Automatica 64 (2016) 217–225

Contents lists available at ScienceDirect

Automatica

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

Brief paper

Collision-free consensus in multi-agent networks: A monotone systems perspective☆

Zhiqiang Miao [a,b], Yaonan Wang [a], Rafael Fierro [b]

[a] *College of Electrical and Information Engineering, Hunan University, Changsha 410082, China*

[b] *Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM 87131-0001, USA*

ARTICLE INFO

Article history:
Received 14 December 2014
Received in revised form
22 June 2015
Accepted 29 October 2015
Available online 7 December 2015

Keywords:
Consensus
Collision-free
Order preservation
Monotone systems
Multi-agent systems
Cooperative control

ABSTRACT

This paper addresses the collision-free consensus problem in a network of agents with single-integrator dynamics. Distributed algorithms with local interactions are proposed to achieve consensus while guaranteeing collision-free among agents during the evolution of the multi-agent networks. The novelty of the proposed algorithms lies in the definition of neighbors for each agent, which is different from the usual sense that neighbors are selected by the distance between agents in the state space. In the proposed strategies, the neighbor set for each agent is determined by the distance or difference between agents in the index space after ordering and labeling all agents according to certain ordering rules including weighted order and lexicographic order. The consensus analysis of the proposed algorithms is presented with some existing results on algebraic graph theory and matrix analysis. Meanwhile, by realizing the relations between order preservation and collision-free, a systematic analysis framework on order preservation and hence collision-free for agents in arbitrary dimension is provided based on tools from monotone systems theory. Illustrated numerical examples are presented to validate the effectiveness of the proposed strategies.

1. Introduction

There has been a growing interest in distributed coordination and cooperative control of multi-agent systems in the past decade (Bai, Arcak, & Wen, 2011; Bullo, Cortes, & Martinez, 2009; Lewis, Zhang, Hengster-Movric, & Das, 2014; Mesbahi & Egerstedt, 2010; Qu, 2009; Ren & Beard, 2008; Ren & Cao, 2011; Semsar-Kazerooni & Khorasani, 2012). This board interest in multi-agent networks is motivated by the emerging applications including vehicle formations, cooperative robotics, sensor networks, and social networks. Consensus as one of the fundamental problems in the coordination of multi-agent systems, refers to the agreement of all agents upon certain quantities of interest. Consensus has been studied extensively in the literature (Cao, Yu, Ren, & Chen, 2013; Fax & Murray, 2004; Moreau, 2005; Olfati-Saber, Fax, & Murray, 2007), and the applications of consensus algorithms can be found in formation control (Lin, Francis, & Maggiore, 2005), rendezvous (Cortes, Martinez, & Bullo, 2006), flocking (Tanner, Jadbabaie, & Pappas, 2007), attitude alignment (Jadbabaie, Lin, & Morse, 2003), and sensor networks (Kar & Moura, 2010). Here we do not intend to provide a completed review on consensus problem, the interested reader is referred to the survey paper (Cao et al., 2013) and the mentioned monographs for more detailed discussions.

The most well-known consensus algorithm is the local voting protocol which can be found in many of the aforementioned studies. Alternatively, cyclic pursuit strategies have recently been investigated for the distributed control multi-agent systems (Juang, 2013; Lin, Mireille, & Bruce, 2004; Marshall, Broucke, & Francis, 2006; Sinha & Ghose, 2006; Smith, Broucke, & Francis, 2005). In cyclic pursuit, each agent pursues its leading neighbor to form a ring network structure. The cyclic pursuit strategy is attractive because it inherently is decentralized and requires a small number of communication links. Cyclic pursuit strategies were utilized to address the agreement problem in Lin et al. (2004); Smith et al. (2005), formation control in Juang (2013) and Marshall

☆ This work was supported in part by the National Natural Science Foundation of China (61573134, 61433016), National High Technology Research and Development Program of China (863 Program: 2012AA111004, 2012AA112312), Hunan Provincial Innovation Foundation for Postgraduate (521298960), and the fellowship from China Scholarship Council (CSC). The material in this paper was partially presented at the 54th IEEE Conference on Decision and Control, December 15–18, 2015, Osaka, Japan. This paper was recommended for publication in revised form by Associate Editor Wei Ren under the direction of Editor Ian R. Petersen.

E-mail addresses: miaozhiqiang@hnu.edu.cn (Z. Miao), yaonan@hnu.edu.cn (Y. Wang), rfierro@unm.edu (R. Fierro).

http://dx.doi.org/10.1016/j.automatica.2015.11.025
0005-1098/

et al. (2006), and rendezvous problem in Sinha and Ghose (2006). Among these studies, a result in Lin et al. (2004) is closely relevant to the topic in this paper. It was showed in Lin et al. (2004) that if the agents in plane initially are ordered counterclockwise (or clockwise), then the order of agents will be preserved with cyclic pursuit, and therefore no collision would occur. Through the strategy proposed by Lin et al. (2004) achieved consensus while ensure collision-free among agents, the results are limited to agents in 2D, which render the applications involving unmanned aerial vehicles (UAV) in 3D difficult. Similar ideas in Lin et al. (2004) were reported in Wang, Xie, and Cao (2013) to solve the circle-forming problem for agents that move on a circle. In Wang et al. (2013), agents also are initially labeled counterclockwise on the circle, and each agent only receives information from its two immediate neighbors, resulting in an undirected ring network topology. The property of order preservation was studied as well to avoid inter-agent collisions. However, the result in Wang et al. (2013) is quite limited, as the motion of agents is restricted to one-dimensional space of a circle.

