Wireless Pers Commun (2018) 99:1423–1441
https://doi.org/10.1007/s11277-018-5282-8

A Novel Approach for Fast Average Consensus Under Unreliable Communication in Distributed Multi Agent Networks

Ali Mustafa[1,2] · Muhammad Najam ul Islam[1] · Salman Ahmed[3] · Muhammad Ahsan Tufail[2]

Published online: 17 January 2018

Abstract In this research an algorithm is proposed to find the total number of agents participating in a multi agent network. Also to achieve hasten distributed average consensus in order to consider a network with reliable and unreliable communication links. Class of algorithm is considered in which fixed initial state values are assigned to all agents in the network, with the iterations they updates their initial values by communicating with their neighboring agents within a multi agent network. Algorithm with weighted matrix satisfy the convergence condition of average consensus and accelerate the method to achieve the consensus. Usually this convergence process is relatively sluggish and take moreover numerous iterations to achieve a consensus. To overcome the above issues, a new approach is proposed in order to minimize the rate of convergence. A two step algorithm has been proposed, where in step one each agent employs a linear predictor to predict future agent values. In second step the computed values are used to proceed further by the other agents to achieve consensus in order to bypass the redundant states. In the end proposed algorithm is compared with other existing consensus frameworks to strengthen the claim regarding the proposed two step algorithm which leads to escalate the rate of convergence and reduces the number of iterations.

Keywords Multi agent systems · Unreliable communication · Distributed estimation and consensus control

✉ Ali Mustafa
ali.mustafa@ciit-attock.edu.pk

[1] Department of Electrical Engineering, Bahria University, Islamabad, Pakistan

[2] Department of Electrical Engineering, COMSATS Institute of IT, Attock, Pakistan

[3] Department of Computer System Engineering, University of Engineering and Technology, Peshawar, Pakistan

1 Introduction

Multi agent systems refers to a kind of dynamic system, in which the agent are distributed throughout the system with each component sub-system governed by one or more self-organized control agents [1].

Different protocols are in practice for computing cumulative information by various multi agent systems. Saber and Murray [2] explain the gossip algorithm for the agent counting. Similarly algorithms like AODV & MAODV are in practice for agent counting in adhoc networks [3, 4]. In [5] genetic algorithm, binary decision diagrams, evolutionary algorithms, particle swarm optimization algorithm and memetic algorithm, are discussed in detail for calculation of total participating agent in a network [6]. Results computed in [7] clearly indicate the movement of agents in a particular direction trendy in a distributed coordination [8].

One of the important technique in practice for computing the network agent is distributed average consensus. It gains a mounting popularity and attracted many researcher from various fields. Its major assignment is to compute the average of defined set of values in numerous iterations by interchanging of local information. In average consensus algorithms foremost objective is to ensure the availability of average value at every node in a communication network to achieve their local and global set of goals. Similarly, such algorithms are vigorous in case of communication link failure. The origin and start of these algorithm can be drafted from the research of Borkar and Varaiya [9] and by Tsitsiklis [10]. Average consensus algorithms are widely used in various real time applications. Vehicle formation is one of the emerging fields in control aspect, application in this area varies from underwater to flying vehicles. Author in [11] projected a new control consensus methods to generate new trajectories. Similarly in the application of synchronization, swarming and distributed decision making in multi agent systems with time-dependent communication links are described in detail by [12, 13]. Moreover, flocking is one of the evolving area in distributed consensus, [14] proposed two different strategies focusing on free-space model along with multiple obstacles. In fact distributed average consensus is of a great significance for multiple applications in various fields as discussed, but at the same time it counters the few challenges during the implementation and designing process. Communication between the network agents under the switching and unreliable graph topologies are the main constraint to achieve the consensus in a network. At the same interval, such networks experience the time delays as well, which effects the network performance and increase the processing time to reach the global objective [15]. Likewise, one more major obstacle in consensus is data agreement in the existence of restricted information under dynamic varying topologies. Dynamic changing network topologies leads the networks towards uncertainty and sure consensus in such conditions is one of the hardest scenarios. Ren and Beard [16] suggested few solution schemes for discrete and continuous updates by using the key concepts of spanning tree from graph theory to achieve the distributed consensus and comprehensively produced the results in his research focusing on directed network with switching interaction topologies.

Motivation behind this study, is to establish a consensus algorithm and to address the challenges, in such a way that all agents in a network harvest a distributed control action and settle on some quantity of interest. While designing such algorithm, the main focus is to counter the time delays which may occur when agents exchange the information. Furthermore communication delays, broken communication links and random topology changes in a network are addressed in such a way that the arrangement of the agents that experiences disparities leads the network towards the action of the consensus algorithm and will compute the total number of agents in a network.

In this study, matrix and algebraic graph theories are used as an elementary tools for scheming the projected algorithm. Proposed algorithm consists of a two steps process. The prime focus of the present study is on computing the average consensus for achieving the network local and global objective by initializing each agent with its local initial value for multi agent system having reliable and unreliable communication link with static and dynamic topologies. As the number of iterations, increases these values will be updated and exchange with the other agents in the network [10, 17, 18]. Once the value is computed, that value will be updated for next iterations and will save time and number of iterations to avoid redundancy. This class of algorithms achieves fast convergence to a common average value in less number of iterations and finally results in less overhead and communication burden in a network [19]. Suggested procedure reach the agreement amongst the communication network in a very effectual way by applying the concept of predictor corrector two-step process. In this framework, proposed algorithm is compared with other existing algorithm to demonstrate its effectiveness in terms of different performance parameters.

The arrangement of the extant manuscript is categorized in to seven sections. After introduction, Sect. 2 is comprised of related work. whereas, matrix and graph theory as a tool has been discussed in Sect. 3. Section 4 is consistent of the convergence analysis and posing the state problem. In Sect. 5, proposed algorithm is elucidated. In Sect. 6, the main simulation results of proposed and existing techniques are incorporated. Lastly, concluding remarks are employed in Sect. 7.

