8

Diffusion protocols for link failure detection and resource utilization

Information diffusion on network systems is a common phenomena which arises in many applications, in particular, networked information services [Pavlin *et al.*, 2010; Zhu, 2006], decision making in sensor networks [Sandhu *et al.*, 2004], manufacturing control [Bussmann *et al.*, 2004], etc. Several information diffusion protocols have been introduced in literature for various applications including multi-agent agreement and coordination in which a set of dynamic agents are connected via a network, known as multi-agent system (MAS) [Olfati-Saber *et al.*, 2007; Mesbahi and Egerstedt, 2010]. One of the important issues in the domain of MASs is related to data fusion among the agents and to take decisions based on the available data (information) at each individual agent.

In this chapter we consider three inter-related problems which apparently seem to be different but have importance in information propagation in real world networked systems. These problems are - (i) automated detection of link failures by observing the nature of information propagation, (ii) ensuring security and utilization of network resources during information propagation, and (iii) a static fixed point convergence of information propagation in a dynamic network. Such problems have been widely studied in the existing literature, however, only a few works consider these from the network science perspective. Existing studies in these directions have mainly considered different agreement or consensus protocols such as [Kashyap *et al.*, 2007; Kuramoto, 2012; Olfati-Saber, 2006; Cortes *et al.*, 2004; Ando *et al.*, 1999; Lin *et al.*, 2004; Cortés *et al.*, 2006] and the references therein.

We mention that the transportation of resources from one place to another often happens through multi-agent networked systems, for example data transmission through Internet, electric power supply through power grid network, road network, rail network and many other real world networks that are used to transfer the resources across the globe. In these networks, failure of a link affects the performance of the entire networked system resulting underutilization of resources or loss of information. Thus detection of link failures in a MAS can help to improve the performance of the affected systems in an effective manner. In the existing literature, researchers have provided some solutions that are very specific to the data transmission through Internet [Long and Sikdar, 2007; Iannaccone *et al.*, 2002; Van Renesse *et al.*, 1998]. In a different direction, authors looked the problem of link failure detection under the agreement protocol in MASs [Rahimian *et al.*, 2012a; Rahimian and Preciado, 2013; Rahimian *et al.*, 2012b].

The three problems which we consider in this chapter are related to information spreading or information propagation over a MAS. A link failure and/or irregularity of amount of desired resources originate only in few of the agents in the system, and then they affect the diffusion process such that the protocol does not converge to a desired state. Indeed in [Jafari *et al.*, 2011; Rahimian and Aghdam, 2013] the authors rigorously describe how the cooperative dynamics over a network may get strongly affected by the network failures due to the removal of some links or nodes in the network. Obviously, it is of paramount interest for development of protocols for information diffusion which can help solving the above three problems by observing the nature of diffused information at system convergence.

In this chapter we propose a diffusion protocol using the approach of networked multi-agent systems in which the diffusible resources in the connected nodes i and j flow through an existing edge (i, j) according to weighted difference of the resources available in the nodes i and j. These weights are reciprocal of the weights which are fixed positive real numbers bigger than 1 associated with the individual agents and assigned in the beginning of the protocol but can be modified during the diffusion process. The diffusion under the proposed protocol converges to a unique state that depends on the weights of the agents. Due to the dependency of the converging sate over the weights, it has many practical applications as weights can be managed according to the requirement of applications. We investigate two immediate applications of this protocol in which the first one is the detection of a link failure and redistribution of resources in a networked system, and the other application is related to the distribution of resources in dynamic networked systems. If the weights are correlated with degrees (structural property of agents) of the nodes in the considered network, then it works as an indicator of the failure of a link. In this case, failure of a link shifts resources from the affected part of the network to the remaining network which improves the utilization of the resources. In the other application, if we require a fix distribution of resources in the network without being affected by the structural changes then we need to fix the weights of the agents that are correlated with the required converging state. We provide analytical proofs for convergence of the proposed diffusion protocol in static network as well as in dynamic networks. Indeed we show that the rate of convergence of the standard agreement protocol is faster than that of the proposed diffusion protocol. Next we describe the effect of link failure over the redistribution of resources in the nodes of the network under the given conditions. Finally the proposed diffusion mechanism is simulated for various real world network scenarios to analyze its performance and applicability in different applications.

The rest of the chapter is organized as follows. In Section 2 we provide a brief literature survey on existing agreement protocols and its applications as information diffusion protocols in different areas. In Section 8.2, we define the diffusion protocol which is investigated under the fixed (or static) and switching (or dynamic) topology of the networks. We discuss the convergence of the proposed diffusion process for both the static and dynamic network topology, and compare the rate of convergence of this protocol with the agreement protocol in [Rahimian and Preciado, 2013]. Details of the applications of the proposed diffusion protocol are discussed in the Section 8.3 followed by Section 8.4, in which simulation results are presented regarding the theory developed in the Section 8.2. The simulated analysis of applications over the proposed diffusion protocols is discussed in Section 8.5. Finally, Section 8.6 concludes the chapter with future directions of this work.

8.1 RELATED WORKS

As the proposed diffusion protocol is a variant of the standard agreement protocol [Jadbabaie *et al.*, 2003; Olfati-Saber and Murray, 2004], we first briefly review it for MASs. For a survey of agreement protocols see [Ren *et al.*, 2005a]. Let G be the connected network associated with a MAS on n agents. The adjacency matrix associated with G is defined by $A = [a_{ij}]$ where $a_{ij} = 1$ if the ith and j^{th} nodes of G are linked by an edge, otherwise $a_{ij} = 0$. The combinatorial Laplacian matrix associated with G is defined as $L = D - A$ where $D = \text{diag}\{d_1, \ldots, d_n\}, d_i = \sum_{j=1}^{n} a_{ij}, i = 1, \ldots, n$. Let $x_i(t)$ be the amount of information available at the ith agent at time t that needs be shared among the other agents in the network. Then a continuous-time consensus protocol can be described as

$$\dot{x}_i(t) = -\sum_{j=1}^{n} a_{ij}(x_i(t) - x_j(t)) \tag{8.1}$$

which can be written in the matrix form

$$\dot{\mathbf{x}}(t) = -L\mathbf{x}(t) \tag{8.2}$$

where $\mathbf{x}(t) = [x_1(t), \ldots, x_n(t)]^T$. Hence the state dynamics of the associated MAS depends on its Laplacian eigenvalues and finally the system reaches to agreement, that is $\|x_i(t) - x_j(t)\| \to 0, i \neq j$ as $t \to \infty$. The rate of convergence depends on the algebraic connectivity of the network which is the least positive Laplacian eigenvalue of G. Note that L is a symmetric positive semidefinite matrix with one of the eigenvalues zero and other eigenvalues are positive when G is connected, hence the eigenvalues can be ordered [Mesbahi and Egerstedt, 2010].

Thus the aim of the agreement problem is to achieve a certain desired goal with the help of local dynamics in the associated MAS in which a node interacts with its neighboring nodes only and the whole system reaches to some required state asymptomatically [Jadbabaie *et al.*, 2003; Li and Guo, 2015]. Some of the applications of agreement and cooperation protocols include load balancing [Kashyap *et al.*, 2007], synchronization [Kuramoto, 2012], flocking [Olfati-Saber, 2006] [Cortes *et al.*, 2004] [Tanner *et al.*, 2007], rendezvous in space [Ando *et al.*, 1999][Lin *et al.*, 2004] [Cortés *et al.*, 2006], information sharing among the devices which are equipped with sensors [Spanos *et al.*, 2005][Olfati-Saber *et al.*, 2007][Miao *et al.*, 2016] [Wan *et al.*, 2016]. See also [Qiu *et al.*, 2016] [Yu *et al.*, 2010][Luck and McBurney, 2008][Tahbaz-Salehi and Jadbabaie, 2008][Tanner *et al.*, 2003]. Agreement protocols for MASs with switching topology can also be found in [Sontag, 1983] [You *et al.*, 2013]. The consideration of bidirectional and unidirectional diffusion is an important subject of study[Li *et al.*, 2013a][Ren *et al.*, 2005a] in this regard. Apart from agreement, a lot of interest is generated for studying the controllability of MASs by either exploiting the agreement protocol or its variants with the introduction of feedbacks to some of the nodes known as leaders, for example, see [Mesbahi and Egerstedt, 2010; Lewis *et al.*, 2016; Wan *et al.*, 2016; Liuzza *et al.*, 2016; Katsoukis and Rovithakis, 2016].

Recently the standard agreement protocol given by the Eq. (8.2) has been exploited for detection and isolation of single or multiple link failures in MASs [Rahimian and Preciado, 2013; Rahimian *et al.*, 2012b] by using a directed information flow graph associated with the MAS. An algorithm for sensor placement is proposed, which enables the designer to detect and isolate any link failures across the network based on the observed jump discontinuities in the derivatives of the output responses of a subset of nodes [Rahimian and Preciado, 2013] in the associated MAS. Nevertheless, for a class of directed networks with rooted out-branchings, it is observed that single and multiple link failures can be detected when Eq. (8.2) governs the diffusion process [Rahimian *et al.*, 2012a,b].

Finally we conclude this section with a note that to the best of authors' knowledge the literature lacks any linear information diffusion protocol for identifying link failures as a byproduct of the protocol, in contrast to calculating the derivatives of output responses of some nodes. In addition to that this chapter contributes to utilization of resources in a MAS efficiently by detecting the link failures. Thus the proposed protocol has the potential to be applied in many practical applications related to resource distribution and fault detection in real-world networked systems.

8.2 PROPOSED PROTOCOL FOR DIFFUSION ON MASS

In this section, we propose a diffusion protocol which is based on the push policy of information sharing on a network. In push dynamics, a node j acts as a sender, and node i, connected by an edge to the node j, acts as a receiver. Static and switching (or dynamic) topologies are considered to investigate the performance of the diffusion process. In a static network, structure of the network remains fixed whereas, in switching (or dynamic) topology of a network, the number of nodes is fixed but the connections between them change over time.

8.2.1 Diffusion on Static Networks

Let G be a simple connected network on n nodes associated with a MAS. Let $x_i(t) \in \mathbb{R}$ be a variable associated with the node i that represents the amount of resource at node i in time t. The rate of change in the amount of resources available at node i at time t depends on the amount of resource available at its neighbors and is described by

$$\dot{x}_i(t) = \sum a_{ij} \left(\frac{x_j(t)}{\omega_j} - \frac{x_i(t)}{\omega_i} \right), t > 0 \tag{8.3}$$

where $\omega_i \geq 1, i = 1, \ldots, n$ is a weight affiliated to the node i which is decided by the designer of the MAS, and a_{ij} denotes the ij^{th} entry of the adjacency matrix corresponding to N. Obviously $\dot{x}_i(t)$ is the cumulative difference of the weighted resources available at node i and its neighbors. Note that the weight ω_i signifies that at any time, total amount of resource (like bandwidth in a communication network or information density in a social network) at the node i that is not visible to its neighbors, however, associated with a numeric value which is proportional to the amount of resources available or consumed by the node. Thus $\omega_i > \omega_j$ indicates that node i has more resources compared to the node j.

