An Efficient and Stable Congestion Control Scheme with Neighbor Feedback for Cluster Wireless Sensor Networks

Xi Hu[1], **Wei Guo**[1]

[1] National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, 611731, China

[e-mail: huxi027@163.com, guowei@uestc.edu.cn]

*Corresponding author: Xi Hu

Received March 4, 2016; revised June 20, 2016; accepted July 25, 2016; published September 30, 2016

Abstract

Congestion control in Cluster Wireless Sensor Networks (CWSNs) has drawn widespread attention and research interests. The increasing number of nodes and scale of networks cause more complex congestion control and management. Active Queue Management (AQM) is one of the major congestion control approaches in CWSNs, and Random Early Detection (RED) algorithm is commonly used to achieve high utilization in AQM. However, traditional RED algorithm depends exclusively on source-side control, which is insufficient to maintain efficiency and state stability. Specifically, when congestion occurs, deficiency of feedback will hinder the instability of the system.

In this paper, we adopt the Additive-Increase Multiplicative-Decrease (AIMD) adjustment scheme and propose an improved RED algorithm by using neighbor feedback and scheduling scheme. The congestion control model is presented, which is a linear system with a non-linear feedback, and modeled by Lur'e type system. In the context of delayed Lur'e dynamical network, we adopt the concept of cluster synchronization and show that the congestion controlled system is able to achieve cluster synchronization. Sufficient conditions are derived by applying Lyapunov-Krasovskii functionals. Numerical examples are investigated to validate the effectiveness of the congestion control algorithm and the stability of the network.

Keywords: Random Early Detection (RED), Additive-Increase Multiplicative-Decrease (AIMD), Lyapunov-Krasovskii functionals, cluster synchronization

This work is supported by the 863 project (Grant Nos. 2015AA01A705, 2014AA01A701); The National Natural Science Foundation of China (Grant Nos. 61271168, 61471104); Foundation of National Key Laboratory of Scien ce and Technology on Communications (Grant No. 9140C020302130C02007).

1. Introduction

Wireless Sensor Network (WSN) is comprised of a set of sensor nodes which monitor physical environment and collect information for further processing [1-3]. Due to increasing number of nodes and scale of WSNs, it is essential to organize clusters in the network to achieve more flexible distributed control and effective management [4].

Cluster WSN (CWSN) consists of large quantity of clusters in order to implement the distributed control [5,6]. Each cluster has a cluster head (CH) and some member nodes. In each cluster, the data collected by member nodes are aggregated into the CH [7]. However, excessive amount of data transmission and interference from neighbor nodes may cause network congestion at the CH [8]. To achieve higher network efficiency and lower network congestion, global distributed congestion control is required for effective network management [9].

Active Queue Management (AQM) is commonly used to solve network congestion problem, which is implemented at source-node side. Random Early Detection (RED), as one of the most popular AQM schemes, keeps long queues proactively for alleviating network traffic problems in incipient congestion [10-12]. Depending on the current queue length threshold, RED randomly drops packets to achieve high utilization. The RED algorithm is adopted to implement the sliding windows at the source node in TCP congestion-avoidance mode [13-15]. The source node receives network congestion detection from RED algorithm and congestion state feedback from neighbor nodes, then drops packets appropriately.

Additive-Increase Multiplicative-Decrease (AIMD) scheme is often used for tackling congestion control problems by making appropriate adjustments on the sending rate [16]. Concentrating on the characterization of the stable state of the AIMD model, in [17] the author obtains the mean value and quantities of the network throughput.

The RED algorithm is initially designed to analyze and forecast congestion at source-node side, instead of the whole network. For global congestion control in CWSNs, we focus upon upgrading the throughput of network and keeping network parameters stable at the optimal state via the RED algorithm. To maximize the throughput of CWSN, a weighted fair scheduling algorithm is needed to pre-arrange the weight of the network parameters for each cluster. It is also difficult to keep network parameters stable at the optimal state due to the influence of many network parameters. We consider the sending rate at source-node side as the key parameter for global CWSN stability in this paper. Excessively restraining the sending rate at source-node side via RED algorithm in CWSNs may seriously decline the system throughput and link utilization, which prevents the network from working at the optimal state. Meanwhile, the sending rate at source-node side may fluctuate unstably around the optimal state. The instability of all source nodes may cause the instability of the whole network.

To describe the global network stability, we define a novel concept as cluster synchronization in CWSNs. Cluster synchronization means the sending rate of every source node in a cluster reaches a stable weighted fair value. Each CH makes appropriate congestion control and management with high rate adjustment to realize global cluster synchronization. The source node makes additive-increase adjustments on its sending rate after receiving the non-congestion state feedback of neighbor nodes, while making multiplicative-decrease adjustments on it on receiving the congestion state feedback of neighbor nodes or forecasting the network congestion via RED algorithm. The sending rate at each source-side can feasibly reach a weighted value which maximizes the throughput of CWSN. The sending rate at source-node side is considered as a continuous variable, and the network model can be established by means of Lyapunov-Krasovskii functionals [18-23].

The CH is able to dynamically adjust its network parameters based upon varying network conditions, especially the feedback from neighbors in the same cluster, which is called as self-adjustment. Existing work has been proposed to tackle the stability problem of WSN based on QoS mechanism. In [24] an analytic study on the interval service time of the worst-case response was presented to evaluate the performance via blackburst mechanism. In [2] the authors proposed to a scheduling strategy, established the WSN dynamical model and adopted traditional control theory to address system state stabilization. Traditional control theory using RED algorithm in [13-15] are conducted on congestion avoidance in wireless dynamic networks. The authors in [10] proposed to vary performance parameters according to various congestion algorithms. Meanwhile, cluster synchronization is a key parameter in network stability, and cluster synchronization analysis is investigated for Lur'e type dynamic network with delay [18,22]. The authors in [18] analyzed the continuous-time and discrete-time Lur'e dynamic delayed network by using pinning control strategy. Under a uniform scheme, synchronization could be realized in the network model, and the conditions were found to achieve different synchronous forms [22].

However, the aforementioned studies have three crucial limitations: i) only congestion avoidance is discussed instead of congestion control in global network circumstance; ii) the approaches focus primarily on source-side rather than the whole CWSN, and excessively restraining the sending rate at source-node side control; iii) there exist numerous traditional control theories for global system state stability and synchronization, but they do not work well in today's dynamic networks, especially in CWSNs congestion control. In order to address these problems, we propose an improved cluster synchronization model using Lyapunov-Krasovskii functionals in Lur'e dynamical WSN network system. i) via feedback from neighbor nodes, congestion control scheme makes appropriate adjustments to sending rate at source-node side to achieve the cluster synchronization in CWSNs; ii) cluster synchronization utilizing RED algorithm at source-node side is extended to implement global network stability; iii) by means of combining reciprocal convex technique, we consider the cluster synchronization of sending rates at source-node side with both lower and upper bounds of delay by using Lyapunov-Krasovskii functionals.

In our paper, significant research efforts are put on self-adjustment, which aims at establishing AIMD adjustment scheme at source-node side in CWSNs based upon congestion state feedback and RED threshold. The sending rate, as a key parameter at source-node side, also depends upon these two factors. The congestion state feedback and overtopping RED traffic threshold would cause multiplicative-decrease of the sending rate; while the non-congestion or non-obtainment of feedback would cause additive-increase of the sending rate. This process of sending rate self-adjustment is expanded from the source node to the global network. Therefore, the throughput of the whole CWSN can be maximized by solving the problem of global network congestion control.

We firstly adopt the AIMD adjustment scheme to analyze the delay in CWSNs, and establish a basic network model. Then, we discuss the global network throughput stability of congestion detection, cluster synchronization and parameter adjustments. Finally, numerical examples are given to demonstrate the effectiveness of theoretical analysis.

The main contributions of this paper include the following aspects:

i) More realistic controller strategy and delay analysis in CWSNs. Currently adopted control policies implemented in CWSNs are lack of global view of the network control architecture. It shows that global cluster synchronization can be achieved by means of the global view in the realistic CWSN. Moreover, the propagation delay is considered to make the analysis more realistic. We define the upper bound of delay as the network propagation delay, and analyze its influences in global view of CWSN.

ii) Wider and more effective applicability to solve the global cluster synchronization problem. Network optimization algorithm maintains the CWSN network states at the optimized values. By model transformations, the cluster synchronization network control system can be viewed as a normal stability control system. Thus, the global cluster synchronization that reflects the stability of CWSN control system can be applied wider and more effective.

iii) More mathematical support based on Lyapunov-Krasovskii functionals. We propose a novel control theoretical methodology for the stability of global CWSN control system instead of traditional control theoretical methodology. Through Lyapunov-Krasovskii functionals, some novel LMI-based stability criteria are established. As a result, the stability converging state of CWSN control system can ensure the global cluster synchronization.

The rest of the paper is organized as follows. In Section II, we establish an analytical network model by implementing AIMD scheme to adjust sending rate and utilizing RED algorithm to detect network congestion. In Section III, we introduce a scheduling problem formulation and some necessary preliminaries. Section IV addresses cluster synchronization of congestion control by means of Lyapunov-Krasovskii functionals, and the sufficient conditions are derived. In Section V, numerical examples are given to demonstrate effectiveness of theoretical results. Finally, the conclusions and direction for future work are presented in Section VI.

Notation: For a symmetric matrix $A, A > 0\ (A < 0)$ means that A is positive (negative) definite. $A \otimes B$ is the Kronecker product of matrices A and B. $He(Z) = Z + Z^T$ is used to shorten formulas. The symbol $*$ means a symmetric block in matrices. Identity matrices of dimension $n \times n$ are denoted by $I_{n \times n}$.

2. Model and Analysis

The CWSN consists of numerous clusters, in which each cluster has a unique CH and some member nodes. The CWSN establishes a closed-loop control system, as shown in **Fig. 1**. The CH acts as the unique congestion control and manage node in each cluster, which appropriately feeds the congestion control message. The source node adjusts its sending rate via the congestion control message by means of RED algorithm and AIMD scheme, and the member nodes in the same cluster still make influence on source-side rate adjustments in CWSNs.