2 Related Work

Distributed average consensus algorithms for linear iterations has been deliberated by various researches from diverse arenas to improve the convergence rate. Author in [20] introduced the time delays and filtering strategy to reach the consensus by introducing a Lyapunov function to achieve the agreement between the agents. Similarly, an innovative approached is implemented by Hu et al. [21] by adapting methods from nondependent cascaded systems aimed to achieve distributive consensus control design intended for non-holonomic bound systems. Using weight matrix in optimization of the results, for attaining fast distributed consensus was initially explored by Xiao and Boyd [22, 23]. Their research flourished the problem formulation of a weight matrix which satisfies the network topology limitations and reduces the convergence rate but at the cost of extensive computational resources with the considerable time delays. This framework poses a concern under dynamic topology of a network, it requires recalculation of weight matrix with every change in network topology. So its efficiency is uncertain for such scenarios. Also, this approach involves the complex mathematical equations from the fields of graph and matrix theory using the optimization algorithm. Alternatively, another optimum methodology with far less demanding algorithm was again proposed by Xiao and Boyd [22]. In this proposed method neighboring edge weights are fixed to a constant for the improved calculation for average consensus, however this technique flops under dynamic topology and demands the full statistics of a connectivity topology [22]. In the same way, alternative scheme is projected by Sundaram and Hadjicostis in [24] but it still have the constraint that it only generates optimal results under fixed topology and fails for dynamic connectivity. It keeps the record of complete weight matrix for each agent. One of the most adopted and efficient technique for feedback control systems is Fuzzy logic. Wang et al. [25] investigates the output feedback control problems for nonlinear industrial processes by utilizing Takagi-Sugeno modeling approach. A new technique is suggested in [26] aimed at a nonlinear time delay systems, established on Lyapunov–Krasovskii function to accomplish constancy for a closed loop response schemes centred on estimation possession of neural systems. Moreover, optimization and tracing problem connected to industrial procedures grounded on prediction approach and output response fault tolerant control is investigated in [27]. Wang et al. [28] inspects the multirate closed loop industrial method by using the output chasing to forecast the inclusive performance of a network by setting a sampling period. and promises the system steadiness. Similarly [29] explores the subject of output feedback control for a nonlinear distributed systems. Two dissimilar methodologies based on Lyapunov functional combined with matrix variation convexification techniques are implemented to project the feedback controller to attain agreement for first-order hyperbolic partial differential equations through Markovian jumping actuator faults. Another approach using the concept of weight matrix is proposed in [30] but the researcher intentionally or unintentionally did not provide any information for designing the weight matrix. Whereas, in comparison, the approach of this study, is in concerting the design of weight matrix step by step providing the detail description and their impact on a proposed algorithm.

3 Preliminary Study

Theories from the algebraic graph and matrix theory plays a vital role as a main building tools for designing and implementing an algorithms for the cooperative control of networked multi agent systems.

3.1 Graph Theory

In mathematical modeling, graph theory is of great significant. A set of objects interconnected with each other via links is called a graph. The links objects are graphically represented by a mathematical term called vertices and an edge is a physically link, which connect some pairs of vertices.

In graph theory edge set can be directed or undirected depending on the network topology. In an undirected graph, the edges have no orientation. In a directed graph, the edges have a direction associated with them. Directed graph can be expressed as a graph in which each pair of discrete agents is directionally linked through an edge, therefore there is a directed path amongst the agents in the network. Moreover in directed graph an edge set can be written in a form of a pair $\left(\upsilon_i, \upsilon_j\right)$. This defines the communication among the agent i as a source and agent j as a receiver. On the other hand in undirected graph (v_i, v_j) refers a bidirectional flow of information between the two agents.

Another concept gains indispensable significance is of directed spanning tree, Which states that any agent in a network must be connected through a directed path in communication topology and accessible by at least one vertices [6, 31, 32].

Similarly the whole graph can be connected or unconnected. A graph is said to be connected if there is a connection between every pair of vertices. Directly connected vertices are called neighbor to each other and simply denoted by $\mathcal{N}_i$ for agent i. A linked graph is denoted by $\mathcal{G} = (V, \mathcal{E})$, it consists of set of vertices and the set of edges, which are responsible for connecting the agents in the direction oriented communication within the communication network. Algebraically set of vertices can be expressed as $V = \{\upsilon_1, \upsilon_2, ...\upsilon_N\}$, additionally edge set can be symbolised as $\mathcal{E} = V \times V$. Moreover it is significant to indicate that N reflects the entire agents in a system [6, 31, 32].

3.2 Matrix Theory

In average consensus, the major role is played by a matrix theory in convergence analysis for multi agent system. Matrix theory was initially presented by [33]. Matrix can be categorized into two types, nonnegative (positive) matrix is a matrix in which all of its entries are positive and negative matrix is vice versa. Similarly any vector is said to be nonnegative (positive) if all its elements are nonnegative (positive).

Additionally, stochastic matrix is defined as a matrix in which matrix row sums are equal to + 1, furthermore the matrix is said to be a (row) stochastic matrix P. Stochastic matrix P is called indecomposable and aperiodic (SIA) if $\underset{k \to \infty}{limt} P^k = \mathbf{1} w^T$, where $\mathbf{1}$ is a column vector having all entries equal to 1, w is some column vector and T represents the transpose operation [32, 34]. In matrix theory, one of the most important concept is of a rank 1 matrix. Attribute of rank 1 matrix is having all rows identical which leads the network topology to satisfy the convergence condition to achieve the consensus.

