Neural Networks 66 (2015) 1–10

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Event-triggered asynchronous intermittent communication strategy for synchronization in complex dynamical networks

Huaqing Li[a,b,*], Xiaofeng Liao[a], Guo Chen[b], David J. Hill[b,c], Zhaoyang Dong[b], Tingwen Huang[d]

[a] School of Electronic and Information Engineering, Southwest University, Chongqing 400715, PR China

[b] School of Electrical and Information Engineering, The University of Sydney, Sydney 2006, Australia

[c] Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong

[d] The Texas A & M University at Qatar, Doha, P. O. Box 23874, Qatar

ARTICLE INFO

Article history:
Received 16 July 2014
Received in revised form 16 November 2014
Accepted 26 January 2015
Available online 14 February 2015

Keywords:
Event-triggering
Distributed control
Complex networks
Synchronization
Strongly connected network
Directed spanning tree

ABSTRACT

This paper presents a new framework for synchronization of complex network by introducing a mechanism of event-triggering distributed sampling information. A kind of event which avoids continuous communication between neighboring nodes is designed to drive the controller update of each node. The advantage of the event-triggering strategy is the significant decrease of the number of controller updates for synchronization task of complex networks involving embedded microprocessors with limited on-board resources. To describe the system's ability reaching synchronization, a concept about generalized algebraic connectivity is introduced for strongly connected networks and then extended to the strongly connected components of the directed network containing a directed spanning tree. Two sufficient conditions are presented to reveal the underlying relationships of corresponding parameters to reach global synchronization based on algebraic graph, matrix theory and Lyapunov control method. A positive lower bound for inter-event times is derived to guarantee the absence of Zeno behavior. Finally, a numerical simulation example is provided to demonstrate the theoretical results.

1. Introduction

In recent years, much attention has been drawn to the study of dynamics of complex networks. The main reason is that many real systems can be described as complex dynamical networks such as Internet networks (Huberman & Adamic, 1999), biological networks (Pastor-Satorras, Smith, & Solé, 2003), epidemic spreading networks (Pastor-Satorras & Vespignani, 2001), collaborative networks (Barabási et al., 2002), social networks (Wang & Sun, 2008), etc. Control and synchronization of complex networks (see, for instance, Chen, Liu, & Lu, 2007, Lü & Chen, 2005 and Lu, Li, & Rong, 2010) have been one of the focal points in many research and application fields such as secure communication (Li, Liao, & Wong, 2004), information processing (Xie, Chen, & Bollt, 2002). The last years also witnessed a growing interest in coordination and cooperative control for complex networks using varieties of continuous-time feedback control techniques (Chen et al., 2007; Li, Liao, & Huang, 2013; Li, Liao, Huang, Zhu, & Liu, 2015; Li, Liao, Lei, Huang, & Zhu, 2013; Li et al., 2004; Lü & Chen, 2005; Lu & Chen, 2006; Lu et al., 2010; Song, Liu, Cao, & Yu, 2012; Xie et al., 2002; Yu, Chen, & Cao, 2011) and their application in pattern recognition (Li, Liao, Li, Huang, & Li, 2011). In particular, Zhu and Cao made an important contribution on constructing some novel synchronization criteria such as pth moment exponential synchronization (Zhu & Cao, 2012) and adaptive synchronization under almost every initial data (Zhu & Cao, 2011). Moreover, the synchronization problem of some new and general models with mixed time delays and/or Markovian switching was first introduced and studied in Zhu and Cao (2010, 2011, 2012). Recently, the synchronization of memristor-based recurrent neural networks and exponential synchronization of Markovian jumping neural networks were investigated in Chandrasekar, Rakkiyappan, Cao, and Lakshmanan (2014) and Rakkiyappan, Chandrasekar, Park, and Kwon (2014). The consensus of multi-agent systems can be also found in Li, Liao, and Chen (2013), Li, Liao, Dong, and Xiao (2012), Li et al. (2014), Li, Liao, Huang, and Zhu (2015) and Wen, Duan, Chen, and Yu (2014).

* Corresponding author at: School of Electronic and Information Engineering, Southwest University, Chongqing 400715, PR China.

E-mail addresses: huaqingli@hotmail.com (H. Li), xfliao@swu.edu.cn (X. Liao), andyguochen@gmail.com (G. Chen), dhill@eee.hku.hk (D.J. Hill), joe.dong@sydney.edu.au (Z. Dong), tingwen.huang@qatar.tamu.edu (T. Huang).

http://dx.doi.org/10.1016/j.neunet.2015.01.006

In practice, autonomous nodes such as mobile robots are often equipped with digital microprocessors which coordinate the data acquisition, communication with other nodes and control actuation. Thus, it is necessary to implement control laws on a digital platform. In other words, control laws can only be updated at discrete times. A commonly used approach in the present literature is time-scheduled control, which might be conservative in terms of the number of control updates, since the constant sampling period has to guarantee stability and convergence of resultant error closed-loop system in the worst-case scenario. Other limitations of time-scheduled control methods also include: (i) It is a detailed and restrictive design process, i.e., all design processes and their time specifications must be known in advance. (ii) The communication and the task scheduling on control units have to be synchronized during operation in order to ensure the strict time specifications in system design (Albert, 2004). Therefore, an efficient implementation for some special cases is impossible, typical examples are multi-rate sampling (Astrom & Bernhardsson, 2002). (iii) In terms of flexibility and scalability, there exist deficiencies in the designed architectures: a small change in one subsystem may generally imply an entire new system design. Thus the system design itself becomes very complicated and there is still a lack of adequate tools for design process (Kopetz, 1991).