By considering AIMD scheme to make the sending rate adjustments, we utilize RED algorithm for detection and the feedback information of neighbor link congestion state for wireless channel competition. We classify closed-loop control system into three parts and analyze parameters of each approach, respectively.

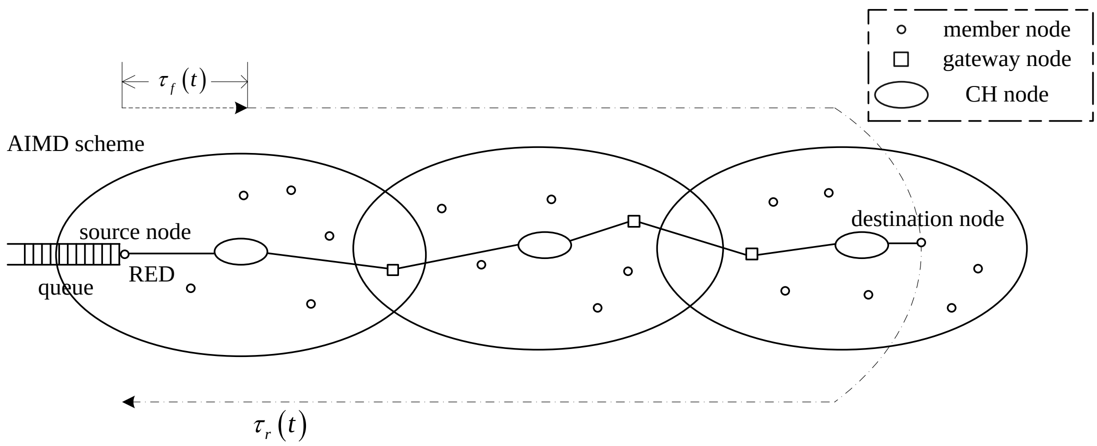

Fig. 1. The congestion control network scenario with delay by utilizing AIMD scheme and RED algorithm in CWSNs.

The notations used in the rest of the paper are summarized in in **Table 1**.

Table 1. The notations used in establishing congestion control model

Symbol	Explanations	Symbol	Explanations
$\tau(t)$	the whole round-trip time delay from the source node to the destination	q	average queue length
$\tau_f(t)$	forward propagation delay	R(t)	denotes round-trip= $\frac{q(t)}{C} + T_p$
$\tau_r(t)$	trip propagation delay	C	link capacity
A	a fixed weight for additive-increase to roughly adjust the sending rate of the source node	T_p	propagation delay
D	a fixed weight for multiplicative-decrease to roughly adjust the sending rate of the source node	N_s	load factor(number of sessions)
B	a fixed weight	p	probability
η	the sensitivity degree of the adjustment	th^{min}	the minimum threshold of dropping random packets through RED algorithm
$\dot{x}(t)$	the time-derivative	th^{max}	the maximal threshold of dropping random packets through RED algorithm
W	average window size		

2.1 Sending Rate Adjustment Considering AIMD Scheme

In traditional network feedback mechanism, the neighbor nodes give their feedback messages to the source node to advertise the current congestion state information. If the congestion occurred in adjacent link, it feeds the message back to the source node within fixed time intervals.

Fig. 1 gives an instance of data transmission process in CWSNs. The end-to-end communication process is modeled as a closed-loop control system, which contains a source node, some CHs in difference clusters, a set of gateway nodes, and a set of links connecting the nodes. Between each CH, the gateway nodes play an important role to connect two CHs in the neighboring clusters. We assume that each member node is a neighbor of the CH in the same cluster. It means that any member node is directly connected to the CH in the same cluster, and any CH in one cluster has only one hop to the gateway node in the same cluster, or two or three hops away from the CH in the neighboring cluster. When a member node has data to transmit, it's called a source node, and it sends data to the CH in the same cluster. The CH in the source cluster puts the data in a queue, and transmits the data in the queue to the neighboring cluster.

Based on the AIMD scheme, the source node makes additive-increase scheme on its rate upon receiving the non-congestion state feedback information from the CH, while making multiplicative-decrease scheme on its rate upon receiving the congestion state feedback information for forecasting the network congestion by means of RED algorithm or from the CH.

Before establishing the congestion control model, we firstly propose some concepts and assumptions for analyzing AIMD scheme.

The whole round-trip time delay is $\tau(\mathrm{t}) = \tau_f(t) + \tau_r(t)$. The end-to-end communication process is formed as a closed-loop control system, as shown in **Fig. 1**.

Assumption 1: The source node has an infinite stream of data awaiting transmission.

Assumption 2: We define two queue lengths in any node: current queue length x and ideal queue length x^*. The sending rate variation of the source node corresponds to the difference value $(x - x^*)$. If $\mathrm{x} - x^* > 0$, the node multiplicative-increase decreases its sending rate; else $x - x^* < 0$, the one additive-increase increases. The larger the difference, the more significant the rate variation at source-node side. Every node implements AIMD scheme by means of the difference value $(x - x^*)$ including the CH, member nodes and gateway nodes.

Assumption 3: The neighbor nodes (including the CH, the neighbor member nodes and the gateway node) explicitly advertise their congestion states via feedback to the source node. We assume that there is a local Congestion State (CS) added into the control message. This feedback message demonstrates the congestion state of the links in the whole round trip. The CS is either positive (congestion occurred) or non-positive (non-congestion).

Suppose the CS message arrives at source-node side at the moment $t_i \geq 0$. The CHs feed the CS message back to the source node, and the time-varying sending rate at source-node side is denoted by $\mathrm{r}(t_i)$. When the CS is non-positive, the source node increases its sending rate proportionally to weight A. Analogously, when the CS is positive, the source node decreases its rate proportionally to weight D. We formulate the behavior equation $\mathrm{r}(t_i)$ as:

$$r(t_i) = r(t_i - 1) - \{A\big[1 - \eta\big(x\big(t - \tau_r(t)\big) - x^*\big)\big] - Dr(t)[\eta(x\big(t - \tau_r(t)\big) - x^*)]\}$$

we have:

$$\frac{r(t_i) - r(t_i - 1)}{t_i - t_{i-1}}$$
$$= -\frac{1}{t_i - t_{i-1}}\{A\big[1 - \eta\big(x\big(t - \tau_r(t)\big) - x^*\big)\big]$$
$$- Dr(t)\big[\eta\big(x\big(t - \tau_r(t)\big) - x^*\big]\}$$
$$= -Br(t_i - \tau(t))\{A[1 - \eta\big(x\big(t - \tau_r(t)\big) - x^*\big] - Dr(t)[\eta(x\big(t - \tau_r(t)\big)$$
$$- x^*)]\}$$

where $Br(t_i - \tau(t)) = \frac{1}{t_i - t_{i-1}}$, and $\eta(x - x^*)$ denotes the probability parameter.

Let $a = AB, b = DB$, and suppose $r = c, x = x^*$ in equilibrium state, thus it yields

$$\begin{cases} \dot{r}(t) = (-ac\eta - bc^2\eta)\left(x\left(t - \tau_r(t)\right)\right) - bc\left(\frac{a}{a+bc}r(t)\right) \\ \dot{x}(t) = r\left(t - \tau_f(t)\right) \end{cases} \tag{1}$$

Eliminating $x(t)$ from (1) gives the second-order differential equation

$$\ddot{r}(t) + \kappa\dot{r}(t) + \vartheta r\left(t - \tau(t)\right) = 0 \tag{2}$$

where $\kappa = \frac{abc}{a+bc}, \vartheta = c\eta(a + bc)$, and $x(t)$ is instantaneous queue length. Note that the second-order system can be modeled into a state variable equation from (2) as follows:

$$\dot{r}(t) = \begin{pmatrix} 0 & 1 \\ 0 & -\kappa \end{pmatrix} r(t) + \begin{pmatrix} 0 & 0 \\ -\vartheta & 0 \end{pmatrix} r\left(t - \tau(t)\right) \tag{3}$$

Eq. (3) represents the sending rate adjustment at source-node side by utilizing AIMD scheme after receiving the CH state feedback. The congestion control network scenario with delay shown in Fig.1 is able to be modeled by (3).

2.2 Collaborative Influence by Utilizing RED Algorithm

The RED algorithm is used to detect the congestion state and adjust the sending rate at source-node side by randomly dropping packets beyond the pre-arranged threshold. In this paper, the RED algorithm collaborates with the AIMD scheme to detect and predict the network congestion situation to adjust sending rate of the source node, as shown in **Fig. 2**. Suppose that the member nodes periodically give the congestion state feedback to the CH in the same cluster. Then, the CH gives the feedback control and management to the source node. It means that the sending rate at source-node side makes appropriate adjustments through the AIMD scheme after receiving the CH control message. Meanwhile, the RED algorithm is utilized to detect the network congestion at source-node side, adjust the sliding window and make the source rate decrease after measuring up the threshold. For the gateway nodes, the feedback of congestion state is not only sent within the cluster, but also between the gateway nodes and the CHs in neighbor clusters. The CH sends data to the destination node in the destination cluster, and controls the sending rate after receiving the gateway node's feedback of the congestion state.

The adjustment of window size is considered as continuous-time variation instead of discrete-time variation based on the queue length at source-node side. It varies by 1 Round Trip Time (RTT) so that the continuous variation is described as dt/RTT.