In matrix theory $A = [a_{ij}]$ represents the adjacency matrix of a graph $\mathcal{G}$ and mathematically it can be expressed as:

$$a_{ij} = \begin{cases} 1 & if\left(v_i, v_j\right) \in \mathcal{E} \\ 0 & otherwise \end{cases}$$

Furthermore one the important matrix used in scheming distributed algorithm is degree matrix and mathematically for any communication graph $\mathcal{G}$, it can be written as $D = [d_{ii}]$. It is accountable for providing the degree of vertices with respect to adjacent agents with agent i in a network. Laplacian matrix utilizes the information from both, adjacency and degree matrix. It can be stated as $L = D - A$ [6].

4 Convergence Condition

This study is focusing on a distributed linear iterations, so the solution for primary predictor feedback law is the prediction of future state of the linear system [35]. The findings of the present study will deal with the distributed linear iterations [36, 37], of the form given below:

$$x_i(k+1) = W_{ii}(k)x_i(k) + \sum_{j \epsilon N_i} W_{ij}(k)x_j(k); \tag{1}$$

where $i = 1, 2, 3, ...n$, and $k = 0, 1, 2, ...n$.

The weight on x_j at agent i is denoted as W_{ij}. To achieve a average consensus, we are considering

$$W_{ii}(k) = (1 - \sum_{j\epsilon N_i} W_{ij}(k)) \tag{2}$$

by substituting $W_{ii}(k)$ in Eq. (1)

$$x_i(k+1) = \left(1 - \sum_{j\epsilon N_i} W_{ij}(k))\right) x_i(k) + \sum_{j\epsilon N_i} W_{ij}(k))x_j(k) \tag{3}$$

Now the above equation can be transformed as

$$x_i(k+1) = x_i(k) - \sum_{j\epsilon N_i} W_{ij}(k)x_i(k) + \sum_{j\epsilon N_i} W_{ij}(k))x_j(k) \tag{4}$$

By taking $\sum_{j\epsilon N_i} W_{ij}(k)$ common, we will get

$$x_i(k+1) = x_i(k) + \sum_{j\epsilon N_i} W_{ij}(k)\left(x_j(k) - x_i(k)\right) \tag{5}$$

Let $W_{ij} = 0$ for $j \epsilon N_i$. The vector form of Eq. (1) can be written as: [36, 38].

$$X(k+1) = W(k)X(k) \tag{6}$$

utilizing the definition of a k-step transition matrix, we can write

$$W(k) = W(k-1)....W(1)W(0)$$

where

$$X(k+1) = \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ ..... \\ ..... \\ x_n(k+1) \end{bmatrix}$$

Similarly

$$X(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ ..... \\ ..... \\ x_n(k) \end{bmatrix}$$

Next step is to choose a weight matrix $W(k)$ in such a way that the convergence condition is satisfied for Eq. (6) and $X(k+1)$ converge to the average vector, for any initial value $X(0)$. To achieve average consensus Eq. (6) can be formulated as

$$lim_{k\to\infty} X(k) = \left(\frac{1}{n}\right) 11^T X(0) \tag{7}$$

where $n =$ total number of agents

$$lim_{k\to\infty}X(k) = lim_{k\to\infty}W(k)X(0) \tag{8}$$

By comparing Eqs. (7) and (8), we get

$$lim_{k\to\infty}W(k)X(0) = \left(\frac{1}{n}\right)11^TX(0) \tag{9}$$

The 1 in Eq. (11 is a vector of ones. Now Comparing the terms in Eq. (11), we can write

$$lim_{k\to\infty}W(k) = \left(\frac{1}{n}\right)11^T \tag{10}$$

The convergence factor for dynamic network topologies defined by [39] is given as

$$r_d(W) = \rho\left(W - \frac{11^T}{n}\right) \tag{11}$$

Now from Eq. (11), convergence time can be defined as

$$\tau_d = \frac{1}{\log(1/r_d(W))} \tag{12}$$

5 Proposed Algorithm

Nowadays, the main interest in the field of multiagent systems is to reduce the convergence rate and to minimize the time required to achieve consensus. According to the applications, different techniques have been developed by researchers to reduce the convergence rate and to improve the performance of consensus methods. In [35], a two step agent state prediction technique is used for average consensus. Increasing the numbers of steps will sometimes help in decreasing convergence rate. Inspired and motivated from this research we are proposing a two step average consensus method. The first step of the proposed technique act as a predictor and second step act as a corrector. Mathematically we can write the proposed algorithm as:

Step 1

$$z_i(k) = x_i(k) + \sum_{j \epsilon N_i}\alpha_{ij}(k)\big(x_j(k) - x_i(k)\big) \tag{13}$$

Step 2

$$x_i(k+1) = z_i(k) + 2\sum_{j\epsilon N_i}W_{ij}\big(z_j(k) - z_i(k)\big) + z_i(k)\left(\sum_{j\epsilon N_i}W_{ij}\big(z_j(k) - z_i(k)\big)\right) \tag{14}$$

where x_i represents the initial state condition for agent i and k represents the discrete instants of time.
Where

$$\alpha_{ij}(k) = \begin{cases} \dfrac{1}{\big(max(d_i(k), d_j(k))\big) + 1} & i \neq j \\ 0 & otherwise \end{cases}$$

and similarly

$$W_{ij}(k) = \begin{cases} \dfrac{1}{\big((max(d_i(k), d_j(k))\big) + 1\big)(2 + x_i(k+1))} & i \neq j \\ 0 & otherwise \end{cases}$$

where d_i is the degree of agent i (i.e., the number of neighbors of the agents i) and d_j is the degree of agent i (i.e., the number of neighbors of the agent i)

5.1 Proof of a Proposed Algorithm

Let

$$X(K) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ \ldots\ldots \\ \ldots\ldots \\ x_N(k) \end{bmatrix}, U(K) = \begin{bmatrix} u_1(k) \\ u_2(k) \\ \ldots\ldots \\ \ldots\ldots \\ u_N(k) \end{bmatrix}$$

