Multi Robot Trajectory Tracking and Rendezvous Algorithm

Amol Patil, Gautam Shah

Amol Patil is the PhD student of Electronics and Telecommunication Engineering,
St. Francis Institute of Technology, Borivali Mumbai, 400 103 (e-mail: agpatil@acpce.ac.in).
Gautam shah is with the St. Francis Institute of Technology, Borivali Mumbai, 400103 (e-mail: gautamshah@sfit.ac.in).

ABSTRACT

This paper explains mathematical framework of physical phenomenon happening in nature and social opinion dynamics. It is divided into two parts. The first part deals with design of the trajectory tracking algorithm for multiple mobile robots and in the second part rendezvous algorithm is described. Trajectory tracking algorithm is designed using leader-follower approach in which each follower robot tracks the reference trajectory generated by the leader. Rendezvous algorithm is designed by deriving consensus equation using algebraic graph theory. Condition for achieving rendezvous point is also derived. Simulation shows the effectiveness of proposed algorithm

Keywords:
Consensus, Co-operative control, Communication graph, Multi-agent sytems, Trajectory tracking algorithm, Rendezevous algorithm.

1. INTRODUCTION

Co-ordination and control between the multi-agents is an active area of research in recent years [1]. Due to continuous progress and development in computing, communication, sensing, and actuation devices, it is possible for large number of autonomous agents to interconnect with each other. These interconnected agents used to perform civilian and military missions. There is an interesting phenomenon of collective behavior happening among the birds, fishes and insects and it has an advantages of seeking food, migrating from one place to another and avoiding predators or obstacles [2]. Collective behavior occurring in nature is used to formulate mathematical framework of co-operative control among the multiple agents. The interconnected engineering systems are also used for industrial and civilian application. This collective behavior was first mathematically modeled and simulated on by Reynolds on a computer [3]. He proposes three simple rules separation, alignment and cohesion. Several coordination and optimization algorithms among the multiple agents have been developed in the literature [4, 5] which was theoretically proved by Reynolds [3]. Algorithms proposed in [4, 5] are designed using distributed approach. The theory of co-operative control of multi-agent systems also used many industrial applications. In satellite formation flying multiple smaller satellites are used to accomplish the large and complex task instead of using single expensive satellite. Formation flying of multiple satellites has an advantage of better scalability and cost reduction [6]. Formation flying of satellites is refereed as attitude synchronization problem in multi-agent systems. Clock synchronization [7] and deployment of wireless sensor network in unknown environment [8] are well studied problems

in co-operative control of network systems. In cooperative control of multi-agent systems consensus is a fundamental problem. Consensus problem was originated in computer science and form the basis of distributed computing. Consensus problem for distributed decision making and parallel computing was first handled by Borkar and Varaiya [9]. In distributed computing, the consensus problem is handled by static agent i.e. software agent. This laid the foundation to design consensus protocols for dynamic agents [10] and [11]. In [10] a theoretical framework of consensus problem and in [11] the alignment behavior is observed using the Vicsek model [12]. In [13] jointly communication graphs are used to derive a sufficient condition for achieving consensus using first-order integrator dynamics. Many of the researchers came up with monographs on co-operative control for multi-agent systems [14, 15, 16, 17]. After 2013, researchers started testing and validating the consensus algorithms on robotics platforms. In [18] the authors simulated and tested the formation control and trajectory tracking of nonholonomic mobile robots. In this paper we first designing consensus equation using algebraic graph theory and later it is validated on tracking and rendezvous application among multiple agents

The organization of paper is as follows. Section 2 presents some preliminaries on algebraic graph theory. Section 3 describes trajectory tracking algorithm. Section 4 gives a design of Rendezvous algorithm. Section 5 provides simulation results for both algorithms and section 6 presents the conclusion and future scope

2. GRAPH THEORY

Consensus is the fundamental control problem in study of cooperative control of multi-agent system. Consensus is nothing but the common agreement among the multiple agents on certain quantity of interest. This is possible if and only if all the agents should connect with each other in prescribed sense. These connection among the agents is known as communication graph G_N. The consensus protocol developed is distributed such that each agent in a group interacts with itself and its neighbor locally by sharing the information over the communication network to achieve the global behavior. The mathematical tools required to design such consensus protocol is called as algebraic graph theory and graph matrices associated with it. For relevant definition and associated graph matrices refer monograph [15] and [16]

Laplacian Matrix L : Eigen values of the Laplacian matrix **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 **L** is simple and located at origin then system is said to be critically stable. For construction of Laplacian matrix **L** from underlying communication graph G_N refer [17]

3. TRAJECTORY TRACKING ALGORITHM

This section explains design of trajectory tracking algorithm among the multi-agent using leader follower approach. The reference trajectory is generated by the leader and the followers follow this reference trajectory. The condition under which trajectory tracking algorithm converges to constant reference state is also being verified.