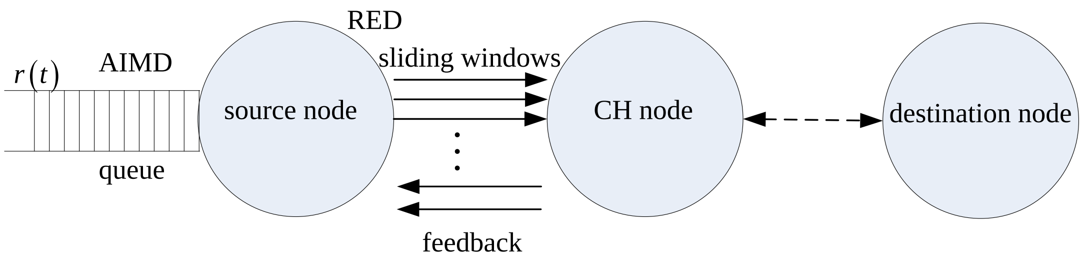

Fig. 2. The closed-loop network control system model with sliding windows and feedback message by collaboratively utilizing AIMD scheme and RED algorithm in CWSNs.

Before considering the equilibria between the window size and the queue length, we simplify our model by ignoring the timeout mechanism [17]. This model relates to the average value of key wireless network variables and is described by the following coupled, nonlinear differential equations:

$$\dot{W}(t)=\frac{1}{R(q(t))}-\frac{W(t)W(t-\tau(t))}{2R(q(t-\tau(t)))}p\big(x(t-\tau(t))\big)$$
$$\dot{q}(t)=\begin{cases} N_s(t)\frac{W(t)}{R(t)}-C, & q>0 \\ max\left\{0,-c+\frac{N_s(t)}{R(t)}W(t)\right\}, & q=0 \end{cases} \tag{4}$$

(4) can be linearized to obtain $R_0 \doteq \frac{q_0}{C}+T_p$. Suppose $N(t)=N_s, R(t)=R_0$ are constants, we have

$$\delta\dot{W}(t)=-\frac{N_s}{R_0^2C}\big(\delta W(t)+\delta W(t-\tau(t))\big)-\frac{R_0C^2}{2N_s^2}\delta p\big(t-\tau(t)\big)$$
$$\delta\dot{q}(t)=\frac{N_s}{R_0}\delta W(t)-\frac{1}{R_0}\delta q(t)$$

where $\delta W \doteq W-W_0, \delta p \doteq p-p_0, \delta q \doteq q-q_0$.
represent the perturbed variables about the operating point, we obtain

$$\delta\dot{W}(t)=-\frac{N_s}{R_0^2C}\big(\delta W(t)+\delta W(t-\tau(t))\big)-\frac{R_0C^2}{2N_s^2}\delta p\big(t-\tau(t)\big)$$

As $r(t)\propto\frac{W(t)}{R_0}$, $\dot{r}(t)\propto\left\{\frac{W(t)}{R_0}\right\}'=\frac{\dot{W}(t)}{R_0}$, we have

$$\dot{r}(t)=-\frac{N_s}{R_0^2C}f\big(r(t)\big)-\frac{N_s}{R_0^2C}f\Big(r\big(t-\tau(t)\big)\Big)-\frac{C^2}{2N_s^2}p\big(t-\tau(t)\big) \tag{5}$$

Considering the probability function $p(x)$ of the RED algorithm that takes as its argument an estimate of the average queue length at the node, we have

$$\begin{cases} 0, & 0\le x\le th^{min} \\ \frac{x-th^{min}}{th^{max}-th^{min}}p^{max}, & th^{min}\le x\le th^{max} \\ 1, & th^{max}<0 \end{cases} \tag{6}$$

Then, the second-order system in dropping packets threshold of RED algorithm combining with the terms in (5) and (6) is following as

$$\begin{aligned}\dot{r}(t) &= \begin{bmatrix} 0 & 0 \\ \frac{R_0 C^2}{2N_s^2} p & 0 \end{bmatrix} r\big(t-\tau(t)\big) + \begin{bmatrix} 0 & 0 \\ 0 & -\frac{N_s}{R_0 C} \end{bmatrix} W(t) + \begin{bmatrix} 0 & 0 \\ 0 & -\frac{N_s}{R_0 C} \end{bmatrix} W\big(t-\tau(t)\big) \\ &= \begin{bmatrix} 0 & 0 \\ \frac{R_0 C^2}{2N_s^2} p & 0 \end{bmatrix} r\big(t-\tau(t)\big) + \begin{bmatrix} 0 & 0 \\ 0 & -\frac{N_s}{R_0 C} \end{bmatrix} f\big(r(t)\big) \\ &+ \begin{bmatrix} 0 & 0 \\ 0 & -\frac{N_s}{R_0 C} \end{bmatrix} f\Big(r\big(t-\tau(t)\big)\Big)\end{aligned} \tag{7}$$

Eq. (7) indicates the collaborative effect by utilizing RED algorithm. The RED algorithm detects the congestion state and adjusts the sending rate based on the detection. The closed-loop network control system model with sliding windows and feedback message is able to be modeled by (7).

2.3 Influence of Wireless Channel Competition among the Neighbor Nodes

In addition to the aforementioned analysis, the problem of wireless channel competition in CWSNs is essential issue impacting on the sending rate adjustment at source-node side. By the broadcasting nature of the wireless medium, the nodes cannot simultaneously use the wireless channel [25]. The neighbor nodes around the source node may have data to send at the same moment. Thus, the source node needs to compete for the shared channel, as shown in **Fig. 3**. It is necessary to unify the sending rate at different source nodes in the same cluster for a period of time.

Fig. 3 gives an instance of wireless channel competition in CWSNs. When node A has data to send, the wireless channel must be listened to avoid collision. At this moment, node B is communicating with CH and occupies the wireless channel. Thus, the node A must wait for it. Meanwhile, when node C and D in the same cluster also have data to transmit, it must also compete with the node A.

The model analyzing the channel competition and interference is the same as the aforementioned analysis (shown in (3) and (7)).

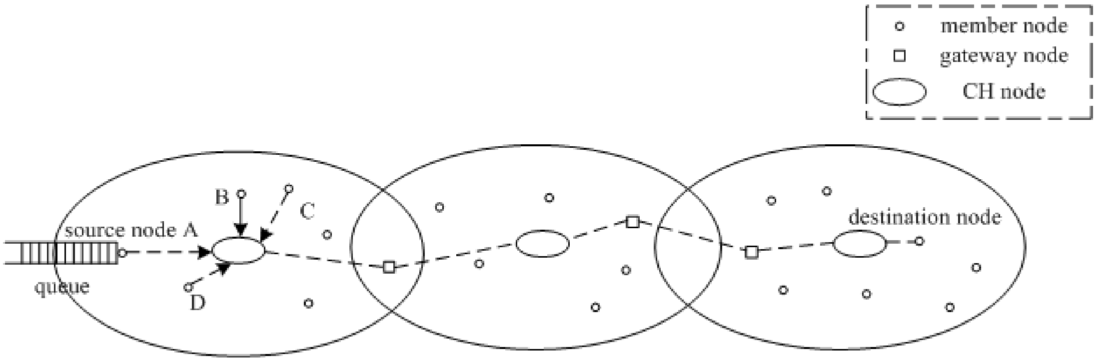

Fig. 3. The channel competition of the neighbor nodes in CWSNs.

For system (9), the following assumption is proposed for our cluster synchronization model throughout this paper.

Assumption 4: we define the influence of the coupled nodes as the reciprocal feedback influence among the nodes. The whole CWSN network in the same cluster is diffusively coupled. Each node is coupled with other nodes in the same cluster. Let $\mathbf{L} = (l_{ij})_{N\times N}$ be Laplace coupling matrix, which stands for the coupling configuration of the CWSN. The diagonal elements of the matrix **L** is defined as $l_{ii} = -\sum_{j=1,j\neq i}^{N} l_{ij}, l_{ij} > 0, i,j = 1,2,\dots,N$. This diagonal matrix represents the topology of the cluster. If there is a connection between node i and j (i.e. node i and j are neighbors), $l_{ij} = l_{ji} = 1$; otherwise, $l_{ij} = l_{ji} = 0 (i \neq j)$. The row sum of **L** is zero. The whole network is connected and the matrix L is irreducible.

Now, we consider the Laplace coupling matrix $\mathbf{L} = (l_{ij})_{N\times N}$ to represent the topology in a cluster. l_{ij} represents whether the node is the neighbor member node ($l_{ij} = 1, i \neq j$) or not ($l_{ij} = 0, i \neq j$). Through the Laplace coupling matrix **L**, the topology of all neighbor nodes around the source node in the same cluster can be described. We define $r_i(t)$ as the sending rate on the source node i. It follows (3) and (7), and the influence of AIMD scheme and RED algorithm in the neighbor nodes are simultaneously considered in the second-order system.

$$\dot{r}_i(t) = \sum_{i=1}^{N} \mathbf{G}^{(1)} l_{ij} r_j(t) + \sum_{i=1}^{N} \mathbf{G}^{(2)} l_{ij} r_j\big(t-\tau(t)\big) + \sum_{i=1}^{N} \mathbf{H}^{(1)} l_{ij} f\left(C^T r_j(t)\right) + \sum_{i=1}^{N} \mathbf{H}^{(2)} l_{ij} f\left(C^T r_j\big(t-\tau(t)\big)\right) \quad (8)$$

where $G^{(k)} > 0, H^{(k)} > 0, k = 1,2$ denote the coupling weights of AIMD scheme adjustment and RED algorithm detection, respectively. Therefore, the coupling influence of each neighbor node for the channel competition among the neighbor nodes has been described in (8).