The diffusion dynamics can be described by the matrix equation

$$\dot{\mathbf{x}}(t) = -L\Omega^{-1}\mathbf{x}(t),$$

where Ω is a diagonal matrix, $\Omega_{i,i} = \omega_i$, and L is the Laplacian matrix associated with G. Hence we have an autonomous linear dynamical system described as

$$\dot{\mathbf{x}}(t) = -M\mathbf{x}(t), \quad M = L\Omega^{-1}. \tag{8.4}$$

It is evident that if $\omega_i = \omega_j = 1$ for all $i \neq j$ then this diffusion protocol is the standard agreement protocol discussed in the previous section. Indeed observe that in Eq. (8.2) the system matrix is symmetric positive semidefinite but in (8.4) the system matrix M is not a symmetric matrix.

For an initial condition $\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^n$ the solution of (8.4) can be derived as

$$\mathbf{x}(t) = e^{-Mt}\mathbf{x}_0, t \geq 0.$$

In order to investigate the asymptotic behavior of $\mathbf{x}(t)$ as $t \to \infty$ we first focus on the eigenvalues of M. We show in the following lemma that in spite of M being a non-symmetric matrix, the eigenvalues of M are non-negative real numbers.

Lemma 8.2.1. *The eigenvalues of $M = L\Omega^{-1}$ are non-negative real numbers.*

Proof. Note that $M = L\Omega^{-1} = \Omega\Omega^{-1}L\Omega^{-1}$. Suppose $Q = \Omega^{-1}L\Omega^{-1}$ which is a real symmetric positive semi-definite matrix and Ω is a positive definite diagonal matrix. Let $\lambda(M)$ be an eigenvalue of M corresponding to an eigenvector $\mathbf{v}$. Then,

$$M = \Omega Q \Rightarrow M\mathbf{v} = \Omega Q\mathbf{v} \Rightarrow \lambda(M)\mathbf{v} = \Omega Q\mathbf{v}$$

This implies,

$$\Rightarrow Q\mathbf{v} = \lambda(M)\Omega^{-1}\mathbf{v} \tag{8.5}$$

and

$$\mathbf{v}^*Q = \lambda^*(M)\mathbf{v}^*\Omega^{-1} \tag{8.6}$$

By pre-multiplying $\mathbf{v}^*$ in Eq. (8.5) and post-multiplying $\mathbf{v}$ in Eq. (8.6) and then subtracting them, we obtain,

$$\begin{aligned}\mathbf{v}^*Q\mathbf{v} - \mathbf{v}^*Q\mathbf{v} &= \lambda^*(M)\mathbf{v}^*\Omega^{-1}\mathbf{v} - \lambda(M)\mathbf{v}^*\Omega^{-1}\mathbf{v}\\ \Rightarrow 0 &= (\lambda^*(M) - \lambda(M))\,\mathbf{v}^*\Omega^{-1}\mathbf{v}\end{aligned}$$

Therefore, $\lambda^*(M) = \lambda(M)$, that is $\lambda(M)$ is real. Further, from Eq. (8.5), we obtain $\lambda(M) = \dfrac{\mathbf{v}^TQ\mathbf{v}}{\mathbf{v}^T\Omega^{-1}\mathbf{v}} \geq 0$. □

The following theorem describes the convergence of the diffusion protocol. The proof is similar to the proof of Proposition 3.11 in [Mesbahi and Egerstedt, 2010].

Theorem 8.2.2. *The diffusion protocol given by Eq. (8.4) converges to the state vector x^* as $t \to \infty$ such that $x^* \propto \omega$ where $\omega = [\omega_1,\ \omega_2\ ,\dots,\ \omega_n]^T$, and the rate of convergence depends on $\lambda_2(M)$, the second smallest eigenvalue of the matrix M.*

Proof. Note that L and M have same rank $n-1$, thus zero is an eigenvalue of M with algebraic multiplicity one. Further as the all-one vector is an eigenvector corresponding to the eigenvalue 0 of L, the weight vector ω is an eigenvector corresponding to the eigenvalue 0 of M. Let $M = XJ(\Lambda_M)X^{-1}$ be the Jordan decomposition of matrix M where X is a non-singular matrix. Then,

$$X^{-1}MX = J(\Lambda_M) = \begin{bmatrix} J(0) & 0 & \dots & 0\\ 0 & J(\lambda_2(M)) & \dots & 0\\ \vdots & \vdots & \vdots & \vdots\\ 0 & 0 & \dots & J(\lambda_n(M))\end{bmatrix}.$$

where $J(0)$ and $J(\lambda_i(M))$ denote the Jordan blocks corresponding to the eigenvalue 0 of M and the i^{th} eigenvalue $\lambda_i(M)$ of M respectively, $i = 1,\dots,n$. Further

$$MX = XJ(\Lambda_M)$$

yields $MX_1 = 0$, hence $X_1 \in \text{span}\{\omega\}$.

$$X^{-1}M = J(\Lambda_M)X^{-1},$$

which further implies that the first row of X^{-1}, that is Y_1, is the left eigenvector of the matrix M associated with its eigenvalue 0. By Lemma 8.2.1 it follows that $\lambda_1(M) = 0$ and all the other eigenvalues of M are positive. Since $X^{-1}X = I$, it follows that $X_1^TY_1 = 1$, where X_1 and Y_1 are the first column and first row vectors of matrices X and $X^{-1} = Y$, respectively. Therefore,

$$e^{-Mt} = X\begin{bmatrix} e^0 & 0 & \dots & 0\\ 0 & e^{J(-\lambda_2(M))t} & \dots & 0\\ \vdots & \vdots & \vdots & \vdots\\ 0 & 0 & \dots & e^{J(-\lambda_n(M))t}\end{bmatrix}X^{-1}.$$

Now we show that

$$\lim_{t\to\infty} e^{J(-\lambda_i(M))t} = 0$$

for $i \geq 2$. Assume that $\lambda_i(M)$ has algebraic multiplicity $s \geq 1$. Then,

$$J(\lambda_i(M)) = \begin{bmatrix} \lambda_i(M) & 1 & 0 & \dots & 0\\ 0 & \lambda_i(M) & 1 & \dots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ 0 & 0 & 0 & \dots & \lambda_i(M)\end{bmatrix}_{s\times s}$$

and hence

$$J(\lambda_i(M)) = \lambda_i(M)I_{s\times s} + N$$

where N is a nilpotent matrix of order s. Consequently, for $i \geq 2$

$$\begin{aligned} \lim_{t\to\infty} e^{J(-\lambda_i(M))t} &= \lim_{t\to\infty} e^{(-\lambda_i(M)I-N)t} \\ &= \lim_{t\to\infty} e^{-\lambda_i(M)It}e^{-Nt} \\ &= 0. \end{aligned} \tag{8.7}$$

Hence,

$$\lim_{t\to\infty} e^{-Mt} = X \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} X^{-1} = X_1 Y_1^T.$$

Finally,

$$\mathbf{x}^* = \lim_{t\to\infty} x(t) = \lim_{t\to\infty} e^{-Mt}\mathbf{x}_0 = X_1 Y_1^T \mathbf{x}_0 = (Y_1^T \mathbf{x}_0)X_1,$$

where $X_1 \propto \omega$ and $Y_1 \propto \mathbf{1}$ are the first right and left eigenvectors of the matrix M respectively and $\mathbf{1} = [1,\ 1,\ \dots, 1]^T$, the all-one vector. Therefore, from Eq. (8.7) it follows that the convergence rate of the system depends on the minimum non-zero eigenvalue of the matrix M that is $\lambda_2(M)$. □

Now we show that the rate of convergence of the proposed protocol is slower than that of the standard agreement protocol, that is, $\lambda_2(L) \geq \lambda_2(M)$.

Theorem 8.2.3. *Let N be a connected network corresponding to a MAS on n agents. Then*

$$\lambda_2(L) \geq \lambda_2(M).$$

Proof. Let $\mathbf{v}$ be a right eigenvector corresponding to an eigenvalue $\lambda(M)$ of $M = L\Omega^{-1}$. Then,

$$L\Omega^{-1}\mathbf{v} = M\mathbf{v} \Rightarrow L\Omega^{-1}\mathbf{v} = \lambda(M)\mathbf{v} \Rightarrow L\Omega^{-1}\mathbf{v} = \lambda(M)\Omega\Omega^{-1}\mathbf{v}$$

$$\Rightarrow L\mathbf{y} = \lambda(M)\Omega\mathbf{y} \text{ and } \Omega^{-1}\mathbf{v} = \mathbf{y}$$

$$\Rightarrow \Omega^{-1}L\mathbf{y} = \lambda(M)\mathbf{y} \Rightarrow By = \lambda(M)\mathbf{y} \text{ and } B = \Omega^{-1}L.$$

Thus the above calculation shows that matrices M and B share the same eigenvalues. Let $\mathbf{z}$ be a unit eigenvector corresponding to the eigenvalue $\lambda_2(L)$ (second smallest eigenvalue) of L. Consider $B^TB = L\Lambda^2L$ where $\Lambda = \Omega^{-1}$. Therefore, both the matrices B^TB and L share the same smallest eigenvalue and the corresponding eigenvector (all-one vector).

Now,

$$\begin{aligned} \mathbf{z}^TB^TB\mathbf{z} &= \mathbf{z}^TL\Lambda^2L\mathbf{z} \\ \Rightarrow \mathbf{z}^TU\Sigma U^T\mathbf{z} &= (\lambda_2(L))^2\mathbf{z}^T\Lambda^2\mathbf{z}, \end{aligned} \tag{8.8}$$

where $U\Sigma U^T$ is the singular value decomposition of the matrix B^TB and $\Sigma_{i,i} = (\lambda_i(M))^2$. We know that B^TB and L share the same eigenvector $\mathbf{u}_1 = [1, 1, ..., 1]^T$ corresponding to the zero eigenvalue (smallest eigenvalue). Then it is evident that the vector $\mathbf{z}$ can be written as the linear combination of singular vectors $\mathbf{u}_2,\ \mathbf{u}_3\ ...\ \mathbf{u}_n$. Hence

$$\mathbf{z} = \sum_{i=2}^{n} b_i \mathbf{u}_i = \sum_{i=1}^{n} b_i \mathbf{u}_i,\ b_1 = 0$$

which implies

$$\mathbf{z} = U\mathbf{b}.$$

where $\mathbf{b} = [0,\ b_2\ ,...,b_n]^T$ and $U = [\mathbf{u}_1,\ \mathbf{u}_2,...,\mathbf{u}_n]$. Thus from Eq. (8.8) we obtain

$$\mathbf{b}^T U^T U \Sigma U^T U \mathbf{b} = (\lambda_2(L))^2 \mathbf{z}^T \Lambda^2 \mathbf{z}.$$

Since $U^T U = UU^T = I$, the identity matrix, it follows that

$$\mathbf{b}^T \Sigma \mathbf{b} = (\lambda_2(L))^2 \mathbf{z}^T \Lambda^2 \mathbf{z}.$$

As B and M share the same eigenvalues,

$$\begin{aligned} \sum_{i=2}^{n} (b_i \lambda_i(M))^2 &= (\lambda_2(L))^2 \mathbf{z}^T \Lambda^2 \mathbf{z} \\ \Rightarrow \mathbf{b}^T \mathbf{b}\, (\lambda_2(M))^2 &\leq (\lambda_2(L))^2 \mathbf{z}^T \Lambda^2 \mathbf{z}. \end{aligned} \tag{8.9}$$

Recall that $\mathbf{z}$ is a unit vector and hence $\mathbf{b}^T \mathbf{b} = 1$. Then from Eq. (8.9) we obtain

$$(\lambda_2(M))^2 \leq (\lambda_2(L))^2 \sum_{i=1}^{n} \frac{z_i^2}{\omega_i^2} \leq (\lambda_2(L))^2 \sum_{i=1}^{n} \frac{z_i^2}{\omega_{\min}^2}$$

which establishes that

$$\lambda_2(M) \leq \frac{\lambda_2(L)}{\omega_{\min}}$$

where $\omega_{\min} = \min_i \omega_i \geq 1$. Thus the desired result follows. □

Now we come to the main result for static networks in this chapter for the detection of unknown failure links which affect the dynamics of diffusion on a network associated with a MAS. Here we consider $\omega_i = d_i$, the degree of the i^{th} node, and hence $\Omega = D$, the degree matrix.