We can write the following:

$$X(k+1) = \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \ldots. \\ \ldots. \\ x_N(k+1) \end{bmatrix}$$

Anticipated for the entire communication network, universal state equation can be articulated as:

$$X(k+1) = X(k) + U(k) \tag{15}$$

Through calculating the universal input vector, we can demarcated it as:

$$U(k) = W_{ij}(k)(A - D)X(k) \tag{16}$$

For unreliable communication the weight matrix W is not constant that's why we are using the notation $W_{ij}(k)$ in Eq. (16). By substituting Eq. (16) in Eq. (15), the equation will be

$$X(k+1) = X(k) + W_{ij}(k)(A - D)X(k) \tag{17}$$

Utilizing the definition of Laplacian matrix i.e. $L = D - A$, we can further write the following:

$$X(k+1) = X(k) - LW_{ij}(k)X(k) \tag{18}$$

$$X(k+1) = (I - LW_{ij}(k))X(k) \tag{19}$$

Now by comparing Eqs. (6) and (19), we get

$$W(k) = \big(I - LW_{ij}(k)\big) \tag{20}$$

From Eq. (10)

$$lim_{k\to\infty} W(k) = \left(\frac{1}{n}\right) 11^T \tag{21}$$

So we can write,

$$lim_{k\to\infty} \left(I - LW_{ij}(k)\right) = \left(\frac{1}{n}\right) 11^T \tag{22}$$

In case of proposed algorithm, the corrector step (Step 2) is written in the form as

$$X(k+1) = X(k) + 2U(k) + X_i(k)U(k) \tag{23}$$

As from Eq. (16),

$$U(k) = W_{ij}(k)[A - D]X(k)$$

As we know, $L = D - A$, so the above equation becomes,

$$U(k) = -LW_{ij}(k)X(k)$$

Substituting $U(k)$ in Eq. (23)

$$X(k+1) = X(k) - 2LW_{ij}(k)X(k) - LW_{ij}(k)X(k)^2 \tag{24}$$

Taking $X(k)$ common from above equation, we get

$$X(k+1) = X(k)\left(I - 2W_{ij}(k)L - W_{ij}(k)LX(k)\right) \tag{25}$$

Comparing Eqs. (19) and (25), we attain

$$X(k)\left(I - 2W_{ij}(k)LX(k) - W_{ij}(k)L\right) = (I - LW_{ij}(k))X(k) \tag{26}$$

Comparing both sides of Eq. (26), we develop

$$\left(I - 2W_{ij}(k)L - W_{ij}(k)LX(k)\right) = \left(I - LW_{ij}(k)\right) \tag{27}$$

Applying $lim_{k\to\infty}$ on Eq. (27), It will become

$$lim_{k\to\infty}(I - 2W_{ij}(k)L - W_{ij}(k)LX(k)) = lim_{k\to\infty}(I - LW_{ij}(k) \tag{28}$$

Substituting Eq. (22) in Eq. (28), it can be modified as

$$lim_{k\to\infty}\left(I - 2W_{ij}(k)L - W_{ij}(k)LX(k)\right) = \left(\frac{1}{n}\right) 11^T \tag{29}$$

let the number of agents are infinite i.e. $n \to \infty$, then

$$lim_{k\to\infty}\left(I - 2W_{ij}(k)L - W_{ij}(k)LX(k)\right) = 0 \tag{30}$$

$$I - lim_{k\to\infty}(2W_{ij}(k)L + W_{ij}(k)LX(k)) = 0 \tag{31}$$

$$I = lim_{k\to\infty}(2W_{ij}(k)L + W_{ij}(k)LX(k)) \tag{32}$$

$$I = lim_{k\to\infty}(W_{ij}(k)L)(2 + X(k)) \tag{33}$$

In case of average consensus, Choosing the suitable value of L, $W_{ij}(k)$ from Metro-Polis Hasting weights, we can express

$$W_{ij}(k) = \frac{1}{\left(max(d_i(k), d_j(k))\right) + 1} \tag{34}$$

By practicing the Heuristics, the final output of Eq. (33) will become

$$W_{ij}(k) = \frac{1}{\left(\left(max(d_i(k), d_j(k)\right) + 1\right)(2 + X(k))} \tag{35}$$

In this paper, we are proposing a method that reduces the convergence rate and compute the total number of agents actively participating in a network under dynamically changing topologies. In order to count the total number of agents in the system, initial values to the agents have been assigned as the initial values to agents as $x_n(0) = 1$ and $x_i(0) = 0$, $\forall i = 1, 2, 3, \ldots n - 1$. In this case, the consensus among agents will developed on $\frac{1}{n}$. Inverse of consensus value will give us the total of agents.

5.2 Local-Degree Weights

There are different approaches to design a weight matrix W. One simple technique is a local degree weights. In this technique, the largest out-degree of two incident agents are assigned as weight on each edge [36].

$$W_{ij}(k) = \begin{cases} \frac{1}{\left(\left(max(d_i(k), d_j(k))\right)\right)} & i \neq j \\ 0 & otherwise \end{cases} \tag{36}$$

In case of local degree weights, it is necessary that each node knows the out-degrees of all its neighboring agents.

5.3 Metropolis Hasting Weights

Another simple approach similar to local degree weights for wight matrix, is Metropolis Hasting weights, denoted as follows [40, 41]

$$W_{ij}(k) = \begin{cases} \frac{1}{\left(\left(max(d_i(k), d_j(k))\right)\right) + 1} & i \neq j \\ 0 & otherwise \end{cases} \tag{37}$$

In case of Metropolis Hasting weights, it is necessary that each agent knows the out-degrees of all its neighboring agents but the set of neighbors vary with time.