Thus, we proposed our model for the global cluster synchronization of the CWSN. The sending rate of each node in the cluster is denoted by $r_i(t), i \in \{1, \ldots, N\}$. Generally, we consider the congestion control model of CWSN with each node being a n-dimensional delayed Lur'e dynamical networks, which can be abstractly formulated by combing (3),(7) and (8) as

$$\dot{r}_i(t) = \mathbf{A}r_i(t) + \mathbf{B}r_i(t-\tau(t)) + \mathbf{D}f\left(\mathbf{C}^{\mathbf{T}}r_i(t)\right) + \mathbf{D}f\left(\mathbf{C}^{\mathbf{T}}r_i(t-\tau(t))\right) + u_i(t) + \sum_{i=1}^{N}\mathbf{G}^{(1)}l_{ij}r_j(t) + \sum_{i=1}^{N}\mathbf{G}^{(2)}l_{ij}r_j(t-\tau(t)) + \sum_{i=1}^{N}\mathbf{H}^{(1)}l_{ij}f\left(\mathbf{C}^{\mathbf{T}}r_j(t)\right) + \sum_{i=1}^{N}\mathbf{H}^{(2)}l_{ij}f\left(\mathbf{C}^{\mathbf{T}}r_j(t-\tau(t))\right) \quad (9)$$

where $r_i(t) = [r_{il}(t), \ldots, r_{in}(t)]^T$ and $\mathbf{A} = (a_{ij})_{n\times n}, \mathbf{B} = (b_{ij})_{n\times n}, \mathbf{D} = (d_{ij})_{n\times n}, \mathbf{C} = (c_{ij})_{n\times n_1} = \{c_1, c_2, \ldots, c_{n_1}\}$ denote the parameters of the CWSN and $f(\boldsymbol{C}^T r_i(\cdot)) = \left\{f_1\left(c_1^T r_i(\cdot)\right), \ldots, f_{n_1}\left(c_{n_1}^T r_i(\cdot)\right)\right\}$ means the RED algorithm of each node in CWSNs, we also assume $G^{(k)} > 0, H^{(k)} > 0, k = 1,2$, which denote the coupling influence strength of the other neighbor nodes, i.e. the influence of state feedback of the other neighbor nodes, $u_i(t)$ is the control input The delay plays a key role in the global cluster synchronization of the CWSN. $\tau(t)$ denotes the network propagation delay that satisfies $0 \le \tau_0 \le \tau(t) \le \tau_m$, $\mu_0 \le \dot{\tau}(t) \le \mu_m < +\infty$, and τ_0, τ_m, μ are constants, $\bar{\tau}_m = \tau_m - \tau_0$. Thus, the maximum value of the propagation delay τ_m represents the upper limit of the CWSN network timeout.

Eq. (8) represents the influence of channel competition among the neighbor node. **Fig. 3** shows the influence of channel competition for data transmission. The equation considers three circumstances by the congestion control model established in section 2.1, 2.1 and 2.3, and combines (3), (7) and (8) for the global congestion control model.

3. A Scheduling Problem Formulation for Maximizing the Global CWSN Throughput

This section introduces maximizing the global CWSN throughput under the congestion control and the global cluster synchronization. The CH as a control and management node can not only send data as a source node and collect data from the member nodes, but also forward the data for other clusters through the gateway nodes. We consider the CH as a criterion node in the CWSN and a scheduling problem needs to be pre-arranged in order to unify the network parameters of each node in a cluster. Following the aforementioned analysis, we have to allocate appropriate sending rates at source-node side based on the feedback of the CH. Ideally, there exists an optimized stable sending rate state in each cluster under congestion control to achieve global cluster synchronization.

Now, we consider maximizing the global throughput with limited wireless network resources in CWSNs. We have a set of k clusters $NC_1, \dots, NC_k$, where each cluster contains only one CH. Each CH NC_i (*i=1,…,k*) has a maximum process capability C_i. We assume that there are only k levels of sending rate at source-node side, which are proportional to weights w_i. Additionally, the CHs can process a set of data flows $P(t)_1, \dots, P_m(t)$ *(j=1,…,m)* by scheduling. All data flows must be transmitted to the CH, and the process capabilities on each CH is C_i. For convenience, suppose $0 < C_1 \le \dots \le C_k < +\infty$. Each data flow P_j is processed by a series of CHs $u_{ij} \in \{NC_1, \dots, NC_k\}$. Ignore the process delays on CH, we simplify the optimized model to maximize the global throughput. The optimization problem can be formulated as following, noted that throughput rate S_i $(\mathrm{t})(\mathrm{i} = 1, \dots, \mathrm{k})$ associated with data flows on the CH.

$$\max_{w_i>0, S_i>0} \sum_i S_i(t)$$

$$s.t.\ \ 0 \le S_i(t) = \textstyle\sum_j C_{u_{ij}} \le C_i,$$

$$0 \le C_{u_{ij}} \le \min\{C_i\} = C_1,$$

$$\frac{S_1(t)}{w_1} = \dots = \frac{S_i(t)}{w_i} = \dots = \frac{S_k(t)}{w_k},$$

$$\textstyle\sum_i C_{u_{ij}} = P_j(t).$$

The data flows within an optimized scheduling can be assigned to maximize the throughput of global CWSN under the aforementioned satisfaction conditions. The problem is to keep the optimization model system stable in the CWSN, which maintains allocated weighted proportion at source-node side as $S(t) = \{S_1(t), S_2(t), \dots, S_k(t)\}$. If the global cluster synchronization of the CWSN is achieved, the CWSN system is stable at the maximal network throughput. Therefore, when the optimized state $S(t)$ is calculated and achieved, each sending rate in a cluster needs to synchronize at this criterion state $S(t)$.

It is preferred to keep the optimized state $S(t)$ stable under congestion control. The sending rate of each node in the same cluster with desired inhomogeneous state needs to be unified as $r_1(t), \dots, r_{m_1}(t) \to S_1(t), r_{m_1+1}(t), \dots, r_{m_2}(t) \to S_2(t), \dots, r_{m_{k-1}+1}(t), \dots, r_{m_k}(t) \to S_k(t)$, where $r_i(t) \to S_\chi(t)$ denotes that $\lim_{t\to+\infty} \| r_i(t) - S_\chi(t) \| = 0$ for $i = 1,2,\dots,N$; $\chi = 1,2\dots,k$, the congestion control of cluster synchronization based on the sending rate is modeled by CWSN systems under optimized input control policy.

In this paper, we consider an appropriate optimized policy control input $u_i(t)$ that is obtained by the aforementioned analysis for system (9), and we have

$$u_i(t) = -v\mathbf{G}^{(1)}\left(r_i(t) - S_\chi(t)\right) - \theta\mathbf{G}^{(2)}\left(r_i\big(t-\tau(t)\big) - S_\chi\big(t-\tau(t)\big)\right) \tag{10}$$

with $\boldsymbol{\Upsilon} = diag\{v_1, v_2, \dots, v_{n_1}\}, \boldsymbol{\Theta} = diag\{\theta_1, \theta_2, \dots, \theta_{n_1}\}$.

Let $e(t) = \{e_1(t), e_2(t), \dots, e_n(t)\}^T = \left\{[r_1(t) - S_\chi(t)]^T, [r_2(t) - S_\chi(t)]^T, \dots, [r_n(t) - S_\chi(t)]^T\right\}$ be the error states to solve the cluster synchronization problem. After calculating the optimized throughput of the global CWSN, we focus on these error states as the variables in order to unify the sending rates at source-node side in the same cluster. Combine the term (9) with (10), and expanded from the source node to the global CWSN analysis (e → e_i), the global cluster synchronization error system of congestion control is described as follows:

$$\begin{aligned}\dot{e}_i(t) = \mathbf{A}e_i(t) + \mathbf{B}e_i(t-\tau(t)) + \mathbf{D}\bar{f}\left(\mathbf{C}^{\mathrm{T}}e_i(t)\right) + \mathbf{D}\bar{f}\left(\mathbf{C}^{\mathrm{T}}e_i(t-\tau(t))\right) - v\mathbf{G}^{(1)}e_i(t) \\ - \theta\mathbf{G}^{(2)}e_i(t-\tau(t)) + \sum_{i=1}^{N}\mathbf{G}^{(1)}l_{ij}e_j(t) + \sum_{i=1}^{N}\mathbf{G}^{(2)}l_{ij}e_j(t-\tau(t)) \\ + \sum_{i=1}^{N}\mathbf{H}^{(1)}l_{ij}\bar{f}\left(C^Te_j(t)\right) + \sum_{i=1}^{N}\mathbf{H}^{(2)}l_{ij}\bar{f}\left(C^Te_j(t-\tau(t))\right)\end{aligned} \tag{11}$$

where $\bar{f}(\mathbf{C}^Te_i(\cdot)) = f\left(\mathbf{C}^Te_i(\cdot) + \mathbf{C}^Ts_\chi(\cdot)\right) - f(\mathbf{C}^Te_i(\cdot))$.

Eq. (11) represents the error state of sending rate at source-node side in CWSNs. The sending rate adjustment is considered by detection of RED algorithm, adjustment of the AIMD scheme and influence of the channel competition among the neighbor nodes.

4. Cluster Synchronization Criteria for Keeping WSN Network Parameters Stable at Optimized States

This section introduces the cluster synchronization criteria in order to derive the sufficient conditions. We adopt the control theoretical methodology of the global stability for achieving the global cluster synchronization in CWSNs. Firstly, we present some mathematical lemmas in this section. Then, we discuss a new stability criterion for the sufficient conditions of global cluster synchronization in our CWSN network system. Finally, the drive system (9) synchronizes with the response system (11).

For presentation convenience, we denote $\bar{\mathbf{A}} = \mathbf{A}\otimes\mathbf{I_N}, \bar{\mathbf{B}} = \mathbf{B}\otimes\mathbf{I_N}, \bar{\mathbf{D}} = \mathbf{D}\otimes\mathbf{I_N}, \bar{\mathbf{C}} = \mathbf{C}\otimes\mathbf{I_N}$, and $\mathrm{e(t)} = \left(e_i^T(t)\right), i = 1,2, \dots, N$.