Theorem 8.2.4. *Consider the diffusion dynamics given by Eq. (8.4) by setting $\Omega = D$, the degree matrix of the corresponding connected network on n agents. Let $\boldsymbol{x}^*$ be the final state of the system after the execution of the process. Then either of the following is true.*

1. *A link failure is detectable by at least $n-1$ observers, if each node is attached with an observer, or*
2. *After the convergence, a failure of the link (i, j) increases the values of $x_k \in \boldsymbol{x}^*$ at every nodes where $k \notin \{i, j\}$, and at least one of $x_i \in \boldsymbol{x}^*$ or $x_j \in \boldsymbol{x}^*$ reduces. The increment in the value of the converging state, $\boldsymbol{x}^*$, is proportional to the degree of the nodes.*

Proof. Let $\mathbf{x}_0 \in \mathbb{R}^n$ be the initial state vector which represents the initial amount of resources available with the agents in the MAS. Let $\mathbf{d} = [d_1,\ ,d_2,\ ,...\ d_n]^T$ where d_i is the degree of the i^{th} node, $i = 1, \ldots, n$.

Let $q(t) = \mathbf{1}^T \mathbf{x}(t)$. Then,

$$\dot{q}(t) = \frac{d}{dt}\mathbf{1}^T \mathbf{x}(t) = \mathbf{1}^T \dot{\mathbf{x}}(t) = \mathbf{1}^T LD^{-1}\mathbf{x}(t) = 0$$

since $\mathbf{1}^T L = 0$. Hence, $q(t) = q$ is an invariant quantity in the network. Thus, for any $t \geq 0$,

$$\mathbf{1}^T \mathbf{x}(t) = \mathbf{1}^T \mathbf{x}_0 = \mathbf{1}^T \mathbf{x}^* \propto \mathbf{1}^T \mathbf{d}. \tag{8.10}$$

as follows from Theorem 8.2.2, where $\mathbf{x}^*$ represents the state at which the diffusion process converges and $\omega = \mathbf{d}$.

Now assume that a link (i, j) is either removed or not working or failed (we consider all the three cases as failed), then the diffusion will happen under remaining network $G \setminus (i, j)$ which has a degree vector $\mathbf{d}_1 = \mathbf{d} - \mathbf{1}_{i,j}$ where $\mathbf{1}_{i,j}$ is the vector of ones at positions i and j, and rest of the entries of $\mathbf{1}_{i,j}$ are zeros. Consequently,

$$\mathbf{1}^T \mathbf{x}(t) = \mathbf{1}^T \mathbf{x}_0 = \mathbf{1}^T \overline{\mathbf{x}}^* \propto \mathbf{1}^T \mathbf{d}_1. \tag{8.11}$$

where $\overline{\mathbf{x}}^*$ is the new converging state of the diffusion process which begins with the same initial state vector $\mathbf{x}_0$. From the Eqs. (8.10) and (8.11),

$$\mathbf{1}^T \mathbf{x}^* = \mathbf{1}^T \overline{\mathbf{x}}^* \Rightarrow \mathbf{1}^T \mathbf{d} = \beta \mathbf{1}^T \mathbf{d}_1.$$

where $\beta > 0$ is a proportionality constant. Thus we obtain

$$\mathbf{1}^T (\mathbf{d} - \beta \mathbf{d}_1) = 0 \Rightarrow \mathbf{1}^T (\mathbf{d} - \beta \mathbf{d} + \beta \mathbf{1}_{i,j}) = 0$$

which implies

$$\mathbf{1}^T (1 - \beta) \mathbf{d} + \beta \mathbf{1}^T \mathbf{1}_{i,j} = 0.$$

If possible, let $\beta < 1$. Then,

$$\mathbf{1}^T (1 - \beta) \mathbf{d} + \beta \mathbf{1}^T \mathbf{1}_{i,j} > 0$$

which contradicts the previous equality and hence $\beta > 1$. This implies $x_k^{'*} > x_k^*$ where $x_k^{'*}$ and x_k^* denote the kth entry of the vectors $\overline{\mathbf{x}}^*$ and $\mathbf{x}^*$ respectively and $k \notin \{i, j\}$ where ith and jth nodes are incident to the failure link.

To make $\mathbf{1}^T (1 - \beta) \mathbf{d} + \beta \mathbf{1}^T \mathbf{1}_{i,j} = 0$, at least one negative change would happen in x_i^* and x_j^*. In this new state of convergence $\overline{\mathbf{x}}^*$, we notice that at least $n - 1$ changes with respect to $\mathbf{x}^*$ which is due to one link failure at (i, j). Consequently, the defined diffusion protocol is able to detect the link failure from at least $n - 1$ observers. □

We mention that since the increment in the amount of the resource at the node k which is not the part of the failed link is proportional to the degree of that node, the diffusion under this protocol has benefit to shift the resources from the nodes which are the part of the failed link to other nodes in the network according to the connectivity of the nodes (degree). Thus we conclude that the change in the converging state works as an indicator of link failure, and hence without sending a message regarding the link failure explicitly to the functional nodes the designer can detect the failure of a link in the network.

Next we investigate the diffusion dynamics in a network with switching (or dynamic) topology.

8.2.2 Diffusion on Dynamic Networks

In the previous subsection we have defined a diffusion protocol for static MASs. In this section, we define a linear diffusion process for dynamic networks in which the nodes changes their adjacency relationships with time. Indeed we assume that the network never gets disconnected at any time instant.

Let G_t be a connected dynamic network on n nodes which changes with time $t \geq 0$ and L_t the combinatorial Laplacian matrix associated with G_t. Similar to the case of static networks, we associate a weight $\omega_i \geq 1$ to the ith node and constitute the matrix $\Omega = \text{diag}\{\omega_1, \ldots, \omega_n\}$ which is a symmetric positive definite matrix. Then we introduce a diffusion process on the dynamic network associated with a MAS as

$$\dot{\mathbf{x}}(t) = -L_t\Omega^{-1}\mathbf{x}(t), t > 0, \mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^n \tag{8.12}$$

where $\mathbf{x}(t)$ is the state vector at time t as mentioned before. Then the concerned system which governs by the Eq. (8.12) converges to a vector which belongs to the null space of the matrix $L_t\Omega^{-1}$ spanned by the vector $\omega = [\omega_1, \omega_2, \ldots, \omega_n]^T$ [Mesbahi and Egerstedt, 2010]. Consider a weak Lyapunov function

$$V(\mathbf{x}(t)) = \frac{1}{2}\mathbf{x}^T(t)\Omega^{-1}\mathbf{x}(t)$$

and hence

$$\dot{V}(\mathbf{x}(t)) = -\mathbf{x}^T(t)\Omega^{-1}L_t\Omega^{-1}\mathbf{x}(t).$$

Note that for any vector $\mathbf{x}(t) \neq 0$, $V(\mathbf{x}(t)) > 0$ since Ω^{-1} is a positive definite matrix, and $\dot{V}(\mathbf{x}(t)) \leq 0$ since $\Omega^{-1}L_t\Omega^{-1}$ is symmetric positive semidefinite which confirm the asymptotic stability of the system. Hence the system given by Eq. (8.12) converges as $t \to \infty$ and the converging state belongs to the set of vectors spanned by the vector ω because the null space of the matrix $L_k\Omega^{-1}$ is independent of t, see chapter 4, section 4.1.1 in [Mesbahi and Egerstedt, 2010]. Indeed by considering the set

$$\Gamma_n = \{G : G \text{ is a connected network on } n \text{ nodes}\}$$

so that for any $t > 0$, $G_t \in \Gamma_n$, an alternative proof can be derived similar to the proof of Theorem 9 in [Olfati-Saber and Murray, 2004]. Moreover the convergence rate is less than

$$\max_{t>0} \lambda_2(L_t\Omega^{-1}).$$

At the steady state of the diffusion in a dynamic network, the distribution of resources in the network only depends on ω, it provides a way to distribute resources according to the requirements of underlying applications without being dependent over the topology of the network.

Note that the solution of the diffusion protocols defined by Eqs. (8.4) and (8.12) depend on $\omega = \text{diag}(\Omega)$. Also observe that if the entries of the matrix Ω is formulated by using structural or spectral properties of the given network then the diffusion pattern of resources over the network depends on the topological structure of the network. This freedom of selecting the matrix Ω can provide a wide range of applicability of the defined diffusion protocol in real-world complex multi-agent systems. Some of them are discussed in the next section.

8.3 APPLICATIONS

In the previous section, we have discussed the dynamics of a diffusion protocol which is applicable in static as well as dynamic networks. The converging state of this diffusion process depends on the matrix Ω which is not unique and can be chosen based on the underlying application. This dependency can be utilized in many practical applications. If the selection of the matrix Ω depends on structural properties of the network, it works as a detector of structural changes in the network. In the opposite direction, if we need a very specific fixed result, independent of the structural dynamics of the network, then the matrix Ω should be selected according to that given fixed point without considering the structural properties of the network. In this case, we know the converging state in advance although the path to reach there is unknown. In this section, we discuss three immediate applications of this diffusion protocol,

1. Link failure detection,

2. Security and utilization of resources, and

3. Static fixed point convergence over a dynamic network.

Next we discuss how the proposed diffusion dynamics can be utilized to develop the above three applications in a connected network.

