Discrete Time Consensus Algorithm In Multi-Agent System

Amol Patil and Gautam Shah

Abstract— **This paper explains mathematical framework of coordination phenomenon happening in nature and social dynamics. Distributed control law is designed for multi-agent dynamical systems using Perron-Frobenius theory. Algorithm convergence is analyzed for balanced and unbalanced communication graph. The consensus value and the topological condition under which the algorithm converges is also derived and explained. Simulation results shows the effective implementation of proposed algorithm**

Keywords : Multi-agent system (MAS), Consensus algorithm, Communication graph, Algebraic graph theory

I. INTRODUCTION

Distributed control of multiple autonomous agent is an active area of research in recent two decades [1]. Collective behavior occurring in nature is used to formulate mathematical framework of co-operative control of interconnected agents. Miniaturization of computing, communication, sensing, and actuation devices witnessed the large number of autonomous agents are interconnected to perform complex civilian and military missions. Such interconnected autonomous agents are referred to as multi-agent systems, where global objective is achieved via exchanging agents information locally. A flock of birds, a school of fish or a swarm of insects are the collective behavior happening in nature and it has an advantages of seeking food, migrating from one place to another and avoiding predators or obstacles. [2]. This collective behavior was first mathematically formulated and simulated by Reynolds on a computer [3]. He proposes three simple rules Separation (Avoid crowding neighbors), Alignment(Steer towards average heading of neighbors) and Cohesion (Steer towards average position of neighbors). Several distributed optimization algorithms for the multi-agent system were proposed by the researchers [4], [5]. Algorithms proposed in [4], [5] are designed using distributed approach. The theory of co-ordination and co-operative control of multi-agent systems also used many space-based applications. In satellite formation flying multiple smaller satellites are used to accomplish the large and complex task instead of using single expensive satellite. Formation control of multiple satellites has an advantage of better scalability and cost reduction [6]. Formation control of satellites is refereed as attitude synchronization problem in multi-agent systems.

Amol Patil is with Phd student of Electronics and Telecommunication Engineering, St. Francis Institute of Technology, University of Mumbai, Borivali Mumbai-400 103 agpatil@acpce.ac.in

Gautam Shah is with faculty of Electronics and Telecommunication Engineering, St. Francis Institute of Technology, University of Mumbai, Borivali Mumbai-400 103 gautamshah@sfit.ac.in

Another example of cooperative control problems in multi-agent systems is clock synchronization [7] and deployment of wireless sensor network in unknown environment [8]. In cooperative control of multi-agent systems consensus is a fundamental problem. Consensus problem was traced back in computer science community and form the foundation of distributed computing. Initial work on the consensus problem for distributed decision making and parallel computing was carried out by Borkar and Varaiya [9]. In distributed computing, the agreement protocol is developed among the static agents i.e. software agents. This laid the foundation to design agreement protocols for dynamic agents [10] and [11]. In particular, a theoretical explanation was provided in [10] for the alignment behavior observed in the Vicsek model [12] and a general framework of the consensus problem for networks of integrator was proposed in [11]. In [13] a sufficient condition was derived to achieve consensus for first-order integrator multi-agent systems with jointly connected communication graphs. Many of the researchers came up with monographs on co-operative control for multi-agent systems [14], [15], [16], [17]. Formation control in multi-agent system is the extension of consensus algorithm where each agent maintain a specific geometric shape while performing the complex task. In formation control more than two agents required in order to maintain specific geometric structure [18] After 2013, researchers started testing and validating the consensus algorithms on robotics platforms. In [19] the authors simulated and tested the formation control and trajectory tracking of nonholonomic mobile robots. The continuous time consensus algorithm in multi-agent system was proposed in [20]. This paper proposes the design of distributed control law for multi-agent system where information exchange among the multiple agents is in discrete time domain. Distributed control law is designed using Perron-Frobenius theory.