Using the Kronecker product, we can rewrite the error system (11) into a more compact form as

$$\begin{aligned}\dot{e}(t) = \bar{\mathbf{A}}e(t) + \bar{\mathbf{B}}e\big(t-\tau(t)\big) + \bar{\mathbf{D}}\bar{f}\big(\bar{\mathbf{C}}^T e(t)\big) + \bar{\mathbf{D}}\bar{f}\left(\bar{\mathbf{C}}^T e\big(t-\tau(t)\big)\right) - \big(\mathbf{Y}\otimes\mathbf{G}^{(1)}\big)e(t) \\ -\big(\boldsymbol{\Theta}\otimes\boldsymbol{G}^{(2)}\big)e\big(t-\tau(t)\big) + \big(\mathbf{G}^{(1)}\otimes\mathbf{L}\big)e(t) + \big(\mathbf{G}^{(2)}\otimes\mathbf{L}\big)e\big(t-\tau(t)\big) \\ + \big(\mathbf{H}^{(1)}\otimes\mathbf{L}\big)\bar{f}\big(\bar{\mathbf{C}}^T e(t)\big) + \big(\mathbf{H}^{(2)}\otimes\mathbf{L}\big)\bar{f}\left(\bar{\mathbf{C}}^T e\big(t-\tau(t)\big)\right)\end{aligned}$$

Lemma 1: [22,26,27] Let $\otimes$ denote the notation of Kronecker product. Then, the following properties are satisfied in appropriate dimensions.

i) $(\alpha\mathbf{A})\otimes\mathbf{B} = \mathbf{A}\otimes(\alpha\mathbf{B})$

ii) $(\mathbf{A}+\mathbf{B})\otimes\mathbf{C} = \mathbf{A}\otimes\mathbf{C} + \mathbf{B}\otimes\mathbf{C}$

iii) $(\mathbf{A}\otimes\mathbf{B})(\mathbf{C}\otimes\mathbf{D}) = (\mathbf{AC})\otimes(\mathbf{BD})$

Lemma 2: [23] For any symmetric matrix $\mathbf{W}\in\mathbb{R}^{n\times n}, \mathbf{W}=\mathbf{W}^T>0$, scalar $0\le h(t)\le h_M$, vector function $\omega: [0, h_M]\to R^n$ such that the follow integrations is well defined, then $-h(t)\int_{-h(t)}^{0}\dot{e}^T(t+s)\mathbf{W}\dot{e}(t+s)ds \le \big[e(t)-e\big(t-h(t)\big)\big]^T\mathbf{W}\big[e(t)-e\big(t-h(t)\big)\big]$;

Lemma 3: [21] Let the functions $f_1(t), f_2(t), \dots, f_N(t): \mathbb{R}^M\to\mathbb{R}$ have the positive values in an open subset $\mathbb{D}$ of $\mathbb{R}^m$ and satisfies $(1/\alpha_1)f_1(t) + (1/\alpha_2)f_2(t) + \dots + (1/\alpha_N)f_N(t): \mathbb{D}\to\mathbb{R}$ with $\alpha_i>0$ and $\sum_{i=1}^{N}\alpha_i = 1$, then the reciprocal convex technique of $f_i(t)$ over the set $\mathbb{D}$ satisfies $\sum_i \frac{f_i(t)}{\alpha_i} \ge \sum_i f_i(t) + \sum_{i\ne j} g_{i,j}(t)$ $\forall g_{i,j}(t): \mathbb{R}^m \to \mathbb{R}$ $\begin{bmatrix} f_i(t) & g_{i,j}(t) \\ g_{i,j}^T(t) & f_i(t)\end{bmatrix} \ge 0$;

Assumption 5: For $i\in\{1,2,\dots,N\}$ and any $\alpha,\beta\in\mathbb{R}, \alpha\ne\beta$, the activation function $f(\cdot)$ satisfies the following conditions: $\sigma^- \le \frac{f(\alpha)-f(\beta)}{\alpha-\beta} \le \sigma^+$ where σ^-, σ^+ are given constants. It is easy to find $f(0)$, we set $\Sigma^+ = diag\{\sigma_1^+, \sigma_2^+, \dots, \sigma_{n_1}^+\}, \Sigma^- = diag\{\sigma_1^-, \sigma_2^-, \dots, \sigma_{n_1}^-\}$, and $\Sigma_1 = diag\{\sigma_1^+\sigma_1^-, \dots, \sigma_{n_1}^+\sigma_{n_1}^-\}, \Sigma_2 = diag\left\{\frac{\sigma_1^+ + \sigma_1^-}{2}, \dots, \frac{\sigma_{n_1}^+ + \sigma_{n_1}^-}{2}\right\}$

Theorem 1: Under the Assumptions 1 to 5, the congestion control system can achieve the desired cluster synchronization, if there exist $n\times n$ constant matrices $\mathbf{P}>0, \mathbf{Q_i}>0$ $(i=1,\dots,5), X_i>0$ $(i=1,2), \mathbf{Y}>0$, $n_1\times n_1$ diagonal matrices $\boldsymbol{\Lambda}>0, \boldsymbol{\Xi}>0, \boldsymbol{\Phi}>0, \boldsymbol{\Pi}>0$, so that the following conditions in (12) to (13) hold:

$$\begin{bmatrix} \mathbf{X_2} & \mathbf{Y} \\ \mathbf{Y^T} & \mathbf{X_2}\end{bmatrix} \ge 0 \tag{12}$$

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Omega_{11}} & \mathbf{X_1} & \boldsymbol{\Omega_{13}} & \mathbf{0} & \boldsymbol{\Omega_{15}} & \boldsymbol{\Omega_{16}} & \boldsymbol{\Omega_{17}} \\ * & \boldsymbol{\Omega_{22}} & \mathbf{X_2 - Y} & \mathbf{Y} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ * & * & \boldsymbol{\Omega_{33}} & \mathbf{X_2 - Y^T} & \boldsymbol{\Omega_{35}} & \boldsymbol{\Omega_{36}} & \boldsymbol{\Omega_{37}} \\ * & * & * & \boldsymbol{\Omega_{44}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ * & * & * & * & \boldsymbol{\Omega_{55}} & \boldsymbol{\Omega_{56}} & \boldsymbol{\Omega_{57}} \\ * & * & * & * & * & \boldsymbol{\Omega_{66}} & \boldsymbol{\Omega_{67}} \\ * & * & * & * & * & * & \boldsymbol{\Omega_{77}} \end{bmatrix} < 0, \quad (13)$$

where

$$\boldsymbol{\Omega_{11}} = \mathbf{Q_1 + Q_2 + Q_5 - X_1 - \Phi\Sigma_1 + N_1(\bar{A} - \Upsilon\otimes G^{(1)} + G^{(1)}\otimes L) + (\bar{A} - \Upsilon\otimes G^{(1)} + G^{(1)}\otimes L)^T N_1}$$

$$\boldsymbol{\Omega_{13}} = \mathbf{N_1\bar{B} + \bar{A}^T N_2{}^T + N_1(-\Theta\otimes G^{(2)} + G^{(2)}\otimes L) + (-\Upsilon\otimes G^{(1)} + G^{(1)}\otimes L)^T N_2{}^T + \bar{B}^T N_3{}^T}$$

$$\boldsymbol{\Omega_{15}} = \mathbf{P + (\Sigma^{+}\Xi - \Sigma^{-}\Lambda) - N_1 + \bar{A}^T N_3{}^T + (-\Upsilon\otimes G^{(1)} + G^{(1)}\otimes L)^T N_3{}^T}$$

$$\boldsymbol{\Omega_{16}} = \mathbf{\Phi\Sigma_2 + N_1\bar{D} + N_1(H^{(1)}\otimes L) + \bar{A}^T N_4{}^T + (-\Upsilon\otimes G^{(1)} + G^{(1)}\otimes L)^T N_4{}^T}$$

$$\boldsymbol{\Omega_{17}} = \mathbf{N_1\bar{D} + N_1(H^{(2)}\otimes L) + \bar{A}^T N_5{}^T}$$

$$\boldsymbol{\Omega_{22}} = \mathbf{-Q_1 + Q_3 - X_2 - X_1}$$

$$\boldsymbol{\Omega_{33}} = \mathbf{(\mu_m - 1)Q_3 + (1 - \mu_0)Q_4 - (1 - \mu_m)Q_5 - X_2 - X_2{}^T + Y + Y^T - \Pi\Sigma_1 + N_2\bar{B} + \bar{B}^T N_2 + N_2(-\Theta\otimes G^{(2)} + G^{(2)}\otimes L) + (-\Theta\otimes G^{(2)} + G^{(2)}\otimes L)^T N_2}$$

$$\boldsymbol{\Omega_{35}} = \mathbf{-N_2 + (-\Theta\otimes G^{(2)} + G^{(2)}\otimes L)^T N_3}$$

$$\boldsymbol{\Omega_{36}} = \mathbf{N_2\bar{D} + N_2(H^{(2)}\otimes L) + \bar{B}^T N_5{}^T + (-\Theta\otimes G^{(2)} + G^{(2)}\otimes L)N_4{}^T}$$

$$\boldsymbol{\Omega_{37}} = \mathbf{\Pi\Sigma_2 + N_2\bar{D} + N_2(H^{(2)}\otimes L) + \bar{B}^T N_5{}^T + (-\Theta\otimes G^{(2)} + G^{(2)}\otimes L)N_5{}^T}$$

$$\boldsymbol{\Omega_{44}} = \mathbf{-Q_2 - Q_4 - X_2}$$

$$\boldsymbol{\Omega_{55}} = \mathbf{(\tau_0{}^2 X_1 + \bar{\tau}_m{}^2 X_2) - N_3}$$

$$\boldsymbol{\Omega_{56}} = \mathbf{(\Lambda - \Xi) + N_3\bar{D} + N_3(H^{(1)}\otimes L) - N_4{}^T}$$

$$\boldsymbol{\Omega_{57}} = \mathbf{N_3\bar{D} + N_3(H^{(1)}\otimes L) - N_5{}^T}$$

$$\boldsymbol{\Omega_{66}} = \mathbf{-\Phi + N_4(\bar{D} + H^{(1)}\otimes L) + (\bar{D} + H^{(1)}\otimes L)^T N_4}$$

$$\boldsymbol{\Omega_{67}} = \mathbf{N_4(\bar{D} + H^{(2)}\otimes L) + (\bar{D} + H^{(1)}\otimes L)^T N_5}$$

$$\boldsymbol{\Omega_{77}} = \mathbf{-\Pi + N_5(\bar{D} + H^{(2)}\otimes L) + (\bar{D} + H^{(2)}\otimes L)^T N_5}$$

Theorem 1 is deployed to propose sufficient conditions by applying Lyapunov-Krasovskii functionals. The congestion control system can achieve desired cluster synchronization if all network parameters satisfy the sufficient conditions derived from Theorem 1. Numerical examples are given to analyze the congestion control system in the following section.