8.3.1 Link Failure Detection

In many real world networked systems, detection of irregularity or faults in the system is one of the prime requirements. A small fault or error in the system may lead to a disastrous situation. For example, failure of major parts of the system can result in cascading failures in the networked systems. To avoid such situations, we need a proper solution that is able to detect even a small fault in the system. In this direction, we utilize the novelty of the diffusion protocol defined in the previous section. The solution of the diffusion Eq. (8.4), $\mathbf{x}^*$, depends on the vector $\boldsymbol{\omega}$. If $\boldsymbol{\omega} \propto \mathbf{d}$, then the solution vector, $\mathbf{x}^*$, depends on the degree vector, $\mathbf{d}$, of the networked multi agent system. Failure of a single link changes the orientation of the degree vector of the remaining networked multi-agent system ($\mathbf{d}_1$) that is still working. It shifts the solution of the diffusion protocol given by Eq. (8.4). Let $\overline{\mathbf{x}}^*$ be the new solution. It will be different from the previous one because of the dependency of matrix Ω over the degree of the nodes in the network. From the Theorem 8.2.4, the change in the amount of resources at the nodes which are not the part of the failed link (i, j), $(\overline{\mathbf{x}}^* - \mathbf{x}^*)_{-ij} \propto (\beta - 1)\mathbf{d}$, where $\beta > 1$ and the difference of resources, $(\overline{\mathbf{x}}^* - \mathbf{x}^*)_{ij}$, is negative in the nodes i or j. It can be noted that the network is static before link-failure. Therefore, Theorem 8.2.4 also holds. The difference in solutions $\mathbf{x}^*$ and $\overline{\mathbf{x}}^*$ is noticeable at the nodes which are not parts of the failed link. In the networked multi-agent system of size n, at least $n - 1$ observers, which are associated with $n - 1$ nodes of the network, would be able to detect a link failure in the system.

Convergence dynamics for link failure detection – If the network do not use a diffusion protocol for link failure detection, the nodes which are connected with the failed link can detect it immediately although others can not. They have to wait until they get an error message initiated by the nodes which are parts of failed link, which may take some additional time. However, in case of applying the diffusion protocol for link-failure detection, a node $k, \neq \{i, j\}$ can be able to detect a link-failure without concern with other nodes. In standard broadcasting of error messages, nodes need to transfer the message to other nodes. All nodes are involved in message propagation. In the proposed diffusion protocol, no additional message needs to be forwarded to other nodes regarding the link failure. It is inbuilt in the diffusion dynamics. The maximum time steps it takes is equal to the diameter of the network, while the normal broadcast of failure messages can take time-steps more than the diameter of the network to inform all the nodes about the event of link-failure. The lack of information about the failure prevents other nodes to take appropriate actions, and therefore link failure can cascade in the network. However, using the diffusion protocol for link failure detection, nodes which are not the part of the failed link, can detect the fault due to the shift in converging state $\mathbf{x}^*$ immediately after the link failure and can take necessary actions. Fluctuation in the steady state after convergence can be an indicator of the link failure. This way we don't have to wait until the next convergence to new steady state after link failure.

Under this application, protocol runs always because link failure event can not be predetermined. It can be noted that the dynamics of each node depends on its immediate neighbors only. So a node $k, \neq \{i, j\}$ would only be able to detect an event of link-failure, not the exact location of failure. To get more information regarding the exact location of link-failure, we need a centralized monitoring system that would not be a part of this network. The communication delay through the centralized monitor can not be avoided. It may cause the serious damage to the network due to failure cascading. Decision taken by a node in the networked system, on the basis of the fluctuation in the steady state of the node (converging state) can reduce the possible failure cascading and hence, communication delay can be avoided.

Link failure detection in dynamic networks –In the case of dynamic networks, if the degree of the each node remains constant during the diffusion under Eq. (8.12), the failure of a link can be detected using the same structure of the matrix Ω as discussed earlier. Diffusion in dynamic network under Eq. (8.12) also converges to the vector ω. If we consider $\Omega = D$, then it will converge to vector $\mathbf{d}$. Similar to the case of diffusion in static network, the failure of a link triggers a change in the converging state, and the outcome will be same as discussed for static network structure. At least $(n-1)$ nodes would be able to detect a link failure.

8.3.2 Security and Utilization of Resources

Underutilization and loss of resources is another problem that can be addressed through the proposed diffusion protocol. Resources, for example, data, power, information etc., are important assets in this real-world. Under many different circumstances, these resources are propagated through a networked system. In the networked system, some locations could be of more capacity and more secure, whereas some others could be of less capacity and less secure. Under this scenario, we need a priority in the distribution of the resources. Flexibility in the selection of matrix Ω can be applied to obtain desired outcome of resource distributions. We can set the value of ω_i according to predefined priorities of the nodes in a network. Higher priorities can be set to more secure or needy nodes to reduce the insecurity or underutilization of the resources distributed in the network. It is known as prioritized distribution of resources where priority can be the trustworthiness of a node.

In this diffusion protocol, when a link stops working and $\omega \propto \mathbf{d}$, some amount of resources at the nodes, which are the part of the failed link, get shifted to other nodes in the connected network. Let $\mathbf{x}^*$ and $\bar{\mathbf{x}}^*$ be the converging state of the diffusion process under the defined diffusion protocol in the network before the failure of a link and after link failure respectively. From the Theorem 8.2.4, the change in the amount of resources at the nodes which are not the part of failed link (ij), $(\bar{\mathbf{x}}^* - \mathbf{x}^*)_{-ij} \propto (\beta - 1)\mathbf{d}$, where $\beta > 1$ and the difference of resources, $(\bar{\mathbf{x}}^* - \mathbf{x}^*)_{ij}$, is negative in the nodes i or j. This phenomena shifts the resources from the nodes which are involved in the failed link to the other nodes in the network. This improves the security and utilization of the resources in the networked multi-agent system. It can be noted that this architecture is different from developing a fault-tolerant system architecture. Fault-tolerance is related to the working of the network after a damage, but here we talk about the automated transfer of the resources after the link failure, from the damage part of the network to the rest of the network. This means that the diffusion protocol can secure the resources. It is applicable in both the types of networks, static and dynamic connected networks.

As we know that diffusion under Eq. (8.12) converges to vector ω. If we consider a dynamic network which maintains the connectivity, degree of the nodes can change during the diffusion such that $\Omega = D$. Then it might be possible that the diffusion does not converge to any state vector. However, as the connectivity of a node reduces, the resources at that node shifts towards other well-connected nodes in the network according to the diffusion protocol given by Eq. (8.12). The rate of change in the pattern of connections affects the rate of shifting of the resources. Nevertheless, if the degree of each node is fixed, only the pattern of connections changes without affecting the connectivity. As a consequence, the failure of a link shifts resources to other parts of the network as we have discussed in case of a static network. Under the given fixed degree vector of the network, dynamic behavior of the network structure does not affect the converging vector. It can be affected by only loosing the connectivity (degree) of a node during the diffusion process. Thus under the above conditions, diffusion dynamics can be used to detect the damage (link failure) in the network.

8.3.3 Static Fixed Point Convergence Over a Dynamic Network

In this section, we consider a dynamic network and the distribution of resources depends on fixed property of nodes, not on the pattern of connections. In the previous application, we have analyzed how a link failure can shift the resources form one part of the network to the other part. In this application, we focus on the fixed distribution of the resources irrespective of connection pattern of the network. In the previous application, fixed prioritized distribution is obtained on the basis of utilization and security that depends on the connectivity of the network. However, in this case we have prioritized the distribution irrespective of the connections. A less connected node can get more priority so that it can get more resources before being disconnected from the rest of the world.

In Section 8.2.2, we have studied the convergence of the proposed diffusion protocol under dynamic networks. In this scenario, the diffusion protocol converges to vector $\mathbf{x}^*$ that depends on ω. In many applications, such as when we need the fixed distribution of resources in the networked multi-agent systems irrespective of the structure of the network, we can get required results after setting the vector ω according to the priorities of the nodes which are independent of the structure of the network. Therefore, the priorities of the nodes can be set in a way such that the ratio of the priorities between any two nodes and the ratio of the required resources between those two nodes are same. This way, we can get the predefined distribution of resources in the dynamic networked multi-agent system.

The case of a static network, which may undergo link failures without loosing the connectivity of the network, can be considered as the dynamic network. We can achieve the desired distribution of resources over the nodes of the network irrespective of link failures. The only required conditions are connectivity of the network and fixed structure of the matrix Ω that can define the pattern of the desired output of the diffusion. In this application, the protocol stops after some time, when the value of the resource at a node i, $x_i(t)$, does not change significantly. A threshold value, to compare the change in the value of $x_i(t)$, can be obtained to stop the diffusion protocol.

8.4 SIMULATIONS TO VALIDATE THE THEORY OF DIFFUSION PROTOCOLS

In this section, we provide the simulation results of the proposed diffusion process under the protocols developed in the section 8.2. Similar to the previous section, three data sets are considered to show the validation of theory under different conditions. The detailed observations are discussed next.

8.4.1 Data

We have used three networks to simulate the defined diffusion protocol and its applications under different scenarios. The data provides the network topology of the Internet network [Knight *et al.*, 2011] in which nodes represent geographical locations of Internet service providers,

Table 8.1 : Details of the Networks used for the simulation

Network	Nodes	Edges
RedBestel	84	93
Belcanada	48	64
Belsouth	51	66

customers or external transit providers; and links represent direct connectivities between these geographical locations [Knight *et al.*, 2011]. We have used simple connection matrix of these three considered networks which provide the connectivity among the different nodes only. The details of the networks are given in Table 8.1. We execute the proposed diffusion protocol over these network structures with static and link failure scenarios, and compute the changes in the converging states with respect to the time.

8.4.2 Simulated Cases

We have considered 5 cases as follows.

(I) Convergence over static networks with $\Omega = D$,

(II) Convergence over the dynamic networks with $\Omega = D$,

(III) Convergence of diffusion in static vs dynamics networks for $\Omega = D$,

(IV) Change of the Converging State of a Node due to Link Failure when $\Omega = D$,

(V) Convergence of diffusion in static vs dynamics networks for constant Ω.

Next we analyze the observations for these five cases.

Case I (Static Network with $\Omega = D$)

In this experiment, we consider the dependency of converging state vector over the structural property of the network when the network is static and $\Omega = D$. The proposed diffusion protocol is implemented over three static networks, as given in Table 8.1, with $\Omega = D$, . Multiple initial state vectors x_0 are considered to simulate the diffusion phenomena. The convergence of $x(t)$ with respect to the time t is plotted in Figs. 8.1(a) to 8.1(c). As we observe from the figure that the diffusion process converges to the same state vector $\mathbf{x}^* \propto \mathbf{d}$, independent of the initial state vector, where $\mathbf{d} = \text{diag}(D)$ is the degree vector of the considered network. It can be noted that the second smallest eigenvalue of the the three networks, that is Bellsouth, RedBestel and Bellcanada, are 0.053, 0.0052 and 0.0152 respectively. The Bellsouth network has larger $\lambda_2(M)$, and the RedBestel network has smaller value of $\lambda_2(M)$. In Figs. 8.1(a), 8.1(b), and 8.1(c), it follows the convergence rate of the diffusion process, which is faster in the Bellsouth network and slowest in the RedBestel network.