Notations and Definitions: The notations used in this paper are standard: Capital bold face letters are used for representing the matrix and the elements of a matrix are represented by small case letters. $\mathbb{R}^{n\times n}$ denote the set of $n\times n$ real matrices. G_N represents the communication graph of dimension N. λ, $\underline{1}$ and w_1 denotes the eigenvalues, right eigen vectors and left eigen vectors of Frobenius matrix **F** respectively. Matrices **F**, **D** and **A** denote Frobenius matrix, diagonal matrix and connectivity matrix respectively

The organization of paper is as follows. Section II presents preliminaries on algebraic graph theory and graph matrices. Section III describes problem formulation. Simulation result is explained in section IV and conclusion and future scope is stated in section V

II. PRELIMINARIES

This section explains algebraic graph theory a mathematical tool require for construction of communication graph for the multi-agent system (MAS). The behaviors and connections of the MAS are possible through edges of a communication graph, hence covering graph theory becomes essential. Information between agents is either unidirectional or bidirectional. Graph matrices associated with communication graph is introduced later in section III where focus is shifted to connectivity and Laplacian matrices which are commonly used in the MAS.

Communication graph is defined as a set of agents and a edge set of unordered pair of nodes connected over the graph. Mathematically a communication graph is represented as $G_N = (V, E)$, where $V = \{v_1, v_2, \ldots, v_N\}$ represents set of agents, and E denoted as (v_i, v_j) signifies the connection between the agents starting from i and terminating at j [15]. The in-degree of a node denotes the number of edges that have v_i as their head, and the out-degree of a node means how many edges has v_i as their tail. The in-degree of a agent in graph equals to the number of neighbors of that node. Knowing the number of neighbors of node i, i.e. $N_i = v_j : (v_i, v_j) \in E$ is crucial in consensus algorithm. The information sharing between the each agents for underlying communication graph is distributed in the sense that each agent communicates with its neighbors locally so as to achieve the global objective. Total number of agents present in a communication graph are N. Information exchange among agents is modeled by directed and undirected graph. The edge weights in a graph are strictly positive. For a graph if there are directed path from v_i to v_j and from v_j to v_i then the graph is bidirectional or undirected graph. Every Undirected graph is balanced graph. For a graph if there is directed path from v_i to v_j and not from v_j to v_i then the graph is directed graph, if such graph does not maintain degree at each agent is known as unbalanced graph. For various other types of graph refer [17]

Communication graph has at least one directed spanning tree in which all the agents of the graph and are reachable from root agent by following the edge arrows. A graph may have multiple spanning trees but existence of one spanning tree is enough for achieving the consensus. If the graph is strongly connected then all the nodes in the graph are root nodes. More about algebraic graph theory is given in [15] and [17]

Communication graph properties can be studied by examining the properties of certain matrices associated with the graph. For the given communication graph the degree of each node is denoted by $d(v_i)$ and it equal to number of incoming branches at node v_i. Diagonal matrix denoted by $\mathbf{D}$ with diagonal elements equal to the degree of each node [17]. Connectivity matrix $\mathbf{A}$ have special importance in a graph since it consists of information regarding the agents and their interconnections [13]. The adjacency matrix $\mathbf{A}$ of a graph is square matrix defined as

$$\mathbf{A} = \begin{cases} 1 & (v_i v_j) \in E \\ 0 & \text{otherwise} \end{cases}$$

For balanced graph the matrix $\mathbf{A}$ is symmetric and for unbalanced graph it is non symmetric. Graph Laplacian matrix $\mathbf{L}$ play a important role for achieving consensus in multi-agent system and is defined by $\mathbf{L} = \mathbf{D} - \mathbf{A}$.

Many properties of a graph may be studied in terms of the matrix $\mathbf{L}$. Here it is vital to denote some of the important characteristics of the matrix $\mathbf{L}$ for the given graph that are applicable to the multi-agent system obtained from [17].