3.1 Problem Statement

Consider the communication graph G_N shown in Fig. 1 where root node is leader and remaining nodes are followers. Consider the case where all agents in the communication graph G_N have scalar input as

$$\dot{x_i} = \mu_i \tag{1}$$

with x_i, $\mu_i \in R$

Consider local control input for each agent i

$$\mu_i = \sum_{j \in N_i} a_{ij}(x_j - x_i) \tag{2}$$

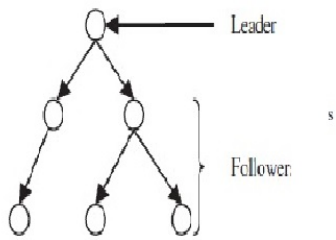

Figure 1: Tree Graph For Trajectory Tracking

Here a_{ij} is a graph edge weight. Equation (2) solves the trajectory tracking problem and determines the values at which trajectory tracking algorithm converged. Substituting equation (2) in equation (1) we get the following close loop equation

$$\dot{x_i} = \sum_{j \in N_i} a_{ij}(x_j - x_i) \tag{3}$$

$$\dot{x_i} = -x_i \sum_{j \in N_i} a_{ij} + \sum_{j \in N_i} a_{ij} x_j \tag{4}$$

$$= -d_i x_i + [a_{i1} ..... a_{iN}] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \tag{5}$$

Defining global state vector $\left[x_1 \dots x_N\right]^T = x$ and diagonal matrix $D = diag\left\{d_i\right\}$ the close loop equation modified as

$$\dot{x} = -Dx + Ax = -(D - A)x \tag{6}$$

$$\dot{x} = -Lx \tag{7}$$

The global input vector $\mu = \left[\mu_1 \dots \mu_N\right]^T$ is given by

$$\mu = -Lx \tag{8}$$

It is seen that local control input (2), the close loop equation (7) depends on Laplacian matrix **L**. The steady state solution of (7) is

$$0 = -Lx_{ss} \tag{9}$$

The steady state information x_{ss} is a null-space of L. At steady state

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = c1 = \begin{bmatrix} c \\ c \\ \vdots \\ c \end{bmatrix} \tag{10}$$

when $x_i = x_j = c$ $\forall$ i, j, the consensus is reached. The consensus value c is determined by

$$c = \sum_{i=1}^{N} p_i x_i(0) \tag{11}$$

where p_i are normalized left eigen vectors of Laplacian matrix L for the eigen value $\lambda_1 = 0$

3.2 Trajectory tracking Modeling

The local control input (2) can be used in common headings of animals in a groups. Simple motion of a group of N agents in (x,y) plane is as shown in Fig 2. Referring to Fig 2, each node dynamics is described by

$$\begin{aligned} \dot{x_i} &= V cos\theta_i \\ \dot{y_i} &= V sin\theta_i \end{aligned} \tag{12}$$

Equation (12) refers heading equation for all agents

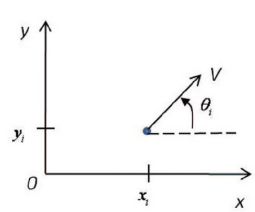

Figure 2: Motions of agents in x-y plane

shown in Fig 1. In equation (12) v is speed of the agents assumed to be same and θ_i is the heading of agent i similar to local control input (2) and it is given by

$$\dot{\theta}_i = \sum_{j \in N(i)} a_{ij}(\theta_j - \theta_i) \tag{13}$$

4. Rendezvous Algorithm

This section explains the design of rendezvous algorithm for two robots and multiple robots

4.1 Two Robot Case

Consider two robots located on a straight line as shown in Fig 3. Both the robots are moving on a flat 2 dimension plane.