Case II (Dynamic Network with $\Omega = D$)

In this subsection, we have simulated the diffusion over the dynamic networks for $\Omega = D$. To impose the effect of dynamic network structure, at each time step we applied the same rewiring process as explained follows. We consider four random nodes i_1, j_1, i_2 and j_2 at each time step such that only edges (i_1, j_1) and (i_2, j_2) exist and there is no other edges among these considered nodes.

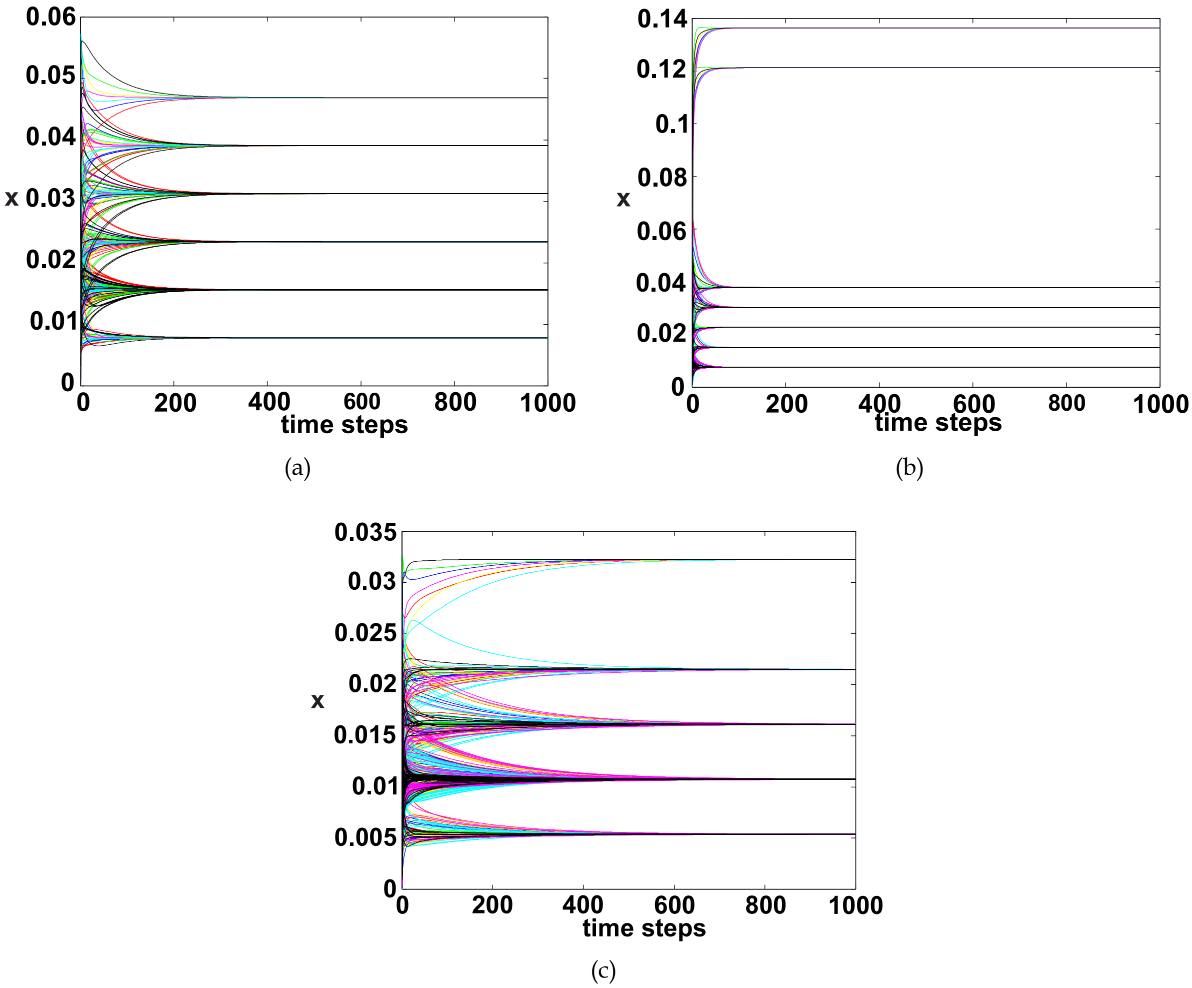


Figure 8.1 : Case I- State vector $\mathbf{x}(t)$ is plotted for the diffusion process in (a) Bellcanada (b) Bellsouth (c) RedBestel networks (static networks) after setting $\Omega = D$. Different colors in the plots are correspond to the diffusion started from different initial points, $\mathbf{x}(0)$. It shows the unique state convergence ($\mathbf{x}^* \propto \mathbf{d}$) of diffusion process in the considered networks while $\Omega = D$.

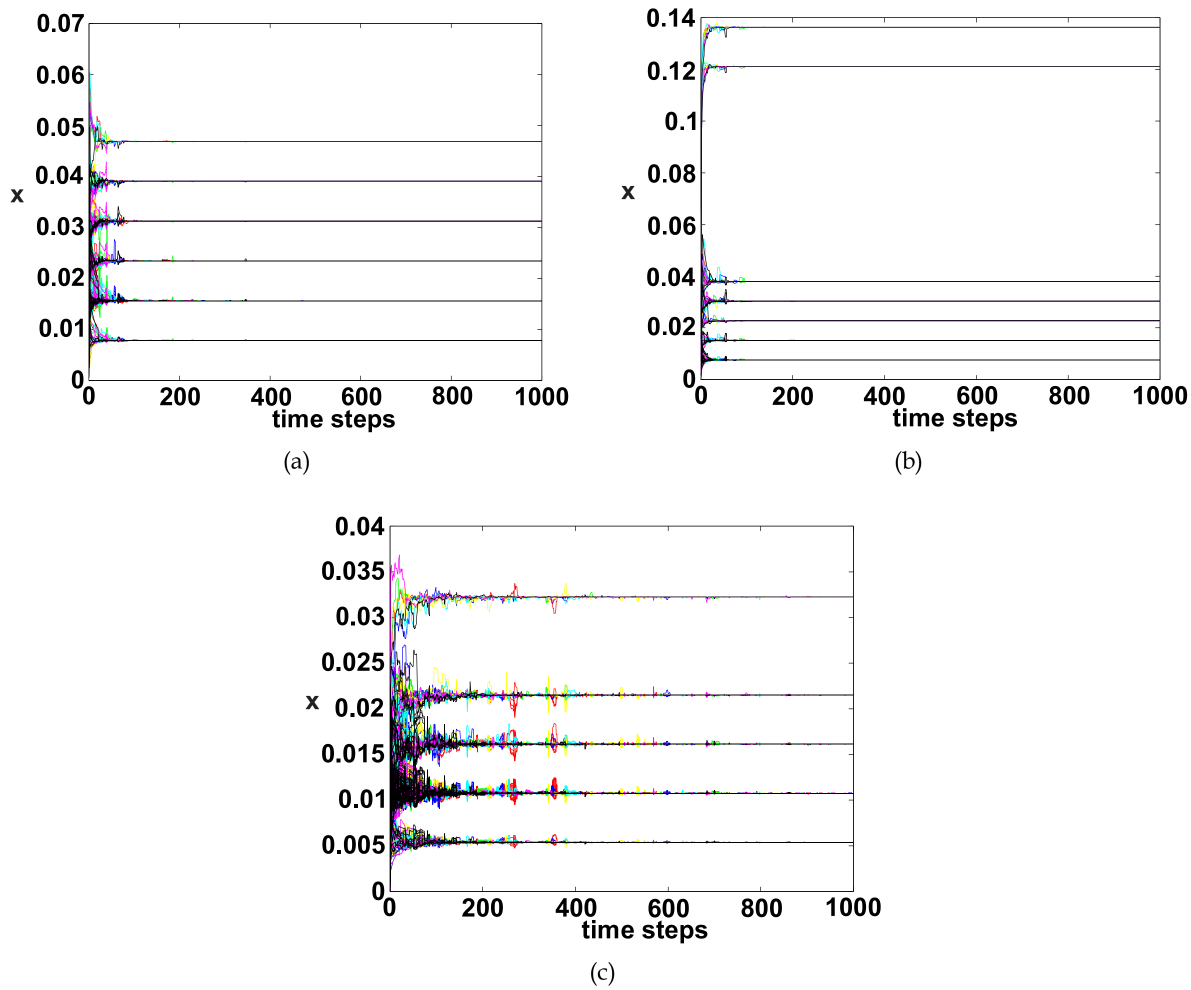


Figure 8.2 : Case II- State vector $\mathbf{x}(t)$ is plotted for the diffusion process in (a) Bellcanada (b) Bellsouth (c) RedBestel networks (dynamic networks) after setting $\Omega = D$. Plots in multiple colors are correspond to the diffusion started from different initial points, $\mathbf{x}(0)$. It shows the unique state convergence ($\mathbf{x}^* \propto \mathbf{d}$) of diffusion process in the considered networks while $\Omega = D$.

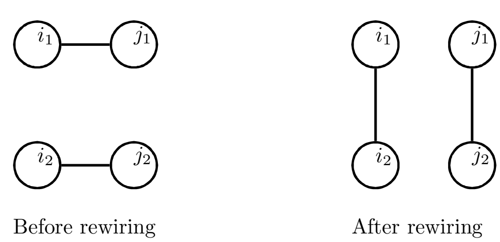


Figure 8.3 : Change in the Orientation of Links among the Considered Nodes during the Rewiring Process.

To consider the effect of dynamic structure of the network, the edges rewired keeping the degree of the nodes same. Graphical representation of the rewiring of the edges at each time step is given in Fig. 8.3.

It can be noted that the network should be connected during the simulation process, and therefore the rewiring procedure takes this into consideration while changing the links. The diffusion under such dynamic conditions is simulated considering multiple initial state vectors $\mathbf{x}_0$. Simulated results are shown in Figs. 8.2(a) to 8.2(c). It can be observed from the figure that the converging state vector $\mathbf{x}^*$ is unique for the given degree vector $\mathbf{d}$, because the rewiring of the edges does not affect the degree of the nodes, as we can observe from Figs. 8.2(a), 8.2(b), and 8.2(c). The rates of convergence of the diffusion in the networks are in the same sequences as discussed in Case I, with the similar explanation of its behavior.