5. Numerical Examples

In this section, numerical examples are used to illustrate the efficiency and stability of our sufficient conditions for global cluster synchronization given in Theorem 1. Following from (12) to (13), we consider the state equation of entire systems (9) with the parameters as follows: $\kappa = 1.55, \vartheta = 0.6, R_0 = 2, C = 16, p = 0.5, N_s = 2, N = 4$ and $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -\kappa \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 0 & 0 \\ \frac{-R_0 C^2 p}{2{N_s}^2} - \vartheta & 0 \end{bmatrix}, \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & -\frac{N_s}{R_0 C} \end{bmatrix}$ where the coupling matrix is $\mathbf{L} = \begin{bmatrix} -3 & 1 & 1 & 1 \\ 1 & -2 & 1 & 0 \\ 1 & 1 & -3 & 1 \\ 1 & 0 & 1 & -2 \end{bmatrix}$ and a cluster synchronization pattern of the initial conditions is considered as $e(t) = (0 \quad 1 \quad 0 \quad 2 \quad 0 \quad 0.5 \quad 0 \quad 1.5)^T$.

We consider a four-dimension matrix denotes four clusters in the closed-loop CWSN system. There are four clusters in the congestion control system and all the nodes in these clusters need to synchronize the sending rate in the same cluster. For simplicity, we assume that the state s_χ is the cluster synchronization of nodes' sending rates.$r_1(t), \ldots, r_{m_1}(t) \rightarrow s_1(t), r_{m_1+1}(t), \ldots, r_{m_2}(t) \rightarrow s_2(t), r_{m_2+1}(t), \ldots, r_{m_3}(t) \rightarrow s_3(t), r_{m_3+1}(t), \ldots, r_{m_4}(t) \rightarrow s_4(t)$, denote the cluster synchronization in the different clusters. The congestion control system can achieve desired cluster synchronization if the error state $e(t)$ approaches to zero. The traditional RED algorithm is not utilized for global stability, so that the CWSN may not achieve global cluster synchronization under the congestion control of traditional RED algorithm. Thus, there is possibility for the error state of closed-loop control system to diverge.

Example 1

This simulation example is demonstrated to show the effective optimized control in non-convergence CWSN network control scenarios.

Here, we first display the error states $e_i(t), i = 1,2,3,4$ in the aforementioned conditions without any control law. We define the inner coupling matrices as $\mathbf{G}^{(1)} = diag\{1,2\}, \mathbf{G}^{(2)} = diag\{4,10\}, \mathbf{H}^{(1)} = diag\{1.5,1\}, \mathbf{H}^{(2)} = diag\{2,3.5\}$ for establishing the non-convergence network control scenario. **Fig. 4** depicts the performance of the unstable (non-convergent) congestion control system based on traditional RED algorithm. As it can be observed from **Fig. 4**, the error states $e_i(t), i = 1,2,3,4$ are not convergent in 8 seconds.

The source node using our congestion control scheme adjusts its sending rate that reflects the efficiency influence of the coupled nodes' feedback. Compared with **Fig. 4**, the result for cluster synchronization in **Fig. 5** $e_i(t), i = 1,2,3,4$ denotes error states with 4 dimension couple control matrices. According to Theorem 1, there exists the feasible solution to LMI in (11)-(12). We obtain the allowable control gain (our congestion control scheme) as $\mathbf{\Upsilon} = diag\{-1.1, -1, -1.5, -2.1\}$, $\mathbf{\Theta} = diag\{-1.2, -1.1, -1.3, -1.1\}$.

Fig. 5 shows that the convergence of these error states has collaborative impact on other coupled nodes' states. Notably, the applied control force is very effective to generate practical implementation and all error states can achieve stability after a finite length of time. Interaction is obviously described with the couple matrices. Thus, the array of the error states in Lur'e dynamical network can achieve cluster synchronization, as still shown in **Fig. 5**.

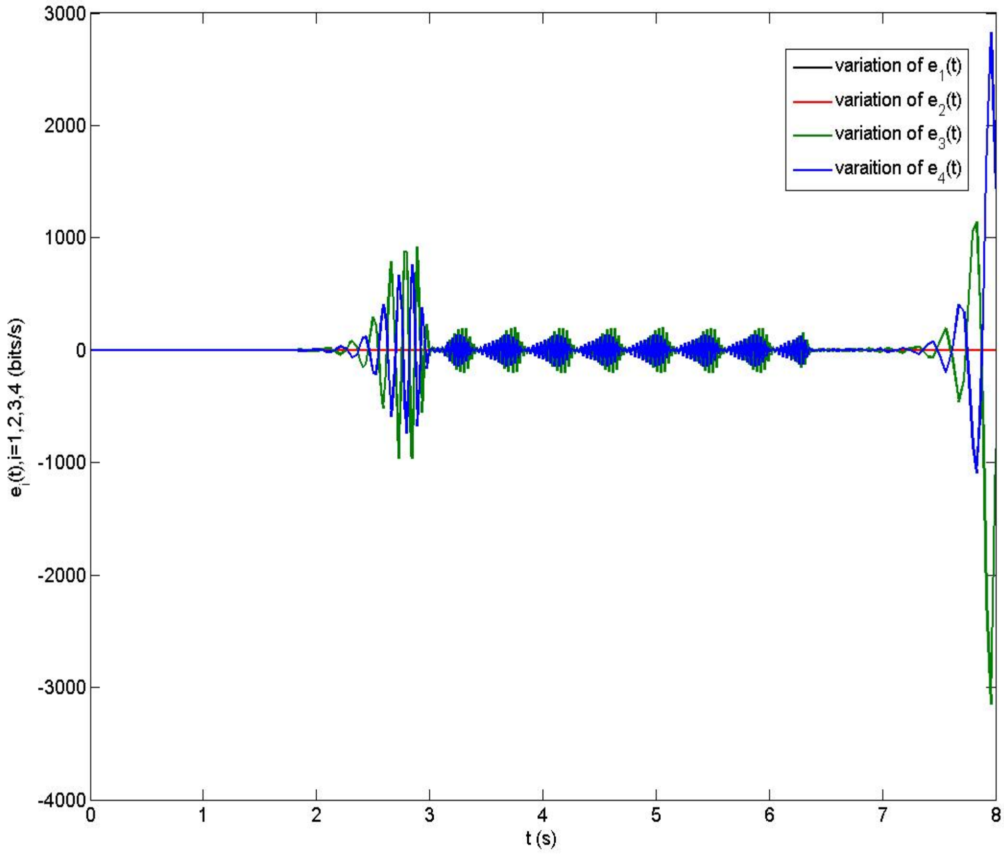

Fig. 4. The trajectories of error state $e_i(t), i = 1,2,3,4$ in the system (11) for non-convergence cluster synchronization.

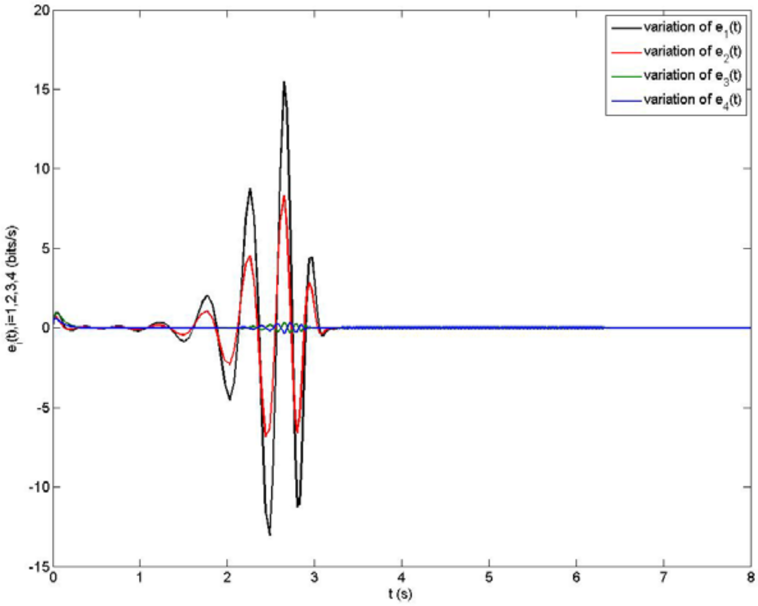

Fig. 5. The trajectories of error state $e_i(t), i = 1,2,3,4$ under the control law in the system (11) for non-convergence cluster synchronization.

Moreover, **Fig. 5** obviously shows both two error states converge to 0 after a finite length of time, which means $\lim_{t\to+\infty} \| r_i(t) - s_\chi(t) \| = 0$ for $i = 1,2,3,4$; $\chi = 1,2,3,4$. Therefore, this figures and numerical results effective optimized control under our congestion control scheme in non-convergence CWSN network control architecture.

Example 2

This simulation example is demonstrated to show the object parameter (the sending rate at source-node side) stability in convergence network control scenarios.

Here, we first display the error states $e_i(t), i = 1,2,3,4$ in the aforementioned conditions without any control law. We define the inner coupling matrices as $\mathbf{G}^{(1)} = diag\{1, 2\}, \mathbf{G}^{(2)} = diag\{2, 2.5\}, \mathbf{H}^{(1)} = diag\{1.5, 1\}, \mathbf{H}^{(2)} = diag\{2, 3.5\}$ for establishing the convergence network control scenario. **Fig. 6** illustrate the performance of the stable congestion control system based on traditional RED algorithm. **Fig. 6** notably shows that the error states $e_i(t), i = 1,2,3,4$ have been converged after a finite length of time.