Figure 3: Undirected graph

Starting from initial position at t=0, the position of robots at the consecutive times t is given by

$$x(t) = vt + x(0) \tag{14}$$

Equation (14) gives the position at each time t. The position of each robot at particular time is given by $t_{i+1} = t_i + \Delta t$, $i = 1, 2, 3...$ where $t_0 = 0$ and Δ t is the sampling time. The position $x_{i+1} = x(t_{i+1})$ for each robot at each t_i is given by

$$x_{i+1} = x(t_{i+1}) = x_i + v\Delta t, \quad i = 0, 1, 2... \tag{15}$$

Equation (15) actually solves the rendezvous problem in two robot case. There are three ways in which the robots can move

1. Robot 1 is static and robot 2 is moving
2. Robot 2 is static and robot 1 is moving
3. Both robots are moving towards each other

It is easy to implement the rendezvous algorithm for two robot case by solving simple linear equation (15). Finding the rendezvous point for multirobot case is a challenging task.

4.2 Multi Robot Case

Consider multirobot robots connected to each other as shown in communication graph Fig 4. The graph G is balanced and fully connected. The Laplacian matrix L for the communication graph G is given by

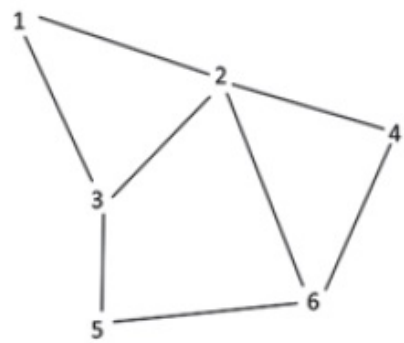

Figure 4: Graph For Rendezvous Algorithm

$$L = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 4 & -1 & -1 & 0 & -1 \\ -1 & -1 & 3 & 0 & -1 & 0 \\ 0 & -1 & 0 & 2 & 0 & -1 \\ 0 & 0 & -1 & 0 & 2 & -1 \\ 0 & -1 & 0 & -1 & -1 & 3 \end{bmatrix} \tag{16}$$

The row sum of Laplacian matrix **L** shown in equation (16) is zero, hence one of the eigen value is simple and located at origin of S plane, remaining 4 eigen values are located on positive real axis of S plane. Eigen values of the equation (16) are positive real,

hence consensus equation become unstable. For stable consensus equation the Laplacian matrix $\mathbf{L}$ is beging inverted which is shown in equation (7). The rendezvous point for multirobot case is called as centroid of all robots and the centroid is the average of initial state information for all robots. The centroid is given by equation (11). The equation (11) is evaluated by solving equation (7) .

5. SIMULATION RESULTS

5.1 Trajectory Tracking Simulation

Trajectory tracking simulation is generated by referring Fig 1. Fig 5 shows reference trajectory is generated by the leader and hence all the followers converges to leaders reference trajectory. Initial information for leader trajectory the followers trajectories is chosen to be random. Speed of all robots considered to be constant. All the trajectories are converged to the point $(6, 62)$

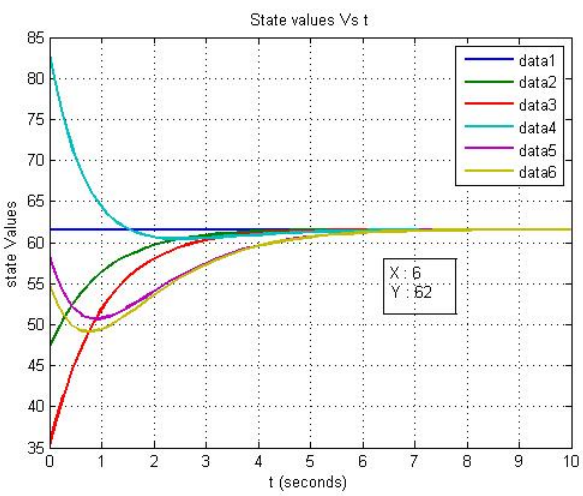

Figure 5: Leader Follower Convergence

5.2 Rendezvous Algorithm

Rendezvous algorithm for two robot is simulated using the equation (15). Let the sampling time be $\Delta t = 0.05sec$. Each robot has a sensor onboard that allows it to measure its relative distance from the other robot, that is robot 2 can measure $x_1(t_i) - x_2(t_i)$ and robot 1 can measure $x_2(t_i) - x_1(t_i)$ Consider the case that robot 1 is stationary and robot 2 is wants to meet up with robot 1 that is starting from $x_2(0) = 5$ robot 2 wants to reach to $x_2(t_i) = x_1(0) = 0$ at some $t_i > 0$ and stop there. Relative distance between them is given by $x_2(t_i) - x_1(t_i)$. Fig 6 shows movement of movement of robot 2 towards robot 1 Similarly Fig 7 shows movement of movement of robot 1 towards robot 2