6 Numerical Examples and Simulation Results

This section is comprised of proposed numerical examples of unreliable communication among multi agents with interaction topologies changing dynamically and compare the performance of proposed algorithm with other existing consensus protocols. It is important to mention here that tolerance considered for all the network in the simulation is $\varepsilon = 10^{-20}$. Where No.of iterations are represented by NI in all the tables.

where

$$e_i(k) = \sum_{j \epsilon N_i} |x_i(k) - x_j(k)|, \quad i = 1, 2, ...n \tag{38}$$

6.1 Example 1

In this example, total 45 number of agents are considered which are connected with dynamically changing topology. As the topology changes in each iteration, the weight matrix will change automatically in each iteration and it will become difficult for agents to develop consensus on a common value. The consensus value in this example is $\frac{1}{45}$. Here we are assigning initial values as $x_n(0) = 1$ and $x_i(0) = 0$, $\forall i = 1, 2, 3, \ldots n-1$. The communication topology considered for this numerical example is shown in Fig. 1. While an error is plotted in Fig. 2 for proposed method, metropolis and local degree method respectively, to evaluate the performances of different methods. From gathering the results of the simulation, it is noticeable that the proposed method consuming less time and achieving faster convergence to consensus value as compared to metropolis and local degree method. To concrete the above results, Simulation outcomes are shown in Table 1.

6.2 Example 2

This example is the continuity of example 1. As in example 1, random number of agents are selected and consensus is developed while in this example, after developing consensus some number of agents are randomly removed from the system. This example is interesting

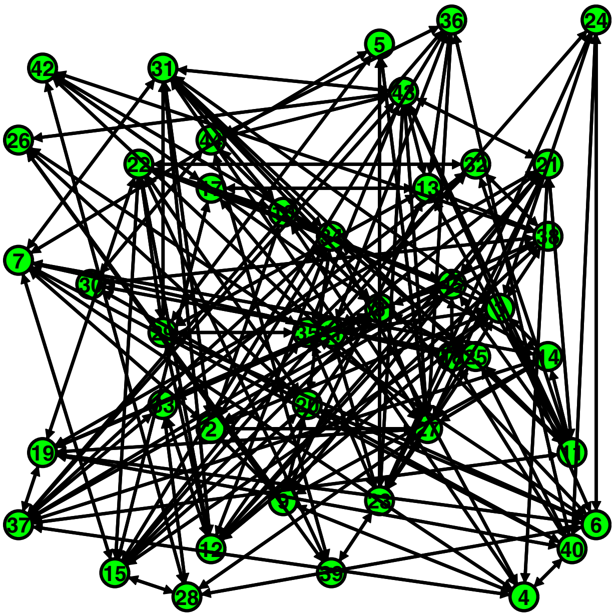


Fig. 1 Network topology considered in computing example 1

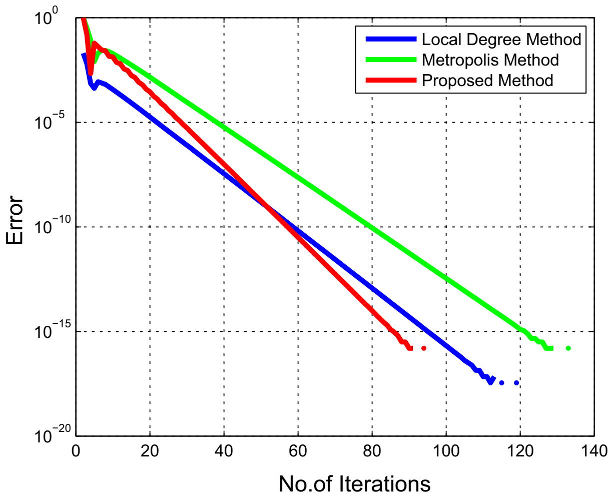


Fig. 2 Error graph for example 1

Table 1 Results of example 1

Iterative methods	NI	$\rho\left(W - \frac{11^T}{n}\right)$	τ_{asym}
Local degree	121	0.726	3.129
Metropolis	136	0.746	3.414
Proposed method	77	0.610	2.027

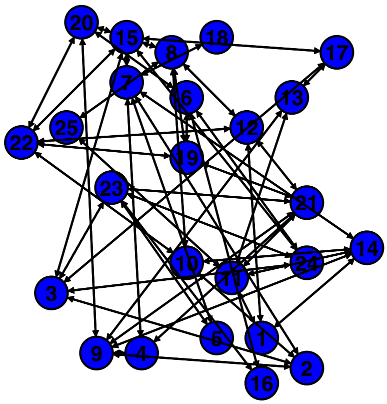


Fig. 3 Network topology for computing example 2

in such a way that in example 1 the consensus is developed at some low value, but now in in this case, the agents have to approach at some higher value to achieve consensus, as some of the agents have been removed from the system. Now the number of iterations are unnecessarily increased instead of decreasing. In current example, 20 agents are removed from the system and network left with total 25 agents, the consensus value changes to $\frac{1}{25}$. The network topology considered for this example is shown in Fig. 3. Figure 4 presents a plots of error for proposed method, metropolis, and local degree method respectively. The

Fig. 4 Error graph for example 2

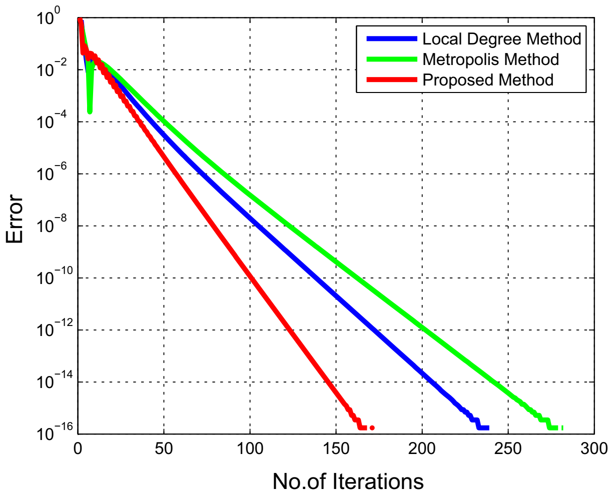


Table 2 Results of example 2

Iterative methods	NI	$\rho\left(W - \frac{11^T}{n}\right)$	τ_{asym}
Local degree	255	0.863	6.794
Metropolis	301	0.880	7.853
Proposed method	164	0.80	4.504