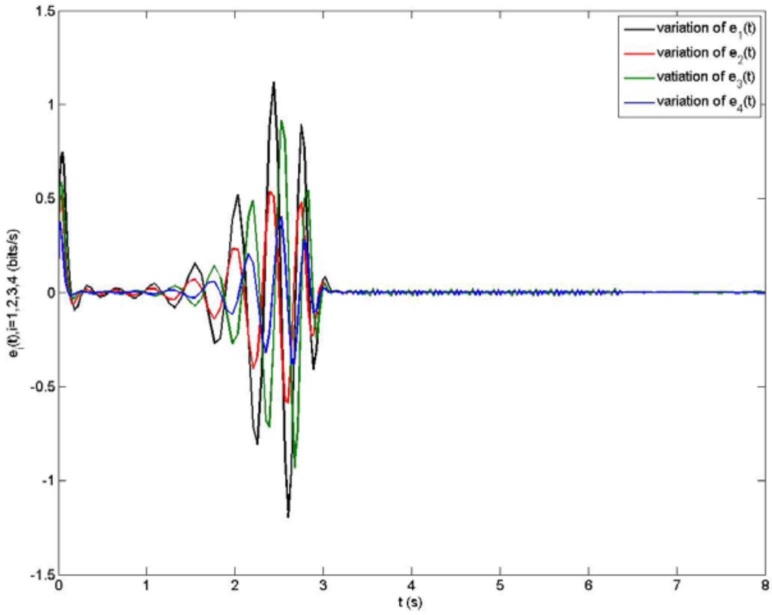

Fig. 6. The trajectories of error state $e_i(t), i = 1,2,3,4$ in the system (11) for convergence cluster synchronization.

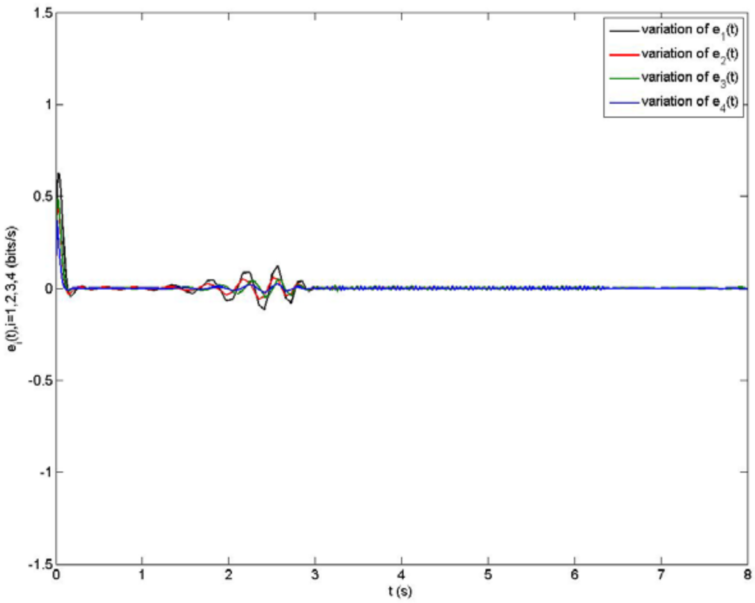

Fig. 7. The trajectories of error state $e_i(t), i = 1,2,3,4$ under the control law in the system (11) for convergence cluster synchronization.

By using our congestion control scheme, the source node adjusts its sending rate that reflects the stability influence of the coupled nodes' feedback. Applying Theorem 1 to this simulation example, we obtain the allowable control gain as $\mathbf{\Upsilon} = diag\{1.1, 1.4, 1.5, 2.1\}$, $\mathbf{\Theta} = diag\{1.2, 1.1, 1.3, 1.8\}$ and the result for cluster synchronization $e_i(t), i = 1,2,3,4$ denotes error states with 4 dimension couple control matrices in **Fig. 7**. Compared with **Fig. 6**, it is clear that the maximum amplitudes have been abbreviated under the control. Therefore, this simulation figures and numerical results stabilize the network parameter – sending rate at source-node side under our congestion control scheme in convergence network control architecture.

6. Conclusion

In this paper, we have adopted the additive-increase multiplicative-decrease (AIMD) scheme and proposed an improved RED algorithm by using neighbor feedback. A congestion control model in CWSNs has been designed, which is a linear system with a non-linear feedback modeled by a Lur'e type system. In the context of delayed Lur'e dynamical network, we have proposed the concept of cluster synchronization and shown that the system is able to achieve cluster synchronization. Sufficient conditions have been derived via Lyapunov-Krasovskii functionals. Numerical examples have investigated to validate the effectiveness of the CWSN optimization and the stability of the network parameter. The approach in this paper can possibly be extended to model and analyze complex synchronization network control mechanisms. For future studies, accurate modeling of RED algorithm would be discussed.

Appendix

A mathematical induction is employed to prove that the CWSN can achieve the desired cluster synchronization from (12) to (13) in Theorem 1.

Proof: Consider the following Lyapunov-Krasovskii functionals acting on system (11) as

$$\mathbf{V}(e(t)) = \sum_{i=1}^{3} \mathbf{V_i}(t)$$

$$\mathbf{V}_1(e(t)) = e^T(t)\mathbf{P}e(t)$$

$$\begin{aligned}\mathbf{V}_2(e(t)) = {} & 2\sum_{i=1}^{n} \lambda_i \int_0^{e_i(t)} [\bar{f}_i(s) - \delta^-(s)]\, ds + 2\sum_{i=1}^{n} \xi_i \int_0^{e_i(t)} [\delta^+ s - \bar{f}_i(s)]\, ds \\ & + \int_{t-\tau_0}^{t} e^T(s)\mathbf{Q_1}e(s)\, ds + \int_{t-\tau_m}^{t} e^T(s)\mathbf{Q_2}e(s) ds + \int_{t-\tau(t)}^{t-\tau_0} e^T(s)\mathbf{Q_3}e(s) ds \\ & + \int_{t-\tau_m}^{t-\tau(t)} e^T(s)\mathbf{Q_4}e(s) ds + \int_{t-\tau(t)}^{t} e^T(s)\mathbf{Q_5}e(s) ds\end{aligned}$$

$$\mathbf{V}_3(e(t)) = \tau_0 \int_{-\tau_0}^{0} \int_{t+\theta}^{t} \dot{e}^T(s)\mathbf{X_1}\dot{e}(s)dsd\theta + \bar{\tau}_m \int_{-\tau_m}^{-\tau_0} \int_{t+\theta}^{t} \dot{e}(s)\mathbf{X_2}\dot{e}(s)dsd\theta$$

with $n_1 \times n_1$ diagonal matrices $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_{n_1}\} > 0$, $\mathbf{\Xi} = \text{diag}\{\xi_1, \dots, \xi_{n_1}\} > 0$ to be determined. For any appropriate dimension matrix $\mathbf{N_i}(i = 1,2,3,4,5)$, we have

$$0 = 2\Big[e^T(t)\mathbf{N_1} + e^T(t-\tau(t))\mathbf{N_2} + \dot{e}^T(t)\mathbf{N_3} + \bar{f}^T(\bar{\mathbf{C}}^T e(t))\mathbf{N_4} + \bar{f}^T\Big(\bar{\mathbf{C}}^T e(t-\tau(t))\Big)\mathbf{N_5}\Big]\Big\{-\dot{e}(t) + \bar{\mathbf{A}}e(t) + \bar{\mathbf{B}}e(t-\tau(t)) + \bar{\mathbf{D}}\bar{f}(\bar{\mathbf{C}}^T e(t)) + \bar{\mathbf{D}}\bar{f}\Big(\bar{\mathbf{C}}^T e(t-\tau(t))\Big) - (\mathbf{Y}\otimes\mathbf{G^{(1)}})e(t) - (\mathbf{\Theta}\otimes\boldsymbol{G^{(2)}})e(t-\tau(t)) + (\mathbf{G^{(1)}}\otimes\mathbf{L})e(t) + (\mathbf{G^{(2)}}\otimes\mathbf{L})e(t-\tau(t)) + (\mathbf{H^{(1)}}\otimes\mathbf{L})\bar{f}(\bar{\mathbf{C}}^T e(t)) + (\mathbf{H^{(2)}}\otimes\mathbf{L})\bar{f}\Big(\bar{\mathbf{C}}^{\mathbf{T}} e(t-\tau(t))\Big)\Big\} \tag{14}$$

It follows from Assumption 5 for positive diagonal matrices, we can get that

$$0 \le \begin{bmatrix} e(t) \\ \bar{f}\Big(\bar{\mathbf{C}}(e(t))\Big) \end{bmatrix}^T \mathbf{J_1} \begin{bmatrix} e(t) \\ \bar{f}\Big(\bar{\mathbf{C}}(e(t))\Big) \end{bmatrix} + \begin{bmatrix} e(t-\tau(t)) \\ \bar{f}\Big(\bar{\mathbf{C}}\Big(e(t-\tau(t))\Big)\Big) \end{bmatrix}^T \mathbf{J_2} \begin{bmatrix} e(t-\tau(t)) \\ \bar{f}\Big(\bar{\mathbf{C}}\Big(e(t-\tau(t))\Big)\Big) \end{bmatrix} \tag{15}$$

where $\mathbf{J_1} = \begin{bmatrix} -\mathbf{\Phi\Sigma_1} & \mathbf{\Phi\Sigma_2} \\ \mathbf{\Phi\Sigma_2} & -\mathbf{\Phi} \end{bmatrix}, \mathbf{J_2} = \begin{bmatrix} -\mathbf{\Pi\Sigma_1} & \mathbf{\Pi\Sigma_2} \\ \mathbf{\Pi\Sigma_2} & -\mathbf{\Pi} \end{bmatrix}$.