- $\mathbf{L}$ is symmetric (respectively non symmetric) if G_N is balanced (respectively if G_N is unbalanced)
- $\mathbf{L}$ is positive semi-definite (respectively positive definite) if G_N is balanced (repectively if G_N is unbalanced)
- $\mathbf{L}$ consist of real eigen values (respectively complex eigen values) if G_N is balanced (repectively if G_N is unbalanced)
- Row sum of $\mathbf{L}$ for each row is equal to zero

Eigen structure of the matrix $\mathbf{L}$ plays the crucial role for analyzing the convergence of the consensus algorithm at the steady state. One of the eigen value of the matrix $\mathbf{L}$ is simple and located at origin then system is said to be critically stable

III. PROBLEM FORMULATION

This section explains the mathematical formulation of discrete time consensus algorithm. For deriving the discrete time consensus equation consider the network topology as shown in Fig. 1

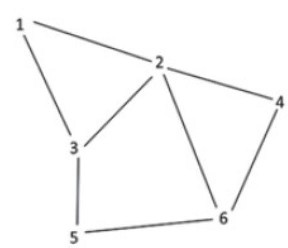

Fig. 1. Communication Graph

Each agent in Fig. 1 has associated with following discrete time state space equation

$$x_i(k+1) = x_i(k) + \mu_i(k) \tag{1}$$

with k is present time instance and $k+1$ is the future time instance. x_i is the state information of agent i and μ_i is the control input of agent i and $x_i, \mu_i \in \mathbb{R}$. The discrete time consensus equation is derived using Perron discrete time system in section III-A

A. Normalized Control Protocol For Discrete-Time Consensus

Consider the following normalized control input for each agent

$$\mu_i(k) = \frac{1}{1+d_i} \sum_{j \in N_i} a_{ij} \left[x_j(k) - x_i(k)\right] \tag{2}$$

where a_{ij} are elements of matrix **A** and d_i is the in-degree of agent i. Substituting equation (2) in equation (1), we get

$$x_i(k+1) = x_i(k) + \frac{1}{1+d_i} \sum_{j \in N_i} a_{ij} \left[x_j(k) - x_i(k)\right] \tag{3}$$

$$\begin{aligned} x_i(k+1) = x_i(k) + \frac{1}{1+d_i} \Bigg(& - x_i(k) \sum_{j \in N_i} a_{ij} \\ & + \sum_{j \in N_i} a_{ij} x_j(k) \Bigg) \end{aligned} \tag{4}$$

But $\sum_{j \in N_i} a_{ij} = d_i$ and if $j = 1....N$ then equation (4) reduces to

$$\begin{aligned} x_i(k+1) = x_i(k) + \frac{1}{1+d_i} \Bigg(& - x_i(k) d_i \\ & + \begin{bmatrix} a_{i1} & \ldots & a_{iN} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \Bigg) \end{aligned} \tag{5}$$

Matrix form representation of equation (5) is given by

$$x(k+1) = \mathbf{I}x(k) + \frac{\mathbf{I}}{\mathrm{I}+\mathbf{D}} \Bigg(- x(k)\mathbf{D} + \mathbf{A}x(k) \Bigg) \tag{6}$$

Here matrix **I** is identity matrix equivalent to 1 in scalar, taking $x(k)$ common equation (6) reduced to

$$x(k+1) = \left[\mathbf{I} + \frac{\mathbf{I}}{\mathrm{I}+\mathbf{D}} \left[- \big(\mathbf{D} - \mathbf{A}\big) \right] \right] x(k) \tag{7}$$

simplifying equation (7) finally we get discrete time consensus equation as

$$x(k+1) = (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})x(k) \equiv \mathbf{F}x(k) \tag{8}$$