Case III (Diffusion Trajectories of Static vs Dynamics Networks for $\Omega = D$)

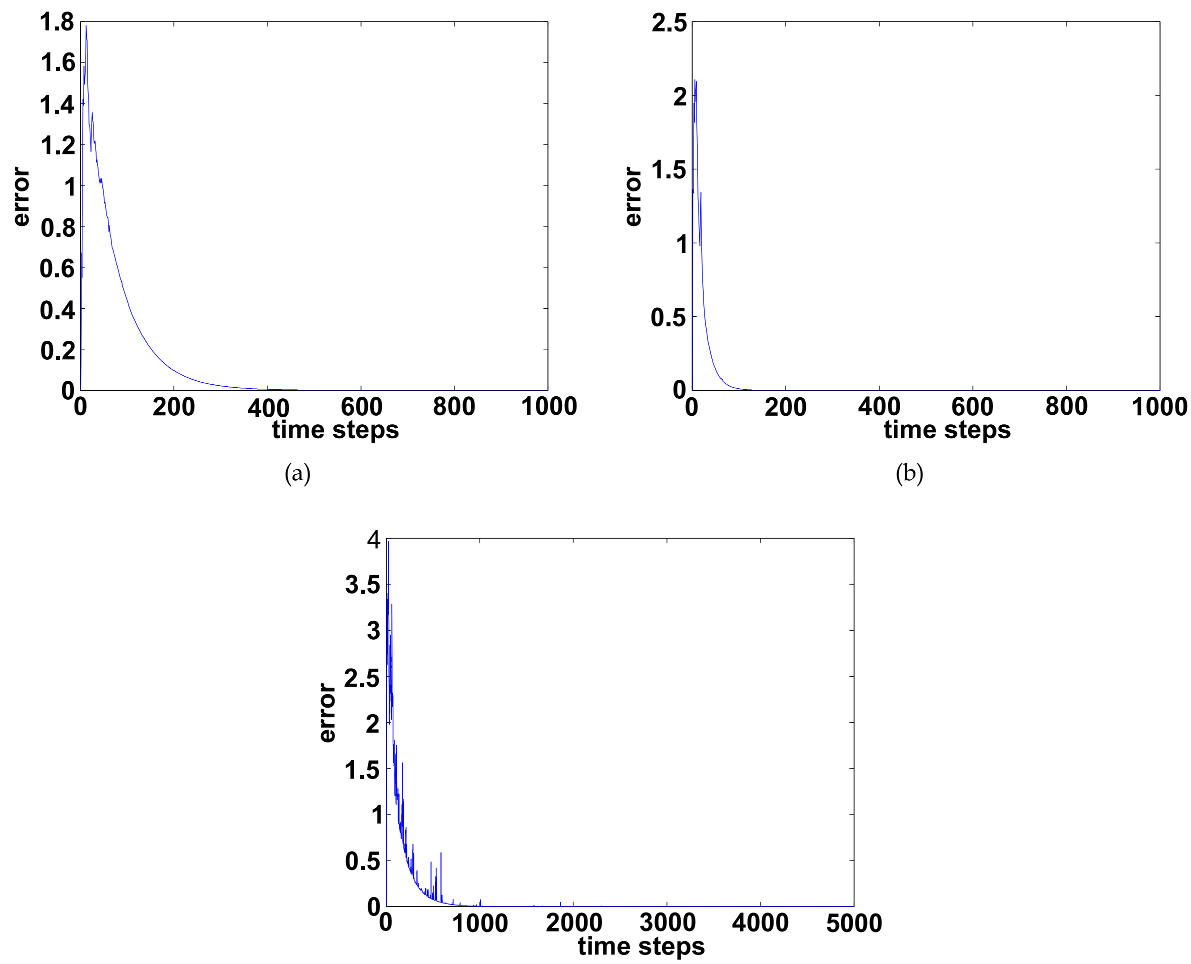


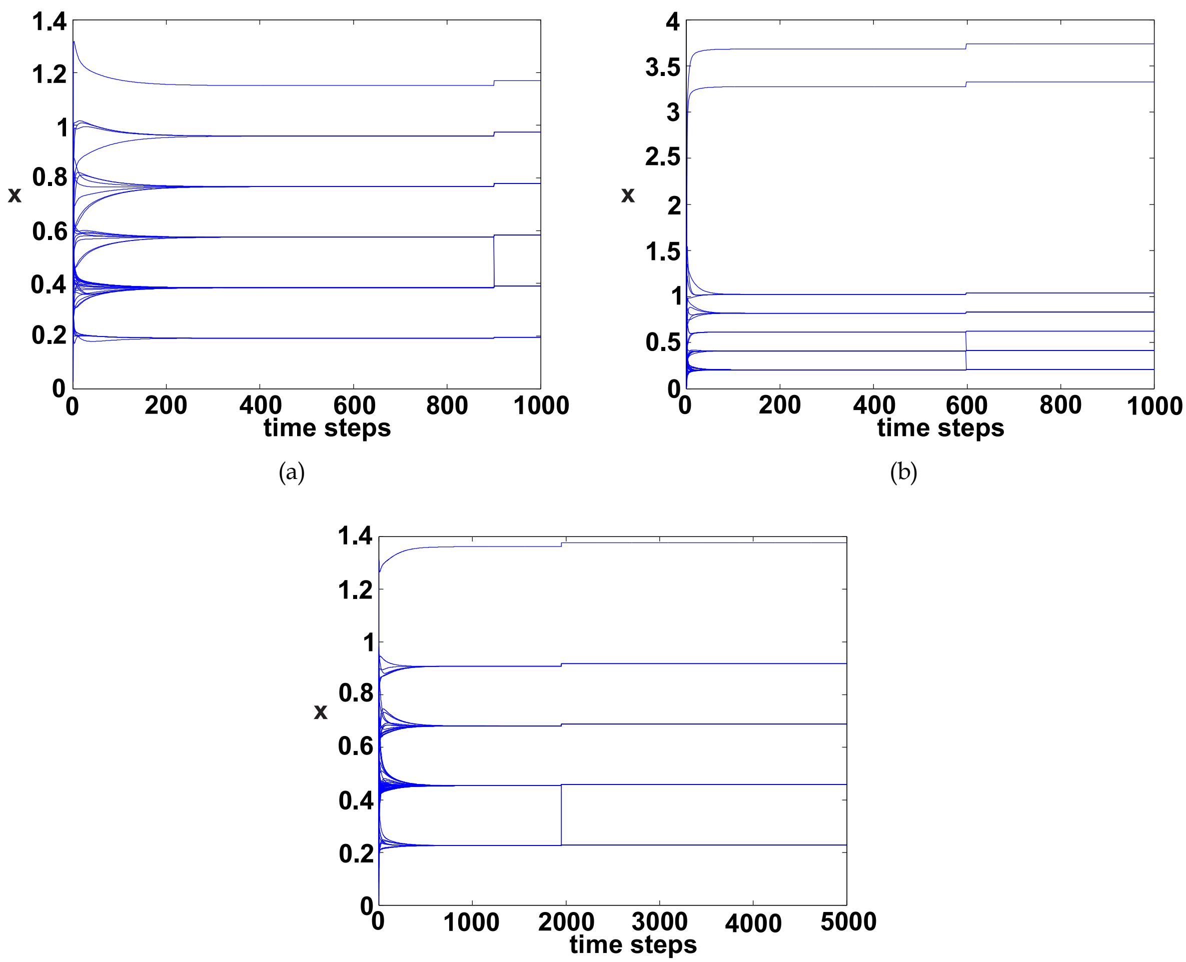
Figure 8.4 : Case III- After setting $\Omega = D$, Error in trajectories of diffusion in (a) Bellcanada (b) Bellsouth (c) RedBestel static networks and corresponding dynamic networks (incorporating rewiring) is shown. Diffusion started from different initial conditions, $\mathbf{x}(0)$.

In this subsection, we study the differences in diffusion trajectories between the diffusion processes over same network under a static scenario and a dynamic scenario. The proposed protocol is executed over a static network first, and then the same network is considered to simulate the diffusion in a dynamic scenario. We adopted the same rewiring process what we discussed in the previous case to consider the effect of dynamic network. Difference between the trajectories of the two processes is termed as *error* which is plotted in Figs. 8.4(a), 8.4(b), and 8.4(c). The measurement of *error* is defined as follows

$$error(t) = \sum_i |x_i(t) - x_i^{'}(t)|$$

, where $\mathbf{x}(t)$ is the state vector at time t of the diffusion process in the static network and similarly $\mathbf{x}'(t)$ is the state vector for the diffusion process in dynamic network. We can observe that for the constant degree vector $\mathbf{d}$, the converging state is unique, independent of the initial condition and the dynamic nature of the networks. Further, from Figs. 8.4(a), 8.4(b), and 8.4(c), as the t become larger *error* goes to zero.

Case IV (Change of the Converging State of a Node due to Link Failure when $\Omega = D$)



(a) (b) (c)

Figure 8.5 : Case IV- State vector $\mathbf{x}(t)$ of diffusion in (a) Bellcanada (b) Bellsouth (c) RedBestel networks is plotted. Link-failure during the diffusion process is captured by vertical sift in converging state $\mathbf{x}^*$.

Here we observe how a link failure during diffusion shifts the converging state. In sub-section 8.5.1, we have discussed how the link failure affects the converging sate of the diffusion process. For the similar condition $\Omega = D$ as considered in the previous cases, when a link fails during the diffusion process, a vertical shift in converging state $\mathbf{x}^*$ appears that is visible in Figs. 8.5(a), 8.5(b), and 8.5(c) In the simulation process, we deleted two links randomly without affecting the connectivity of the network. To observe the shifting in converging lines in Figs. 8.5(a), 8.5(b), and 8.5(c) , we deleted the two links after sufficient time. The same procedure is applied for the simulation of all the three networks which are mentioned in the Table 8.1. All the converging lines in time series plot of state vector $\mathbf{x}(t)$ show the vertical shift due to new converging state $\bar{\mathbf{x}}^*$ which appears after the deletion of links as discussed in sub-section 8.5.1. This is an indication of link failure. An intuitive justification of the same can be as follows. The nodes of the networks can be classified into groups according to their degree distribution. In this scenario, the nodes of a same group converges to the same value of $\mathbf{x}_i^*$ due to similar degree distribution. When a link fails, degree distribution of the nodes changes, and new groups are thus formed. This shifting in groups appears as the vertical shifting in the converging lines of the time series plot, as we can observe in Figs. 8.5(a), 8.5(b), and 8.5(c).