Now, applying Assumption 4 and Lemmas 2 and 3, we estimate the time derivative of $\mathbf{V}(e(t))$ along the system as follows:

$$\dot{\mathbf{V}}_\mathbf{1}(e(t)) = 2e(t)\mathbf{P}\dot{e}(t)$$

$$\begin{aligned}\dot{\mathbf{V}}_\mathbf{2}(e(t)) \le\ & 2[\bar{f}^T(e(t))(\mathbf{\Lambda} - \mathbf{\Xi})\dot{e}(t) + e^T(t)(\mathbf{\Sigma^+\Xi} - \mathbf{\Sigma^-\Lambda})\dot{e}(t)] \\ & + e^T(t)[\mathbf{Q_1} + \mathbf{Q_2} + \mathbf{Q_5}]e(t) + e^T(t-\tau_0)[-\mathbf{Q_1} + \mathbf{Q_3}]e(t-\tau_0) \\ & + e^T(t-\tau_m)[-\mathbf{Q_2} - \mathbf{Q_4}]e(t-\tau_m) \\ & + e^T(t-\tau(t))[(\mu_m - 1)\mathbf{Q_3} + (1-\mu_0)\mathbf{Q_4} + (\mu_m - 1)\mathbf{Q_5}]e(t-\tau(t))\end{aligned}$$

$$\dot{V}_3(e(t)) \le \dot{e}^T(t)[{\tau_0}^2\mathbf{X_1} + {\tau_m}^2\mathbf{X_2}]\dot{e}(t) - [e(t) - e(t-\tau_0)]^T\mathbf{X_1}[e(t) - e(t-\tau_0)] - \begin{bmatrix} e(t-\tau_0) \\ e(t-\tau(t)) \\ e(t-\tau_m) \end{bmatrix}^T \mathbf{K} \begin{bmatrix} e(t-\tau_0) \\ e(t-\tau(t)) \\ e(t-\tau_m) \end{bmatrix} \tag{16}$$

where $\mathbf{K} = \begin{bmatrix} \mathbf{X_2} & \mathbf{Y} - \mathbf{X_2} & -\mathbf{Y} \\ * & 2\mathbf{X_2} - \mathbf{Y} - \mathbf{Y^T} & \mathbf{Y^T} - \mathbf{X_2} \\ * & * & \mathbf{X_2} \end{bmatrix}$.

It follows from (14) to (16), that $\dot{\mathbf{V}}(t) \le \varsigma^T(t)\mathbf{\Omega}\varsigma(t)$, where $\mathbf{\Omega}$ is respectively expressed in (13), and

$$\varsigma^T(t) = \Big[e^T(t) \quad e^T(t-\tau_0) \quad e^T(t-\tau(t)) \quad e^T(t-\tau_m) \quad \dot{e}^T(t) \quad \bar{f}^T\Big(\bar{\mathbf{C}}^{\mathbf{T}} e(t)\Big) \quad \bar{f}^T\Big(\bar{\mathbf{C}}^{\mathbf{T}} e(t-\tau(t))\Big)\Big]$$

Based on $\Omega < 0$ in (12) and convex combination technique that $\dot{V}(t) \leq 0$ and $\dot{V}(t) = 0$ if and only if $\varsigma(t) \equiv 0$. Thus, proof is completed, and the sufficient conditions are proposed for global cluster synchronization in CWSNs.

References

[1] K. Mikhaylov and J. Tervonen. "Data collection from isolated clusters in wireless sensor networks using mobile ferries," in *Proc. of Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on. IEEE*, 2013. Article (CrossRef Link)

[2] C. Qu, H. Li, and W. Chen. "Virtual queue based distributed data traffic scheduling in WSNs for distributed voltage control in smart micro-grid," *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on. IEEE*, 2015. Article (CrossRef Link)

[3] G. Mao, B. Fidan, and B. D.O. Anderson. "Wireless sensor network localization techniques," *Computer networks* 51.10, 2529-2553, 2007. Article (CrossRef Link)

[4] B. Johansson, C. M. Carretti, and M. Johansson, "On distributed optimization using peer-to-peer communications in wireless sensor networks," in *Proc. of Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON'08. 5th Annual IEEE Communications Society Conference on. IEEE*, 2008. Article (CrossRef Link)

[5] C. H. Tsai and Y. C. Tseng, "A path-connected-cluster wireless sensor network and its formation, addressing, and routing protocols," *Sensors Journal, IEEE* 12.6, 2135-2144, 2012. Article (CrossRef Link)

[6] A. Liu, P. Zhang, and Z. Chen, "Theoretical analysis of the lifetime and energy hole in cluster based wireless sensor networks," *Journal of Parallel and Distributed Computing* 71.10, 1327-1355, 2011. Article (CrossRef Link)

[7] D. Zhao and X. Shen, "Design of a large scale multi-cluster wireless sensor network," in *Proc. of Communications and Networking in China, 2009. ChinaCOM 2009. Fourth International Conference on. IEEE*, 2009. Article (CrossRef Link)

[8] R. Periyasamy and D. Perumal, "A Game Theory-Based Hybrid Medium Access Control Protocol for Congestion Control in Wireless Sensor Networks," *Game Theoretic Analysis of Congestion, Safety and Security. Springer International Publishing.* 1-25, 2015. Article (CrossRef Link)

[9] Q. Lin, R. Wang, J. Guo and L. Sun, "Novel congestion control approach in wireless multimedia sensor networks," *The Journal of China Universities of Posts and Telecommunications 18*, 1-8., 2011. Article (CrossRef Link)

[10] S. Patel, "Performance analysis and modeling of congestion control algorithms based on active queue management," in *Proc. of Signal Processing and Communication (ICSC), 2013 International Conference on. IEEE*, 2013. Article (CrossRef Link)

[11] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *Networking, IEEE/ACM Transactions on 1.4*, 397-413, 1993. Article (CrossRef Link)

[12] V. Misra, W. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," *ACM SIGCOMM Computer Communication Review*, Vol. 30, No. 4, ACM, 2000. Article (CrossRef Link)

[13] C. V. Hollot, V. Misra, D. Towsley and W. Gong, "On designing improved controllers for AQM routers supporting TCP flows," in *Proc. of INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. Vol. 3. IEEE*, 2001. Article (CrossRef Link)

[14] C. V. Hollot, V. Misra, D. Towsley and W. Gong, "Analysis and design of controllers for AQM routers supporting TCP flows," *Automatic Control, IEEE Transactions on47.6*, 945-959, 2002. Article (CrossRef Link)

[15] C. V. Hollot, V. Misra, D. Towsley, and W. Gong, "A control theoretic analysis of RED," in *Proc. of INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 3, 2001. Article (CrossRef Link)

[16] Y. R. Yang and S. S. Lam, "General AIMD congestion control," in *Proc. of Network Protocols, 2000. Proceedings. 2000 International Conference on. IEEE*, 2000. Article (CrossRef Link)

[17] F. Baccelli and D. Hong, "AIMD, fairness and fractal scaling of TCP traffic," in *Proc. of INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 1. IEEE, 2002. Article (CrossRef Link)

[18] T. Wang, Ting, T. Li, X. Yang and S. Fei, "Cluster synchronization for delayed Lur'e dynamical networks based on pinning control," *Neurocomputing 83*, 72-82, 2012. Article (CrossRef Link)

[19] L. Li and J. Cao, "Cluster synchronization in an array of coupled stochastic time delayed neural networks via pinning control," *Neurocomputing74.5*, 846-856, 2011. Article (CrossRef Link)

[20] Z. Ma, Z. Liu, and G. Zhang, "A new method to realize cluster synchronization in connected chaotic networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science 16.2*, 023103, 2006. Article (CrossRef Link)

[21] P. G. Park, J. W. Ko, and C. Jeong, "Reciprocally convex approach to stability of systems with time-varying time delays," *Automatica 47.1*, 235-238, 2011. Article (CrossRef Link)

[22] Q. Song and Z. Zhao, "Cluster, local and complete synchronization in coupled neural networks with mixed time delays and nonlinear coupling," *Neural Computing and Applications 24.5*, 1101-1113, 2014. Article (CrossRef Link)

[23] T. Li, A. Song, S. Fei and T. Wang, "Global synchronization in arrays of coupled Lurie systems with both time delay and hybrid coupling," *Communications in Nonlinear Science and Numerical Simulation 16.1*, 10-20, 2011. Article (CrossRef Link)

[24] N. Boughanmi, Y. Song, and E. Rondeau, "Priority and adaptive QoS mechanism for Wireless Networked Control Systems using IEEE 802.15.4," in *Proc. of IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society. IEEE*, 2010. Article (CrossRef Link)

[25] Y. Zhang, J. Chen and Z. Dai, "Wireless channel resource allocation in WSNs using finger-guessing game," in *Proc. of Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on. IEEE*, 2012. Article (CrossRef Link)

[26] J. Cao and L. Li, "Cluster synchronization in an array of hybrid coupled neural networks with delay," *Neural Networks 22.4*, 335-342, 2009. Article (CrossRef Link)

[27] W. Yu, J. Cao, G. Chen, J. Lu., J. Han, and W. Wei, "Local synchronization of a complex network model," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 39.1*, 230-241, 2009. Article (CrossRef Link)

Xi Hu was born in Wuhan, China. He is currently working towards the Ph.D. degree in National Key Laboratory of Science and Technology on Communications at University of Electronic Science and Technology of China, Chengdu, China. His research interests span the broad area of wireless communication and networks, network control system. Recently, he has been working on the stability control and robust control for network congestion in WSNs.

Wei Guo received his B.S. and M.S. degrees from University of Electronic Science and Technology of China, Chengdu, China. He currently works in National Key Laboratory of Science and Technology on Communications at University of Electronic Science and Technology of China as a professor. His research interest includes wireless communication and networks, Satellite and space communications technology, software systems and medical systems and telecare/telehealth networks.