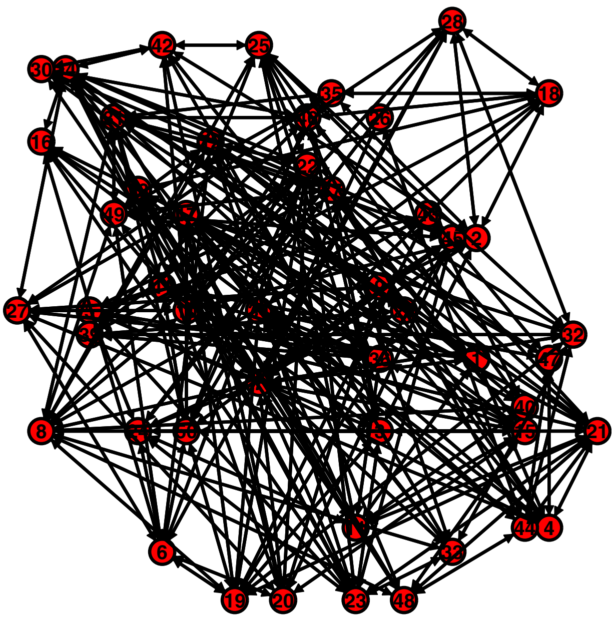
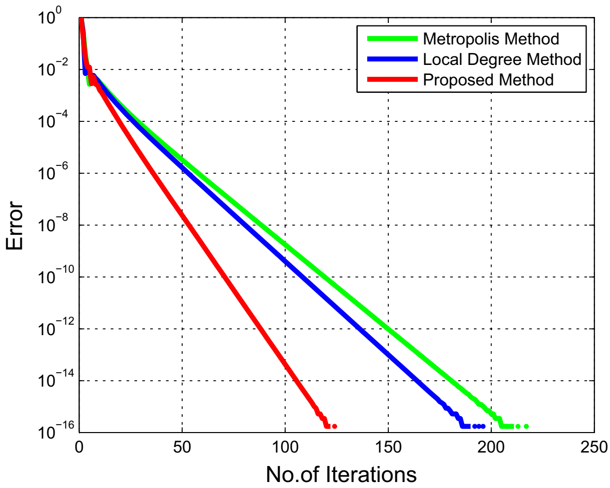


Fig. 5 Network topology for computing example 3

Fig. 6 Error graph for example 3



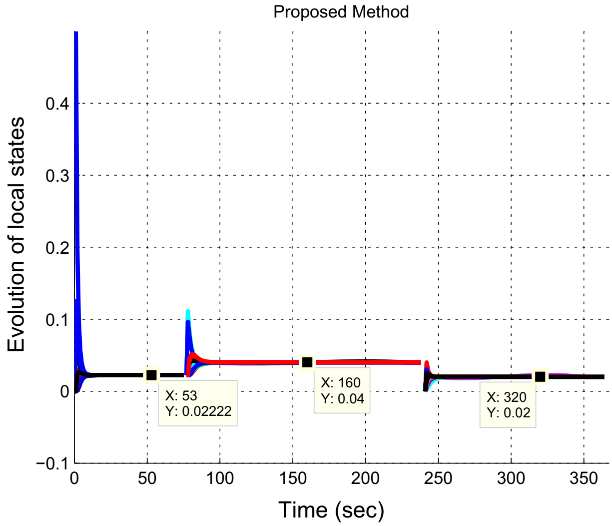


Fig. 7 Combined results of all examples utilizing proposed algorithm

number of iterations, convergence time and asymptotic convergence factor for all three methods are shown in Table 2. From the results, it has been observed that the proposed algorithm is generating the best results as compare to the other methods.

6.3 Example 3

Similarly this example is the continuity of example 2. After removing the number of agents from the system and developing consensus, now some number of agents are randomly added in the system. The continuity examples are one of the best way to analyze the performances of iterative methods. In this example, the randomly added agents are 25, so the total agents in the system are now 50. The consensus value of for this case becomes $\frac{1}{50}$. The graph topology considered in this example is shown in Fig. 5. Similarly error of all three methods for current scenario is plotted in Fig. 6. In Figs. 7, 8 and 9 a combined graph of all three examples for proposed method, metropolis and local degree method have been presented. The number of iterations, convergence time and asymptotic convergence factor for all three methods are shown in Table 3. From the result shown in Table 3, it is concluded from the results that the proposed algorithm consuming the less time to develop consensus as compared to other consensus protocols.