with $x = [x_1......x_N]^T \in \mathbb{R}^N$. The matrix **F** is referred as the Frobenius matrix. Laplacian matrix **L =D-A** has one eigenvalue at origin of complex s-plane and the remaining eigenvalues in the right half of the complex s-plane, makes system unstable. To make it stable poles of the system brought to the left of complex s-plane by assigning negative sign to matrix **L= D-A** refer equation (7) and according to Gershgorin circle criterion in the normalized Laplacian matrix, $(\mathbf{I+D})^{-1}(\mathbf{D\text{-}A})$, all eigen values are found on the right-hand side of the complex z-plane within a disk centered at $\frac{d_{max}}{1+d_{max}}$ with the same size radius [15]. The shaded region of the z-plane is shown in Fig. 2 where all eigenvalue reside. If $\frac{d_{max}}{1+d_{max}} < 1$ then this shaded region is always inside the unit circle. Thus **F** has a simple eigenvalue of $\lambda_1 = 1$ and the remaining eigen values are strictly inside the unit circle. This result make equation (8) is marginally stable and of Type 1, and state approaches a steady-state value is reached [15].

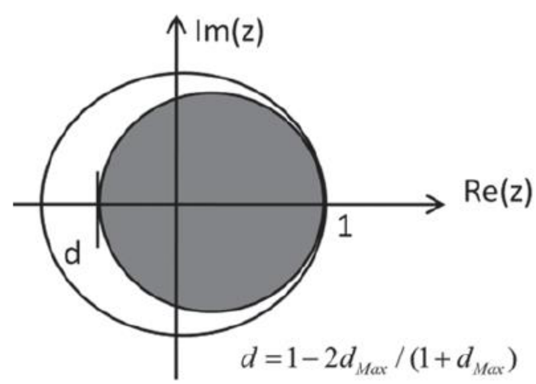

Fig. 2. Region of eigenvalues of Perron matrix P

Since Laplacian matrix **L** has row sums of zero, then matrix **F** has row sums of 1, and so is a row stochastic matrix. That is

$$\mathbf{F}\underline{1} = \underline{1} \tag{9}$$

and $\underline{1}$ is the right eigenvector for the eigen value $\lambda_1 = 1$. Let w_1 be a left eigenvector of matrix **L** for $\lambda_1 = 0$. Then w_1 is also the left eigenvector of matrix **F** for $\lambda_1 = 1$

B. STEADY STATE ANALYSIS

This section explains the convergence of discrete time consensus algorithm at steady state. If the system (8) reaches to steady state, then

$$x_{ss} = \mathbf{F} x_{ss} \tag{10}$$

If the given network topology has a directed spanning tree then only possible solution is $x_{ss} = c\underline{1}$ for some $c > 0$. Then the consensus achieved such that $x_i = x_j = c \;\forall\; i, j$. Let $w_1 = [p_1, ..., p_N]^T$ be a left eigenvector of $\mathbf{F}$ for $\lambda_1 = 1$, then

$$w_1^T x(k+1) = w_1^T \mathbf{F} x(k) = w_1^T x(k) \tag{11}$$

So that the quantity

$$\bar{x} = w_1^T x = [p_1, ..., p_N]^T \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_N \end{bmatrix} = \sum_i p_i x_i \tag{12}$$

is invariant. This quantity $\bar{x} = \sum_i p_i x_i$ is the constant of motion. So $\sum_i p_i x_i(0) = \sum_i p_i x_i(k) \;\forall\; k$. Therefore if the network topology has a spanning tree, at steady state one can reached the consensus so that $x_i = x_j = c \;\forall\; i, j$ where the consensus value is given by

$$c = \frac{\sum_i p_i x_i(0)}{\sum_i p_i} \tag{13}$$

If the graph is balanced, then row sums of $\mathbf{L}$ are equal to column sums, and this property carries over to $\mathbf{F}$. Then w_1 is the left eigenvector of $\mathbf{F}$ for $\lambda_1 = 1$ and consensus is the average of the initial state information of each agent refer [21]

IV. SIMULATION RESULT

For implementing discrete time consensus algorithm considered the unbalanced graph and balanced graph as shown in Fig. 3 and Fig. 4