Case V (Diffusion Trajectories of Static vs Dynamics Networks for Constant Ω)

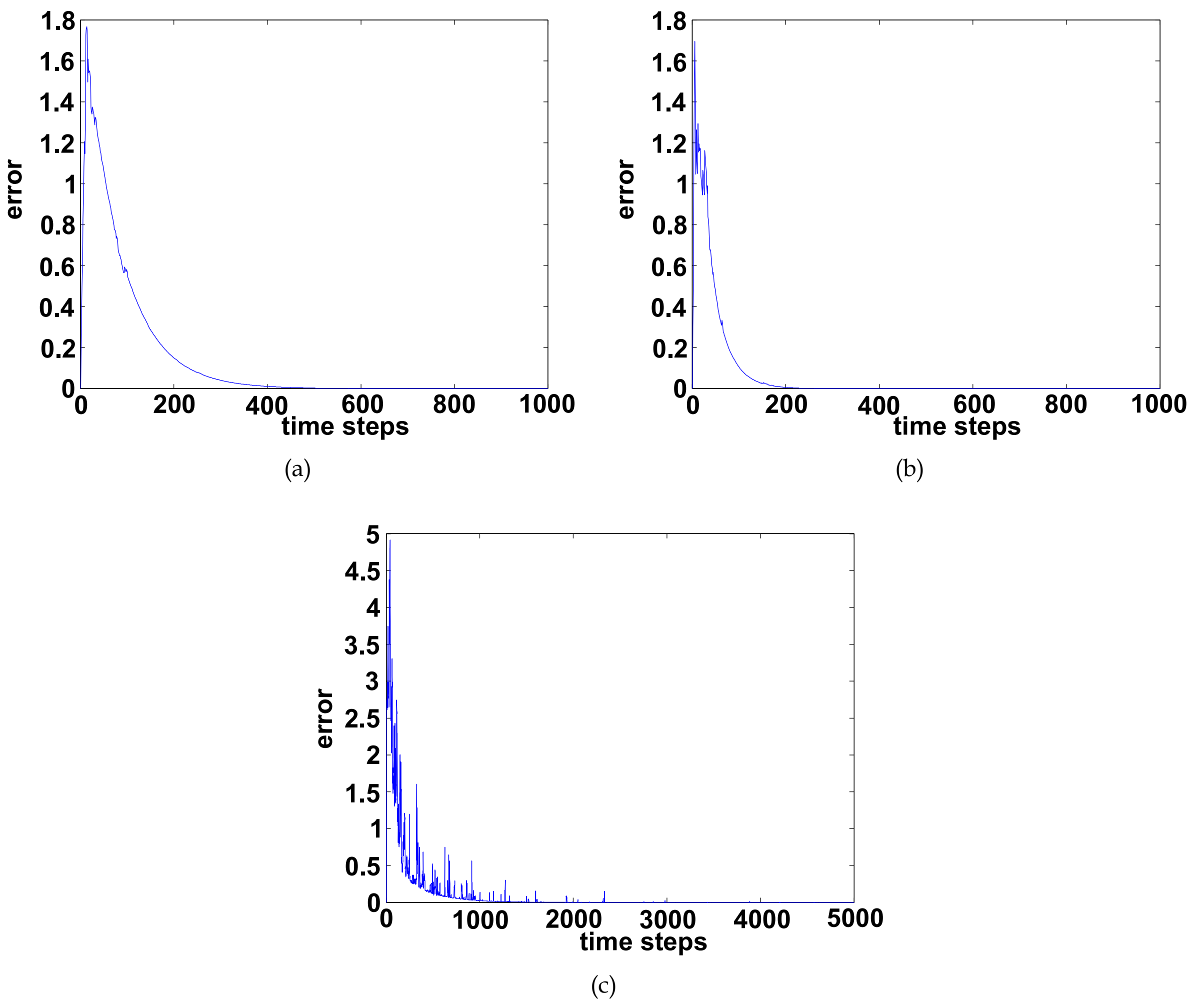


Figure 8.6 : Case V- For a constant matrix Ω which has non-zeros diagonal entries, difference ($error$) in trajectories of diffusion in (a) Bellcanada (b) Bellsouth (c) RedBestel static networks and corresponding dynamic networks (incorporating rewiring) are plotted. Diffusion started from different initial conditions, $\mathbf{x}(0)$.

In this subsection, we do the same experiment which is being carried out in Case III, except that now we consider a constant Ω instead of $\Omega = D$. We plotted the difference in the trajectories of the state vectors during the diffusion in the static network and the dynamic network for the fixed matrix Ω that is independent of the structural properties of the considered networks. Diffusion for the unique matrix Ω converges to the fixed converging state $\mathbf{x}^*$ independent of the structural changes in the networks. As time t become larger, $error(t)$ goes to zeros, as we can observe in Figs. 8.6(a), 8.6(b), and 8.6(c).

8.4.3 Summary of Observations and Salient Features of the Proposed Diffusion Protocol

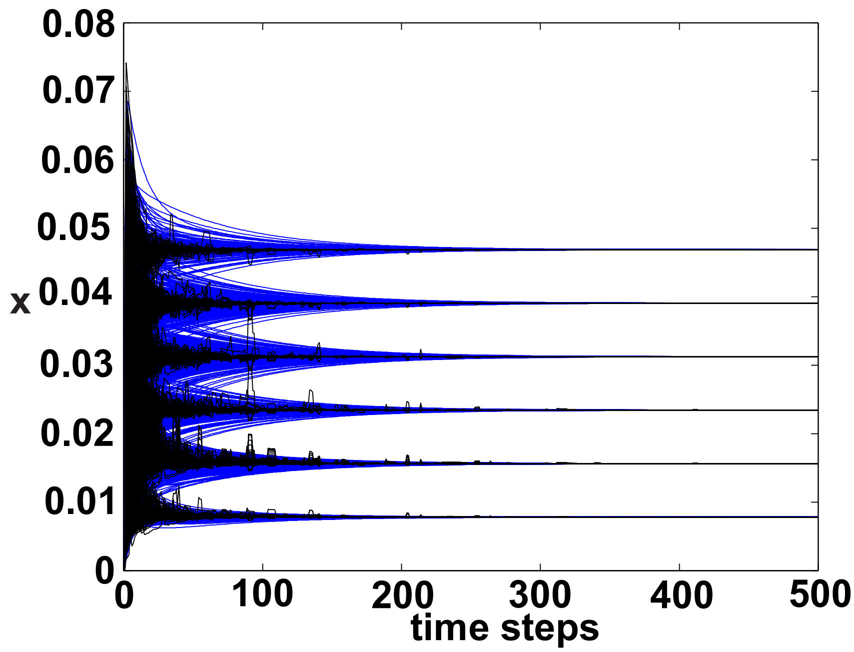


Figure 8.7 : Plots in blue color represents the diffusion in static network and plots in black color represents the diffusion in dynamic network that is corresponding to considered static network of Bellcanada.

It is observed that the convergence of the diffusion process in a static network is slower than the diffusion in a dynamic network corresponding to the same static network, as we can observe from Fig. 8.7. In Fig. 8.7, black lines represent the convergence of diffusion process in a dynamic network which has constant degree of the nodes. Similarly, blue lines represent the convergence of diffusion in the static network having same degree of nodes in the network as in the case of dynamic network. Plot in black color converges faster than blue one. To incorporate the dynamic nature in the structure of the network, we applied edge rewiring keeping the degree of the nodes same. Dynamic nature of the links creates paths of small lengths between a pair of nodes during the diffusion of the resources in the network. It promotes faster diffusion of resources in the network under the considered diffusion protocol. In static networks, paths are deterministic which cannot always facilitate short path length between the nodes that have uneven (different from the converging value of x_i^*) distribution of resources. In the previous section, we have discussed that the converging vector of the diffusion under the considered diffusion protocol depends on the matrix Ω that can either depends on structural properties, for example $\Omega = D$, of the considered network, or can be a predefined constant diagonal matrix which has the constant positive real numbers associated with the nodes in the network that can be interpreted as the priority of the nodes in the network. In the reverse engineering, if we assume that the diffusion in the network happens using the defined diffusion protocol, then the converging vector will represent the relative significance (centrality) of the nodes in the network. However, it is important to show the central position of the nodes during the diffusion phenomena.

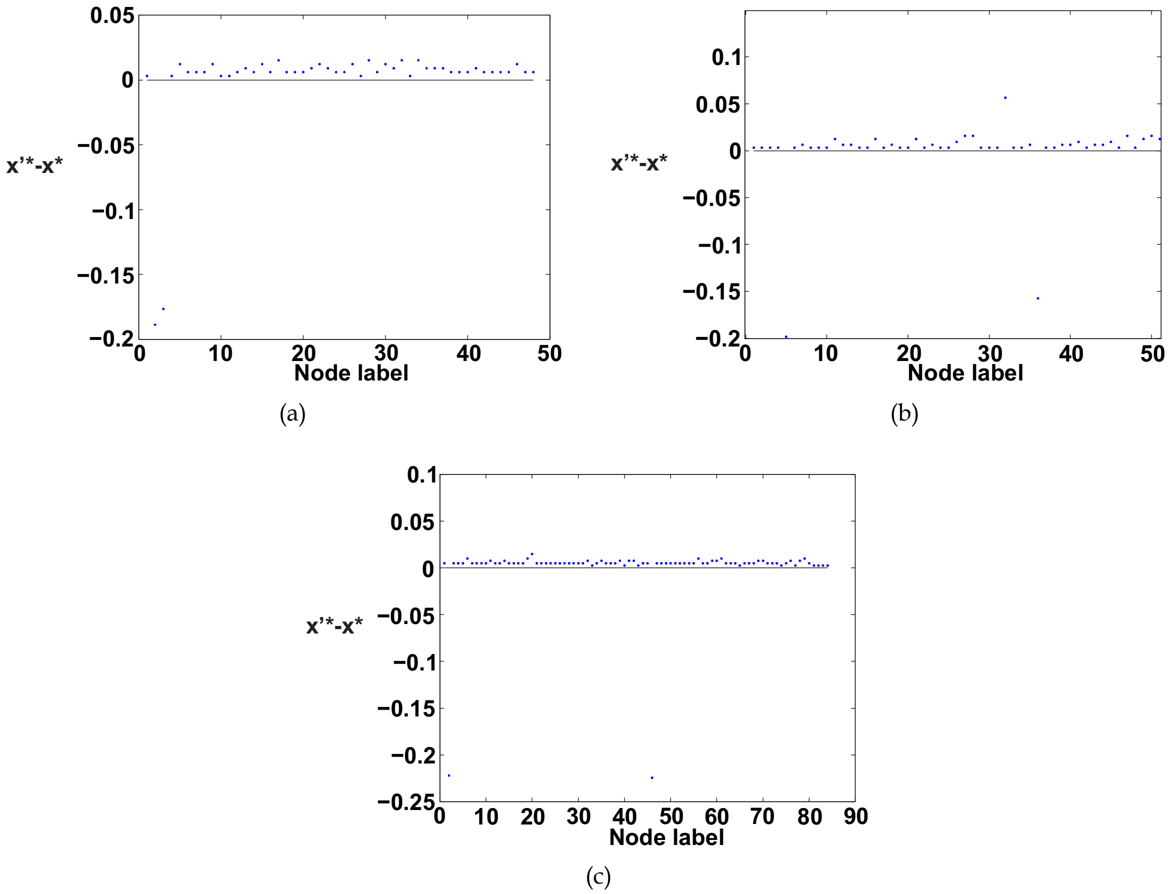


Figure 8.8 : Shifting in the Resources due to Link Failure. Plot shows the change in converging state, $(\overline{\mathbf{x}}^* - \mathbf{x}^*)$, after the deletion of a link in (a) Bellcanada (b) Bellsouth (c) RedBestel networks.