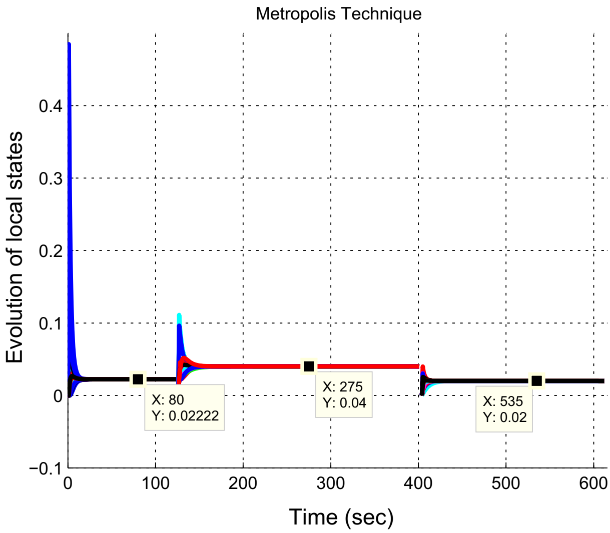


Fig. 8 Combined results of all examples utilizing Metropolis Hasting method

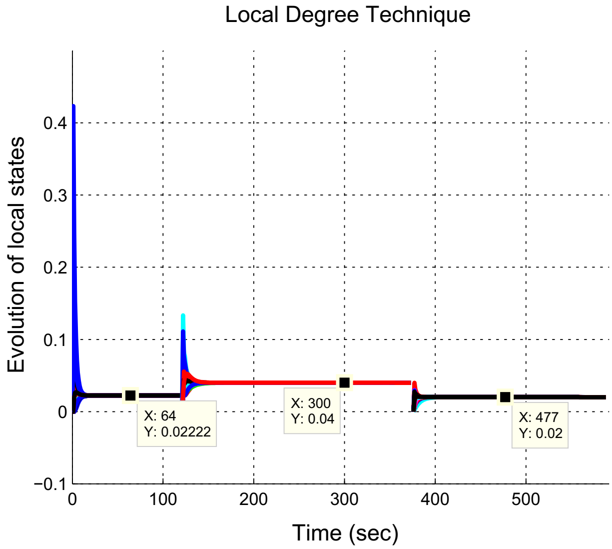


Fig. 9 Combined results of all examples utilizing local degree method

Table 3 Results of example 3

Iterative methods	NI	$\rho\left(W - \frac{11^T}{n}\right)$	τ_{asym}
Local degree	215	0.837	5.641
Metropolis	237	0.846	5.992
Proposed method	126	0.748	3.456

7 Conclusion

Experimental simulation results exhibited in the present research, clearly indicate that the proposed method is much efficient and fast in achieving the consensus amongst the multi agents to achieve their local and global goal as compared to the rest of the existing methods in terms of number of iterations and time. Performance of different algorithms are clearly visible from the combined graphs in Figs. 7, 8 and 9. Proposed algorithm is achieving consensus in 300 iterations while Metropolis is taking 600 iterations and Local Degree Method completing consensus in 500 iterations. So it is evident that proposed algorithm is quite fast and converge to a consensus value in almost half of the time utilized by the other existing algorithms.

References

1. Kahani, M., & Beadle, H. (1997). Decentralized approaches for network management. *Computer Communications*, *27*, 36–47.
2. Demers, A., Greene, D., Hauser, C., Irish, W., Larson, L., Shenker, S., Sturgis, H., Swinehart, D., & Terry, D. (1987). Epidemic algorithms for replicated database maintenance. In *Proceedings of 6th annual ACM symposium on principles of distributed computing* (pp. 1–12).
3. Evers, J., Kiss, D., Kowalczyk, W., Navilarekallu, T., Renger, A. L. S. M., Timperio, A. A. V. V., Wijk, S. V., & Yzelman, A. J. (2011). Node counting in wireless ad-hoc networks. In *Proceedings of the 79th European study group mathematics with industry* (pp. 49–73).
4. Perkins, C. E., & Royer, E. M. (1999). Adhoc on-demand distance vector routing. In *Second IEEE workshop on mobile computing systems and applications* (pp. 90–100).
5. Rehan, S., & Bansal, M. (2013). Performance comparison among different evolutionary algorithms in terms of node count reduction in bdds. *International Journal of VLSI and Embedded Systems*, *4*, 491–496.
6. Mustafa, A., Ahmed, S., Islam, N., & Tufail, A. (2016). An efficient agent scheming in distributed time varying networks. In *In proceedings of SAI intelligent systems conference* (pp. 804–809).
7. Vicsek, T., Czirok, A., Benjacob, E., Cohen, I., et al. (1995). Novel type of phasetransition in a system of self-driven particles. *Physical Review Letters*, *75*(6), 122–1229.
8. Yu, H., Xia, X., & Zhang, T. (2011). A less conservative method for average consensus with multiple time-varying delays. *IEEE Africon*, 1–6.
9. Borkar, V., & Varaiya, P. (1982). Asymptotic agreement in distributed estimation. *IEEE Transactions on Automatic Control*, *27*(3), 650–655.
10. Tsitsiklis, J. (1984). Problems in decentralized decision making and computation. Ph.D. Thesis, Massachusetts nstit. Technology, Cambridge.
11. Fax., J. (2001). Optimal and cooperative control of vehicle formations. Ph.D. Thesis, Control Dynamical Syst., California Inst. Technol., Pasadena, CA.
12. Moreau, L. (2005). Stability of multi-agent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, *50*, 169–182.
13. Moreau, L. (2004). Stability of continuous-time distributed consensus algorithms. In *Proceedings of the 43rd IEEE conference on decision and control* (pp. 3998–4003).
14. Saber, R. (2006). Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, *51*(3), 401–420.