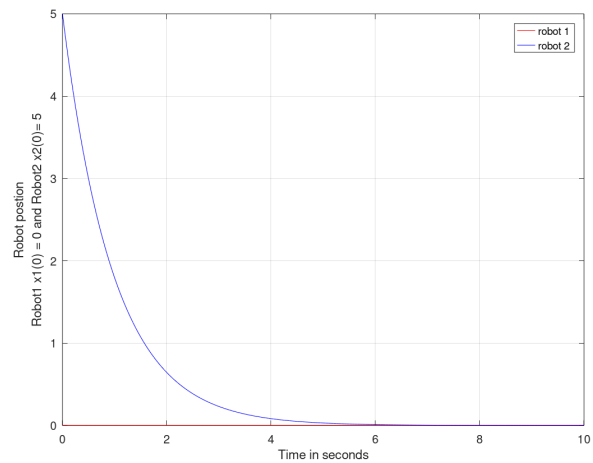

Figure 6: Robot 2 Movement

If both robots move towards each other then they should meet at average of their initial positions. Fig 8 shows movement of robot 1 and robot 2 towards each other and they are meeting at average of their initial position that is 2.5

When more than two robots are connected over communication graph G_N then it is challenging task to find the convergence point of all robot. In fact all robot are converges to the average of their initial state information. Fig 9 shows the convergence of rendezvous algorithm explained in section 4.2. The initial state information of all robots at $t = 0$ are assumed as $\{1, 2, 3, 4, 5, 6\}$. The rendezvous point is 3.5 average of robots initial state information.

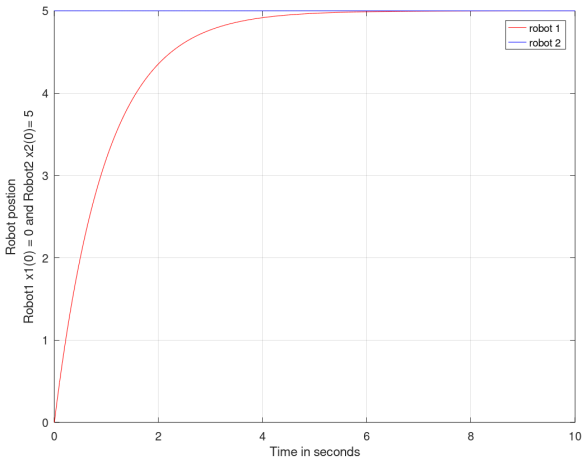

Figure 7: Robot 1 Movement

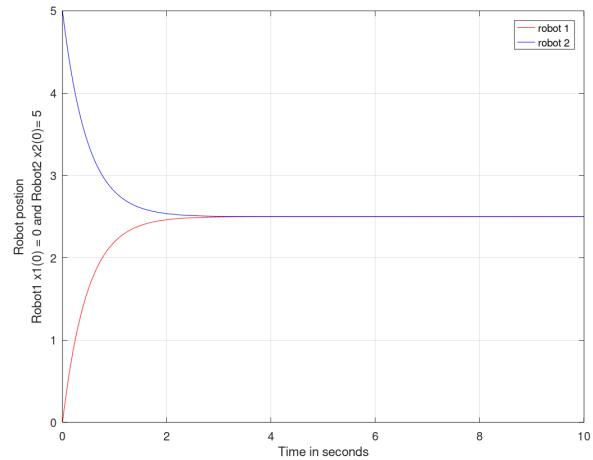

Figure 8: Robot 1 and Robot 2 Movement

6. Conclusions

Section 5.1 explains trajectory tracking algorithm among the multiple agents. It is also called as leader follower consensus algorithm. The trajectory tracking algorithm can be viewed as mathematical framework of animal flocking in physical world where all followers nicely follows the motion of leader to migrate from one place to other place. Tree communication graph as shown in Fig 1 is consider for deriving and convergence of trajectory tracking algorithm. Initial state information for leader and follower is randomly generated by the algorithm. Refer Figure 5

Section 5.2 describes rendezvous algorithm. Rendezvous means a common meeting point of all agents.The information shared among the agents is distributed in the sense that each agent will share its information locally to its neighboring agent only. No each agent knows the global objective but they can achieve global objective via local interaction with each other. So rendezvous algorithm is also called as distributed control of multi-agent system. Rendezvous algorithm first tested on two robot case and they reach to average of their initial state information position see Figure 8.