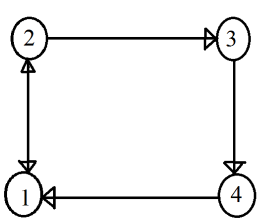

Fig. 3. Unbalanced Graph

consider all edge weight is equal to 1 for mathematical sophistication. The Frobenius matrix $\mathbf{F}$ for unbalanced graph shown in Fig. 3 is given by

$$\mathbf{F} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{14}$$

and Frobenius matrix $\mathbf{F}$ for balanced graph shown in Fig. 4 is given by

$$\mathbf{F} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \end{bmatrix} \tag{15}$$

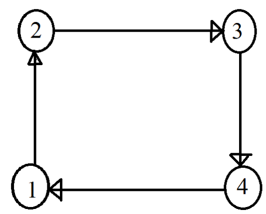

Fig. 4. Balanced Graph

Set initial state information of agents as their numbers i.e. $[1, 2, 3, 4]$. Fig. 5 and Fig. 6 shows convergence of consensus algorithm for unbalanced and balanced graph respectively

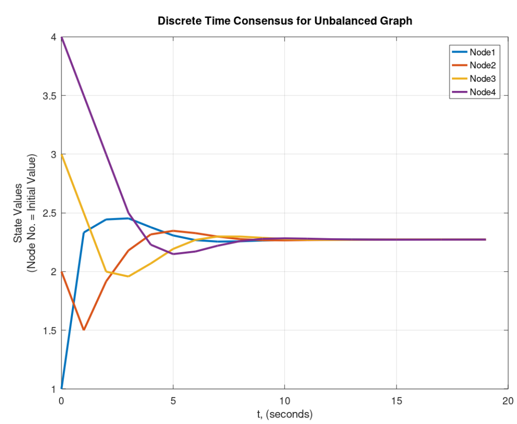

Fig. 5. Algorithm Convergence For Unbalanced Graph

Eigen values for the balanced graph are $\lambda_1 = 0$ $\lambda_{2,3} = 0.5 \pm 0.5i$ and $\lambda_4 = 1$ and for the unbalanced graph are $\lambda_1 = -0.11578$ $\lambda_{2,3} = 0.47455 \pm 0.36699i$ and $\lambda_4 = 1$. All the eigenvalues are inside the Gershgorin unit circle expect $\lambda_4 = 1$ which makes the system marginally stable. However the states approach steady state value and reach consensus using equation (13)

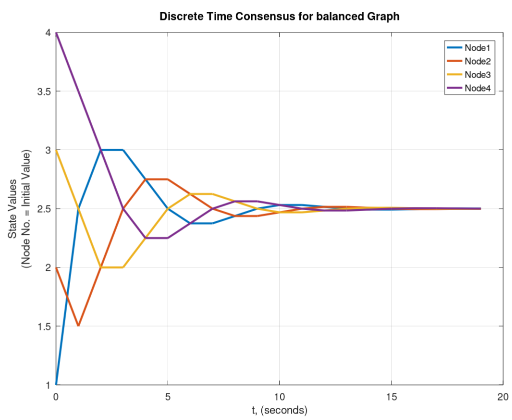

Fig. 6. Algorithm Convergence For Balanced Graph

V. CONCLUSION AND FUTURE SCOPE

A. Conclusion

Discrete time consensus algorithm is implemented for balanced and unbalanced graph. For unbalanced graph the initial state information of each agent converges to $y = 2.2$ in 13 seconds shown in Fig. 5 and for balanced graph the initial state information of each agent converges to average of their initial state information in 15 second. Communication between agents is in discrete time domain and state information of each agent is updated for every 1 second. Cardinality of balanced graph is 1 so equal amount of control effort is required for each agent to reach the consensus. But cardinality for unbalanced graph for agent 1 is 2 so less amount of control effort is required for agent 1 to reach the consensus refer Fig. 5. For balanced graph algorithm converges to average of initial state information of all agents hence called as average consensus algorithm. Hence it is concluded that average consensus achieves if communication graph is balanced as well as d-regular and there exist at least one minimum spanning tree in a graph