15. Saber, R. O., & Murray, R. M. (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, *49*(9), 1520–1533.
16. Ren, W., & Beard, R. W. (2005). Consensus seeking in multi agent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, *50*(5), 655–661.
17. Xiao, L., Boyd, S., & Lall, S. (2005). A scheme for robust distributed sensor fusion based on average consensus. In *Proceedings of IEEE/ACM international symposium on information processing in sensor networks* (pp. 63–70).
18. Xiao, L., Boyd, S., & Kim, S. (2007). Distributed average consensus with least-mean-square deviation. *Journal of Parallel and Distributed Computing*, *67*(1), 33–46.
19. Aysal, T. C., Oreshkin, B. N., & Coates, M. J. (2009). Accelerated distributed average consensus via localized node state prediction. *IEEE Transactions on Signal Processing*, *57*(4), 1563–1576.
20. Saber, R. O., & Murray, R. M. (2003). Consensus protocols for networks of dynamic agents. In *Proceedings of the American control conference*.
21. Hu, H. X., Yu, L., Zhang, W. A., & Song, H. (2013). Group consensus in multi-agent systems with hybrid protocol. *Journal of the Franklin Institute*, *350*(3), 575–597.
22. Xiao, L., & Boyd, S. (2004). Fast linear iterations for distributed averaging. *System and Control Letters*, *53*(1), 65–78.
23. Boyd, S., Ghosh, A., Prabhakar, B., & Shah, D. (2006). Randomized gossip algorithms. *IEEE Transaction of Information Theory*, *52*(6), 2508–2530.
24. Sundaram, S., & Hadjicostis, C. (2007). Distributed consensus and linear function calculation in networks: An observability perspective. In *Proceedings of IEEE/ACM of international symposium of information processing sensor networks* (pp. 99–108).
25. Wang, T., Qiu, J., Gao, H., & Wang, C. (2016). Network-based fuzzy control for nonlinear industrial processes with predictive compensation strategy. In *IEEE transaction on systems, man, and cybernetics: systems* (pp. 1–11).
26. Wang, T., Qiu, J., & Gao, H. (2016). Adaptive neural control of stochastic nonlinear time-delay systems with multiple constraint. In *IEEE transaction on systems, man, and cybernetics: systems* (pp. 1–9).
27. Wang, T., Huijun Gao, F., & Qiu, J. (2016). A combined fault-tolerant and predictive control for network-based industrial processes. *IEEE Transaction on Industrial Electronics*, *63*(4), 2529–2536.
28. Wang, T., Gao, H., & Qiu, J. (2016). A combined adaptive neural network and nonlinear model predictive control for multirate networked industrial process control. *IEEE Transaction on Neural Networks and Learning Systems*, *27*(2), 416–425.
29. Qiu, J., Ding, S. X., Gao, H., & Yin, S. (2016). Fuzzy-model-based reliable static output feedback h control of nonlinear hyperbolic PDE system. *IEEE Transaction on Fuzzy Systems*, *24*(2), 388–400.
30. Cao, M., Spielman, D. A., & Yeh, E. M. (2006). Accelerated gossip algorithms for distributed computation. In *Proceedings of 44th annual conference of communication, control and computation* (pp. 952–959).
31. Feng, Y., Xu, S., & Zhang, B. (2014). Group consensus control for double integrator dynamic multi-agent systems with fixed communication topology. *International Journal of Robust and Nonlinear Control*, *24*(3), 532–547.
32. Ren, W.,. Beard, R. W., & Atkins, E. M. (2005). A survey of consensus problems in multi-agent coordination. In *American control conference* (pp. 1859–1864).
33. Gantmacher, F. R. (1959). *Matrix theory* (Vol. 2). New York: Chelsea.
34. Meng, X., & Chen, T. (2013). Event-based agreement protocols for multi-agent networks. *Automatica*, *49*(1), 2125–2132.
35. Wei, Y., & Lin, Z. (2015) On the delay bounds of linear systems under delay independent truncated predictor feedback: The state feedback case. In *54th annual IEEE conference on decision and control*.
36. Xiao, L., & Boyd, S. (2003). Fast linear iterations for distributed averaging. In *Proceedings of of 42nd IEEE conference on decision and control*, *5*, pp. 4997–5002.
37. Merris, R. (1994). Laplacian matrices of graphs: A survey. *Linear Algebra Application*, *197*, 143–176.
38. Kar, S., & Moura, J. (2008). Sensor networks with random links: Topology design for distributed consensus. *IEEE Transactions on Signal Processing*, *56*, 3315–3326.
39. Kokiopoulou, E., & Frossard, P. (2009). Polynomial filtering for fast convergence in distributed consensus. *IEEE Transactions on Signal Processing*, *57*, 342–354.
40. Hastings, W. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*, *57*, 97–109.
41. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, *21*, 1087–1092.

Ali Mustafa is currently a Ph.D. scholar in Bahria University Islamabad, Pakistan in the department of Electrical Engineering. Also he is serving as an Assistant Professor in Electrical Engineering department in a well reputed Pakistani University (COMSATS, National ranking 1st in Information Technology and 4th in overall universities of Pakistan). His area of interest is in the field of wireless communications, multi agent systems, and wireless sensor networks. He did his B.S. in computer engineering from Pakistan and completed his master degree in Mobile Computing from the University of Bradford, United Kingdom in 2008.

Dr. Muhammad Najam ul Islam is working as a DEAN of the Electrical Engineering department of Bahria University, Islamabad since January 2011. Najam did his Ph.D. in Electrical Engineering from Telecom ParisTech, France in 2010. During his doctoral studies, he was working at EURECOM, Sophia Antipolis on the research theme of flexible radios for multi-standard wireless communication devices. He received his Masters Degree in Embedded Systems from University of Nice-Sophia Antipolis, France, in 2006. Before moving to France, Najam completed his Masters in Computer Sciences from LUMS, Lahore in 2005. He worked as adjunct faculty, research assistant and teaching assistant at LUMS in 2003, 2004. Najam has also worked for NESCOM (1999– 2003) as Assistant Manager. He joined NESCOM after completing his Bachelors in Electrical Engineering from UET, Lahorein1999.

Dr. Salman Ahmed is an Assistant Professor in Computer system engineering department in University of Engineering and Technology, Peshawar, Pakistan. He has done his Ph.D. in Control Systems form University of Alberta, Canada and B.S./M.S. in computer system engineering from University of Engineering and Technology, Peshawar, Pakistan. His area of interest is in the field of control system, multi agent systems, wireless sensor networks and feedback systems.

Muhammad Ahsan Tufail is currently a student of Bachelor of Science in Electrical Engineering from COMSATS Institute of Information Technology, Attock, Pakistan (A well reputed Pakistani University (COMSATS, National ranking 1st in Information Technology and 4th in overall universities of Pakistan). His area of interest is in the field of wireless communications, multi agent systems, and wireless sensor networks. He is also working on various projects.