network of discrete-time linear agents with communication noises. *Int J Syst Sci* 2015; 46(10): 1874–1888.
14. Morita R, Wada T, Masubuchi I, et al. Multiagent consensus with noisy communication: Stopping rules based on network graphs. *IEEE Trans Control Netw Syst* 2015; 3(4): 358–365.
15. Liu J, Ming P and Li S. Consensus gain conditions of stochastic multi-agent system with communication noise. *Int J Control Automation Syst* 2016; 14(5): 1223–1230.
16. Wang Y, Cheng L, Ren W, et al. Seeking consensus in networks of linear agents: Communication noises and markovian switching topologies. *IEEE Trans Automatic Control* 2014; 60(5): 1374–1379.
17. Kar S and Moura JM. Distributed consensus algorithms in sensor networks with imperfect communication: link failures and channel noise. *IEEE Trans Signal Process* 2008; 57(1): 355–369.
18. Tariverdi A, Talebi HA and Shafiee M. Fault-tolerant consensus of nonlinear multi-agent systems with directed link failures, communication noise and actuator faults. *Int J Control* 2019: 1–15.
19. Li M, Deng F and Ren H. Scaled consensus of multi-agent systems with switching topologies and communication noises. *Nonlinear Anal Hybrid Syst* 2020; 36: 100839.
20. Li M and Deng F. Necessary and sufficient conditions for consensus of continuous-time multivalent systems with markovian switching topologies and communication noises. *IEEE Trans Cybernetics* 2019; 50(7): 3264–3270.
21. Ming P, Liu J, Tan S, et al. Consensus stabilization of stochastic multi-agent system with markovian switching topologies and stochastic communication noise. *J Franklin Inst* 2015; 352(9): 3684–3700.
22. Park MJ, Kwon OM, Park JH, et al. Randomly changing leader-following consensus control for markovian switching

multi-agent systems with interval time-varying delays. *Nonlinear Anal Hybrid Syst* 2014; 12: 117–131.
23. Pei Y and Sun J. Consensus of discrete-time linear multi-agent systems with markov switching topologies and time-delay. *Neurocomputing* 2015; 151: 776–781.
24. Ding L and Guo G. Sampled-data leader-following consensus for nonlinear multi-agent systems with markovian switching topologies and communication delay. *J Franklin Inst* 2015; 352(1): 369–383.
25. Shang Y. Consensus seeking over markovian switching networks with time-varying delays and uncertain topologies. *Appl Math Comput* 2016; 273: 1234–1245.
26. Jiang Y, Zhang Y and Wang S. Distributed adaptive consensus control for networked robotic manipulators with time-varying delays under directed switching topologies. *Peer-to-Peer Networking Appl* 2019; 12(6): 1705–1715.
27. Liu J, Zhang H, Liu X, et al. Distributed stochastic consensus of multi-agent systems with noisy and delayed measurements. *IET Control Theor Appl* 2013; 7(10): 1359–1369.
28. Zong X, Li T and Zhang J-F. Consensus conditions of continuous-time multi-agent systems with time-delays and measurement noises. *Automatica* 2019; 99: 412–419.
29. Ming P, Liu J, Tan S, et al. Consensus stabilization in stochastic multi-agent systems with markovian switching topology, noises and delay. *Neurocomputing* 2016; 200: 1–10.
30. Enns D, Bugajski D, Hendrick R, et al. Dynamic inversion: An evolving methodology for flight control design. *Int J Control* 1994; 59(1): 71–91.
31. Stein S and Padhi R. Automatic path planning and control design for autonomous landing of uavs using dynamic inversion. In: American Control Conference, St. Louis, MO, 10–12 June 2009, pp. 2409–2414.
32. Padhi R and Chawla C. Neuro-adaptive augmented dynamic inversion based pigc design for reactive obstacle avoidance of uavs. In: AIAA Guidance, Navigation, and Control Conference, Portland, OR, 8–11 August 2011, p. 6642.
33. Mondai S. and Padhi R.. Formation flying using genex and differential geometric guidance law. *IFAC-PapersOnLine* 2015; 48(9): 19–24.
34. Caverly RJ, Girard AR, Kolmanovsky IV, et al. Nonlinear dynamic inversion of a flexible aircraft. *IFAC-PapersOnLine* 2016; 49(17): 338–342.
35. Horn J. Non-linear dynamic inversion control design for rotorcraft. *Aerospace* 2019; 6(3): 38.
36. Lombaerts T, Kaneshige J, Schuet S, et al. Nonlinear dynamic inversion based attitude control for a hovering quad tiltrotor evtol vehicle. In: AIAA Scitech 2019 Forum, San Diego, CA, 7–11 January 2019.
37. Mondal S and Tsourdos A. Optimal topology for consensus using genetic algorithm. *Neurocomputing* 2020; 404: 41–49.
38. Padhi R, Rao PN, Goyal S, et al. Command tracking in high performance aircrafts: A new dynamic inversion design. *IFAC Proc Volumes* 2007; 40(7): 79–84.
39. Padhi R, Chawla C and Das PG. Partial integrated guidance and control of interceptors for high-speed ballistic targets. *J Guidance, Control Dyn* 2014; 37(1): 149–163.



https://dspace.org/

The consensus of non-linear agents under switching topology using dynamic inversion in the presence of communication noise and delay

Mondal, Sabyasachi; Tsourdos, Antonios

2021-05-03
Sage
https://doi.org/10.1177/09544100211012795
Mondal S, Tsourdos A. (2022) The consensus of non-linear agents under switching topology using dynamic inversion in the presence of communication noise and delay, Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, Volume 236, Issue 2, February 2022, pp. 352-367

Downloaded from a DSpace repository. https://dspace.org/