In order to overcome the conservativeness of time-scheduled control, the event-triggering control is proposed in Astrom and Bernhardsson (2002), where updates of control law are only determined by certain events that are triggered depending on the nodes dynamic behaviors. The event-triggering control has been adopted for control engineering applications, such as wireless networks and networked control systems (Mazo & Tabuada, 2011). Generally, an event triggering is a control signal that is derived from an event. The event can originate either from activities within the computer system (e.g., termination of a task) or from state changes in the natural environment (e.g., alarm condition indicated by a sensor element). The main distinct feature of event-triggering control is that control law is updated only when some specific significant events occur, other state changes or occurrences of real-time entity are considered insignificant and are neglected (Kopetz, 1991). Event-triggering control mechanism presents many advantages in comparison with time-scheduled control methods. They can be summarized as follows: (i) Event-triggering systems have ability to fast react to asynchronous external events which are not known in advance. Thus, they show a better real-time performance in comparison with time-scheduled systems (Astrom & Bernhardsson, 2002). (ii) In event-triggering systems, it is very easy to modify an operative task to an existing node, since all scheduling and synchronization decisions are deferred activation of this task at run time. Therefore, event-triggering systems possess a higher flexibility and extensibility that allow in many cases the adaptation to the actual demand without a redesign of the complete system. (iii) In event-triggering systems, only those tasks that have been activated under the actual circumstances have to be scheduled. Since the scheduling decisions are made dynamically, the CPU will be available again after the actual (and not the maximum) task execution time. Therefore, if load conditions are low or average, then the resource utilization of an event-triggering system will be much better than that of a comparable time-scheduled system. (iv) Event-triggering systems have a better implementation in an actual circumstance. In a time-scheduled system, implementation requires a detailed design phase. In this design phase the maximum execution time of all time critical programs must be established and the execution schedules for operational modes must be calculated. However, if an event-triggering control strategy is chosen, this detailed planning phase is not necessary.

Up to date, great efforts have been made on consensus of multi-agent systems by designing the control laws based on event-triggering sampling schemes. Following the ideas proposed in Tabuada (2007), a decentralized event-triggering strategy is presented to determine updates of control law in Dimarogonas, Frazzoli, and Johansson (2012). A limitation of the control strategies presented in Dimarogonas et al. (2012) is the fact that they still require continuous communication between neighboring agents to constantly monitor whether the designed events occur or not, but keeping the benefit of less updates of control law. In Seyboth, Dimarogonas, and Johansson (2013), the event-triggering average consensus problem for single-integrators and double-integrators was studied. In Hu, Cao, Hu, and Guo (2015), the mean square consensus for multiple agents affected by noises over directed networks is investigated. The average consensus for multi-agent systems is studied in Liu and Chen (2010) while in Demir and Lunze (2012) the synchronization problem of multi-agent systems with event-based communication is considered. In Guinaldo, Dimarogonas, Johansson, Sanchez, and Dormido (2011), a distributed event-triggering control strategy for a networked dynamical system consisting of N linear time-invariant interconnected subsystems was presented. The triggering conditions used in Guinaldo et al. (2011) and Seyboth et al. (2013) focus on some state independent trigger functions. However, there are still some important yet challenging questions deserving further attention. Primarily, the existing literature purely consider simple dynamics, without being aware of nodes inherent nonlinear dynamics for the sake of convenience of theoretical derivation. Then the information interaction topology is undirected and the presented methods could not be generalized to directed case, which is ubiquitous in actual applications yet.

Motivated by above statements, the aim of this paper is to propose a unified framework for synchronization of general directed complex networks by a novel distributed event-triggering sampling control mechanism. A kind of event which avoids continuous communication between neighboring nodes is designed to drive the controller update of each node. The advantage of the event-triggering strategy is the significant decrease of the number of controller updates for synchronization task of complex networks involving embedded microprocessors with limited on-board resources. To describe the system's ability reaching synchronization, a concept about generalized algebraic connectivity is introduced for strongly connected networks and then extended to the strongly connected components of the directed network containing a directed spanning tree. Two sufficient conditions are presented to reveal the underlying relationships of corresponding parameters to reach global synchronization based on algebraic graph, matrix theory and Lyapunov control method. A numerical simulation example is provided to demonstrate the theoretical results.

We list some mathematically standard notations throughout this paper. Let Z, R, and C be the integral number set, the real number set, and the complex number set, respectively. R^n and $R^{m\times n}$ refer to the n-dimensional Euclidean space and the set of $m \times n$ real matrices, respectively. I_m denotes the identity matrix of order m and O denotes the zero matrix with appropriate dimension. 1_N denotes a vector with all elements being 1. For a real symmetric matrix A, let $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ denote respectively its minimum and maximum eigenvalue, and write $A > 0$ ($A < 0$) if A is a positive (negative) definite matrix. For a vector ξ, $\xi > 0$ represents all elements of ξ are positive. Let A^{T} be the transpose of a real matrix or a real vector A. Denote $|M|$ the cardinality of the set M. Unless specifically mentioned, all referenced norms $\|\cdot\|$ used in this paper refer to 2-norm for vectors or matrices. $\otimes$ indicates the Kronecker product. For a vector $z = [z_1, z_2, \ldots, z_n]^{\mathrm{T}}$, $\mathrm{diag}\{z\}$ represents a diagonal matrix with element in the position of ith row and ith column being z_i and other non-diagonal elements are zeros.