For more than two robot case rendezvous algorithm we consider the communication graph as shown in Fig 4. For more than two robot case algorithm converges to average of robot initial state information. When rendezvous algorithm converges to average of robot initial state information then it is also called as average consensus algorithm. Condition for achieving average consensus algorithm is that given communication graph must be balanced

6.1 Future Work

Further open research issues in above derived algorithms are

- Design of leaderless trajectory tracking algorithm
- Effect of time delay in information sharing among the agents in renzevous algorithm
- What could be algorithm convergence if communication graph is not balanced ?
- Each agent endowed with continuous time dynamics what will happen if each agent associated with discrete time dynamics?

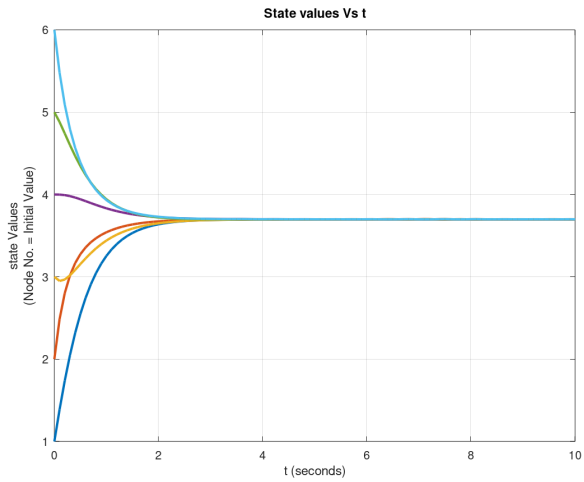

Figure 9: Rendezvous Plot for Multiple Robot

ACKNOWLEDGMENT

The authors would like to thank **Prof. Navdeep Singh** from VJTI for continuous support and encouragement for this research article

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 Trans. on automatic Control, 2012.

[2] A Okubo, "*Dynamical aspects of animal grouping: Swarms, schools, flocks and herds*," Advances in Biophysics, Vol. 22, pp. 1_4, 1986.

[3] C. W. Reynolds, "*Flocks, herds and schools: A distributed behavioral model* ," Computer Graphics, Vol. 21, no.4, pp. 25_34, 1987.

[4] R. Olfati-Saber, "*Flocking for multi-agent dynamic system: Algorithms and theory*," IEEE Trans. on Automatic Control, Vol. 51, no. 3, pp. 401_420, 2006.

[5] H. G. Tanner, A. Jadbabaie and G. J. Pappas, "*Flocking in fixed and switching networks*," IEEE Trans. on Automatic Control, Vol. 52, no. 5, pp. 863_868, 2007.

[6] M. Tillerson, G. Inalhan and J. P. How, "*Coordination and control of distributed spacecraft systems using convex optimization techniques*," International Journal of Robust and Nonlinear Control, Vol. 12, pp. 207_242, 2002.

[7] F. Sivrikaya and B. Yener, "*Time synchronization in sensor networks: A Survey*," IEEE Trans. on Automatic Control, Vol. 18, no. 4, pp. 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, Vol. 27, no. 3, pp. 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, Vol. 48, no. 6, pp. 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, Vol. 49, no. 9, pp. 1520_1533, 2004.

[12] T. Vicsek and et. al, "*Novel type of phase transition in a system of self-driven particles* ," Physical Review Letters, Vol. 75, no. 6, pp. 1226_1229, 1995.

[13] W. Ren and R.W. Beard, "*Consensus seeking in multi-agent systems under dynamically changing interaction topologies*", IEEE Trans. on Automatic Control, Vol. 50, no. 5, pp. 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] Akshit and et al., "*Formation Control and Trajectory Tracking of Nonholonomic Mobile robots*", IEE trans. on Control System Technology, Nov-18

Amol Patil received a B.E.(1998) in Electronics Engineering from North Maharashtra University Jalgaon and M.Tech (2009) in Electrical Engineering with specialization in Control System from VJTI Mumbai. His research includes cooperative control system, Machine learning and Artificial Intelligence.

Gautam Shah received a B.E. in Electronics Engineering from Shivaji University Kolhapur, M.E. in Digital Techniques and Instrumentation from DAV, Indore and PhD in Electronics and Telecommunication Engineering from NMIMS, Mumbai. He has been life time member of IETE India. His research includes control system, digital system and electronics systems. He is involved in electronics system analysis in transform domian.