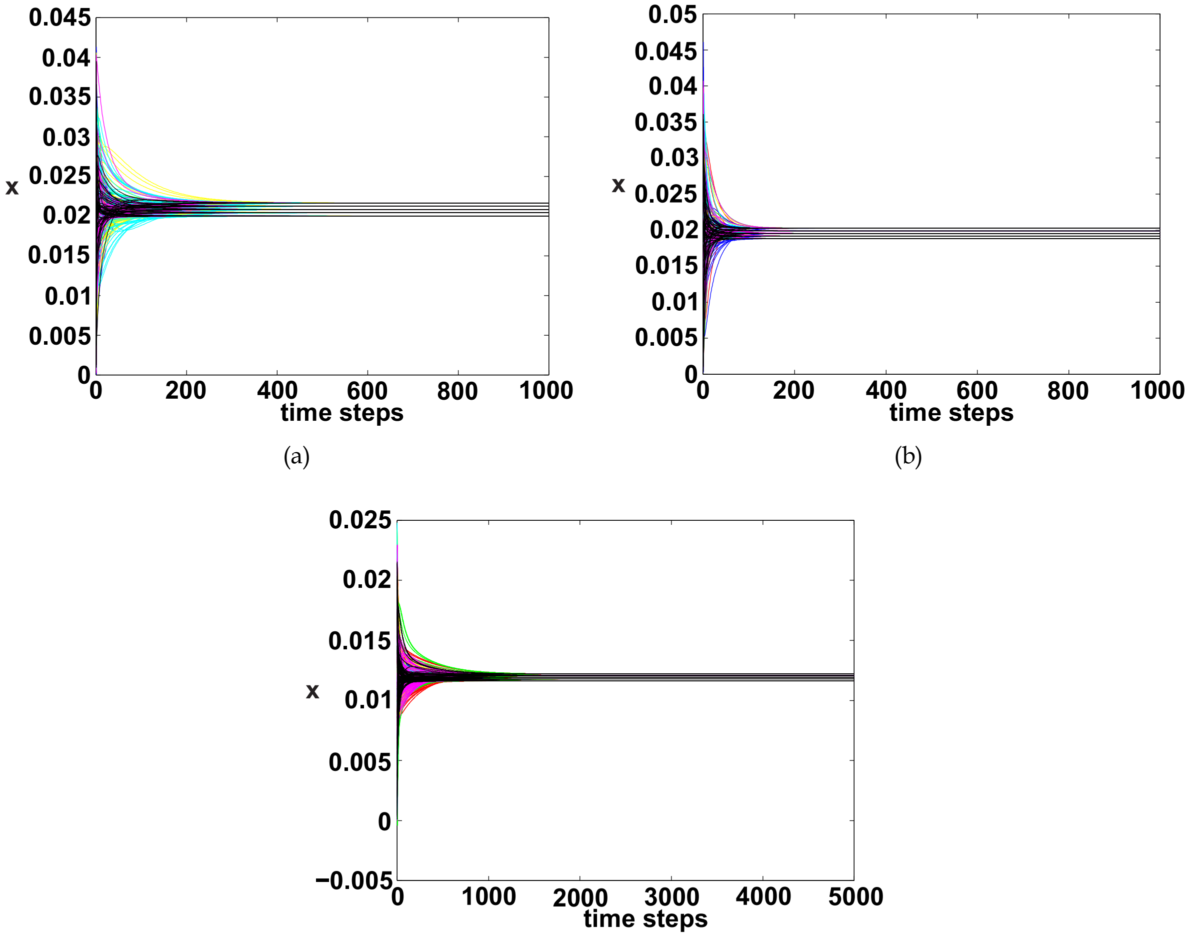
8.5 SIMULATIONS OF APPLICATIONS OF DIFFUSION PROTOCOL

In this section, we provide simulation results of applications developed in the section 8.3. Three different data sets are considered to show the validation of the proposed theory, as discussed in the previous section with Table 8.1.

8.5.1 Simulation of Diffusion With and Without Link Failures

In this case, we study the diffusion process in the static network with the consideration of $\Omega = D$. We have already discussed about the dependency of the converging state $\mathbf{x}^*$ over the structural property of the network if $\Omega = D$. Fist, we run the diffusion protocol over the three networks considering $\Omega = D$, after that a link is deleted from each network, and then again we execute the same protocol. Let us assume that $\mathbf{x}^*$ and $\overline{\mathbf{x}}^*$ are the two converging states, $\mathbf{x}^*$ is the converging state before the deletion of the link, and $\overline{\mathbf{x}}^*$ is the converging state after the deletion of the link. Deletion of the link changes the converging state because of the dependency of the converging state over the vector $\mathbf{d}$. We plot the change in converging states as shown in Figs. 8.8(a), 8.8(b), and 8.8(c). $(\overline{\mathbf{x}}^* - \mathbf{x}^*)$ is plotted in Figs. 8.8(a), 8.8(b), and 8.8(c) which shows that deletion of a link (i, j) decreases the amount of resources in the nodes i and j and increases resources at other nodes. After deletion of a link, the network must be connected. The non-zero values of $(\overline{\mathbf{x}}^* - \mathbf{x}^*)$ indicates that the failure of a link and shifting of resources from the affected part (nodes associated with failed link) to a more secure part of the network increase the security and utilization of the resources. The discussed simulated results verify the practicality of applications explained in sub-sections 8.3.1 and 8.3.2. Similar results are obtained for dynamic network structure also, as discussed in sub-sections 8.3.1 and 8.3.2.

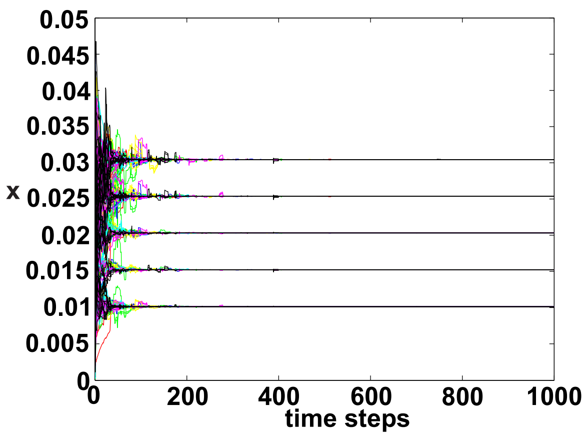
8.5.2 Simulation of Fixed Point Convergence Over the Static and Dynamic Network Structures



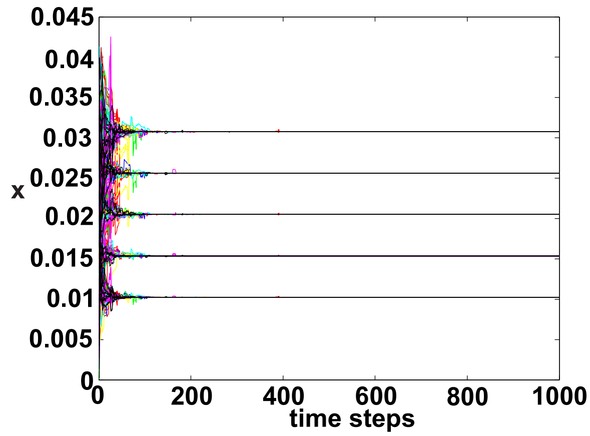
(a) (b) (c)

Figure 8.9 : Fixed point Convergence over the Static Network Structure. State vector $\mathbf{x}(t)$ is plotted for the diffusion process in (a) Bellcanada (b) Bellsouth (c) RedBestel networks (static networks) for a constant diagonal matrix Ω, having non-zeros diagonal entries. Plots in multiple colors are correspond to diffusion started from different initial points, $\mathbf{x}(0)$. It shows the convergence of diffusion process in (a) Bellcanada (b) Bellsouth (c) RedBestel networks.

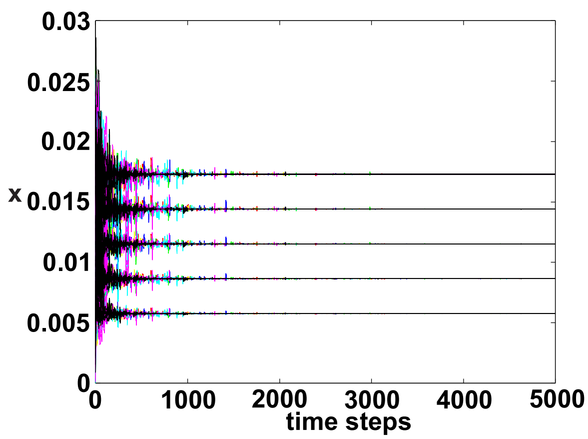
In this subsection, we study the diffusion in the networks when matrix Ω is a constant diagonal matrix which has non-zeros diagonal entries, independent of the structural properties of the network. To generate the matrix Ω, a random integer number is generated in the range of 1 to 5 corresponding to each entry of the diagonal of the matrix Ω. All the nodes of the network are grouped in 5 groups. We have performed the simulation in a static network, similar to the Case I as discussed in the previous section. The diffusion process converges to a unique state vector x^* which depends on the vector $\omega = \text{diag}(\Omega)$. Figs. 8.9(a), 8.9(b), and 8.9(c) show the evolution of the state vector $\mathbf{x}(t)$ with respect to the time steps. There are only five converging lines in the time series plot of the state vector $\mathbf{x}(t)$ in Fig. 8.9(a), 8.9(b), and 8.9(c), that corresponds to five group of nodes in the network according to matrix Ω.



(a)



(b)



(c)

Figure 8.10 : Fixed point Convergence over the Static Network Structure. State vector $\mathbf{x}(t)$ is plotted for the diffusion process in (a) Bellcanada (b) Bellsouth (c) RedBestel networks (dynamic networks) for a constant diagonal matrix Ω, having non-zeros diagonal entries. Plots in multiple colors are correspond to diffusion started from different initial points, $\mathbf{x}(0)$. It shows the convergence of diffusion process in (a) Bellcanada (b) Bellsouth (c) RedBestel networks.

Next, we consider the diffusion in dynamic networks for the fixed matrix Ω which is generated as discussed in the previous case. Diffusion phenomena is simulated for the multiple initial state vector $\mathbf{x}_0$ and constant matrix Ω considering dynamic network structure. We use the similar rewiring process (Fig. 8.3) as discussed in the previous section for dynamic networks. To make the situation more realistic, the additional link deletion process is imposed, however which is not very frequent during the simulation. The connectivity of the network is maintained throughout the diffusion process. The results are shown in Figs. 8.10(a), 8.10(b), and 8.10(c). In Figs. 8.10(a), 8.10(b), and 8.10(c), there are 5 converging lines in time series plot of the state vector $\mathbf{x}(t)$ which depends on the entries of the matrix Ω. In this case, deletion of the link does not affect the converging state vector $\mathbf{x}^*$. Using this diffusion protocol with fixed Ω, which is independent of the network structure, a fixed distribution of the resources in the network can be achieved irrespective of the dynamic changes in the structure of the network.

8.6 CONCLUSION

In this chapter, we proposed a diffusion protocol for networks with static and switching topologies. Diffusion process under the defined protocol converges to a desired solution depending on the priority of its nodes or the degree vector. Due to this dependency, the protocol has multiple practical applications as we have shown in the chapter. We show that it can be used to detect failure of links in networks, as well as to distribute the resources in the network according to the requirements, in spite of changes in the network topology. The effectiveness, convergence and stability of the proposed protocol have been analyzed through simulation results in different real life networks. To the best of our belief, the theoretical foundation laid by the proposed diffusion protocol can be utilized in the applications of dynamic and time varying networks, which we plan to explore in details as a future extension of this work.

…