B. Future Work

Further open research issues in above derived algorithm is

- Effect of time delay in consensus algorithm
- Design of consensus algorithm using double integrator dynamics
- Formation control using DT consensus algorithm

REFERENCES

[1] Yongcan Cao, Wenwu Yu, Wei Ren and Guanrong Chen, "An Overview of Recent Progress in the Study of Distributed Multi-agent Coordination", *IEEE transaction on automatic Control*, 2012

[2] A Okubo, "Dynamical aspects of animal grouping: Swarms, schools, flocks and herds",*Advances in Biophysics*, 22:1-94, 1986

[3] C. W. Reynolds Flocks, herds and schools: A distributed behavioral model. Computer Graphics, 21(4):25-34, 1987

[4] R. Olfati-Saber,"Flocking for multi-agent dynamic system: Algorithms and theory", *IEEE Transactions on Automatic Control*, 51(3):401-420, 2006

[5] H. G. Tanner, A. Jadbabaie and G. J. Pappas," Flocking in fixed and switching networks",*IEEE Transactions on Automatic Control*, 52(5):863-868, 2007

[6] M. Tillerson, G. Inalhan and J. P. How, " Co-ordination and control of distributed spacecraft systems using convex optimization techniques",*International Journal of Robust and Nonlinear Control* ,12(2-3):207-242, 2002

[7] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: A Survey",*IEEE Network*, 18(4):45-50, 2004

[8] A. Howard, M J Mataric and G S Sukhatne,"Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem", *Proceeding of the 6th International Symposium on Distributed Autonomous Robotics Systems* pp. 299-308, 2002

[9] V. Borkar and P. Varaiya, "Asymptotic agreement in distributed estimation", *IEEE Transactions on Automatic Control*, 27(3):650–655, 1982.

[10] A. Jadbabaie, J. Lin, and A.S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules",*IEEE Trans. on Automatic Control*, 48(6):988–1001, 2003.

[11] R. Olfati-Saber and R.M. Murray, "Consensus problems in networks of agents with switching topology and time-delays", *IEEE Trans. on Automatic Control*, 49(9):1520–1533, 2004.

[12] T. Vicsek and et. al, "Novel type of phase transition in a system of self-driven particles", *Physical Review Letters*, 75(6):1226–1229, 1995.

[13] W. Ren and R.W. Beard, "Consensus seeking in multi-agent systems under dynamically changing interaction topologies",*IEEE Transactions on Automatic Control*, 50(5):655–661, 2005.

[14] Lin Z, Francis B, Maggiore M,*Distributed Control and Analysis of Coupled Cell Systems*, VDM Verlag: Germany, 2008

[15] Ren W, Beard R,*Distributed Consensus in Multi-vehicle Cooperative Control*, Spring: New York, 2008

[16] Bullo F, Cortes J, Martinez S *Distributed Control of Robotic Networks:A Mathematical Approach to Motion coordination Algorithms*, Princeton University Press: Princeton and Oxford, 2009

[17] Mesbahi M, Egerstedt M. *Graph Theoretic Methods in Multi-agent Networks*, Princeton University Press: Princeton and Oxford, 2010

[18] Pushpak Jagtap and er al., "Complex Laplacian based Algorithm for Output Synchronization of Multi-agent Systems using Internal Model Principle", *IEE conference on Multisystem Conference*, Nov-15

[19] Akshit and et al., "Formation Control and Trajectory Tracking of Nonholonomic Mobile robots", *IEEE transaction on Control System Technology*, Nov-18

[20] Amol Patil and Gautam Shah, "Multi-Robot Trajectory Tracking and Rendezvous Algorithm", *IETE Journal of research* , August-2020

[21] Frank Lewis and et al.,*Cooperative Control of Multi-Agent systems*, Springer-Verilog: London, 2014