Most of the existing work on the distributed control of multi-agent networks pays more attention on the final or ultimate state rather than the entire trajectory of the agents. Take the consensus task for example, all agents are expected to reach a common value eventually, disregarding the transient behavior of agents. Nevertheless, the transient behavior of agents needs to be restrained to meet certain constraints in some cases. Inter-agent collision-free is an example of such constraints. Intuitively, if there exist certain quantities or qualities of the network that remain unchanged during the evolution, then the information from these invariants can help us better understand the transient behavior of agents throughout evolution. Order preservation is a kind of invariants, which makes the transient behavior of agents more predictable. As observed by Lin et al. (2004) and Wang et al. (2013), the order preservation property is particularly useful to prevent collisions between agents. Order preservation also has been explored in some social networks like Krause's opinion dynamics (Blondel, Hendrickx, & Tsitsiklis, 2009; Hendrickx, 2008; Yang, Dimarogonas, & Hu, 2014). A concise proof on order preservation was provided by Blondel et al. (2009) with simple arguments. However, the arguments only hold for agents in one dimension, which render the results rather conservative. A systematic analysis framework for arbitrary dimension remains to be provided.

In this paper, order preservation will be studied in the context of monotone systems (Altafini, 2013; Angeli & Sontag, 2003; Hirsch & Smith, 2005; Smith, 2008). Monotone systems are order-preserving systems, the trajectories of which preserve a given partial order through time. By realizing the relations between order preservation and collision-free, the problem of consensus with collision-free in a network of agents is addressed here. To be more specific, the contribution of this paper lies in designing distributed control laws such that all agents in the network reach the same value while guaranteeing inter-agent collision-free navigation. Preliminary results in this paper were presented in Miao, Wang, and Fierro (in press), where each agent is assumed to have the same number of neighbors. In this paper, we relax this assumption by generalizing the result to agents that have different number of neighbors. Moreover, we provide additional simulation case studies and a comparison of the proposed strategy with the cyclic pursuit strategy in Lin et al. (2004). The main contributions can be summarized as:

(i) Weighted order and lexicographic order are explored to define order for agents in arbitrary dimensional space;
(ii) Neighbors for each agent are defined in the index space rather than in state space. Notice that agents in the network do not necessarily have the same number of neighbors.
(iii) A systematic analysis framework based on monotone systems theory is provided to investigate order preservation and hence collision-free for agents.

The remainder of this paper is organized as follows. In Section 2, some mathematical preliminaries first are presented. Problem is formulated in Section 3. In Section 4, the proposed strategies and main results are stated. Simulation results for illustrating the effectiveness of the proposed strategies are given in Section 5. Section 6 concludes the paper.

2. Preliminaries

2.1. Notation

The standard notations are used throughout this paper. $\mathbb{N}$, $\mathbb{R}$ and $\mathbb{C}$ denote the sets of natural numbers, real numbers and complex numbers respectively. $\mathbb{R}_+$ denotes nonnegative real numbers, and $\mathbb{R}_+^n$ is the set of n-tuples for which all components belong to $\mathbb{R}_+$. Let $I_n \in \mathbb{R}^{n\times n}$ be the n-dimensional identity matrix; $1_n \in \mathbb{R}^n$ be the vector of all ones; $0_n \in \mathbb{R}^n$ be the vector of all zeros. For vectors $x, y \in \mathbb{R}^n$, we denote $x \leq y$ if and only if $x_i \leq y_i$ for all i; $x < y$ if and only if $x \leq y$ and $x \neq y$; and $x \ll y$ if and only if $x_i < y_i$ for all i.

2.2. Algebraic graph theory and matrix analysis

In this subsection, some elements and results on graph theory and matrix analysis that will used in this paper are reviewed. These results can be found in Ren and Beard (2008) and Ren and Cao (2011), and references therein.

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, A\}$ be a digraph with a node set $\mathcal{V} = \{1, 2, \ldots, n\}$, an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and an adjacency matrix $A = [a_{ij}]_{n\times n}$ with nonnegative elements. A directed edge denoted by (j, i) means that node i has access to node j, i.e., node i can receive information from node j. The elements of adjacency matrix A are defined as follows: If there is a directed link from node j to i $(j \neq i)$, then $a_{ij} > 0$; otherwise, $a_{ij} = 0$. We assume that $a_{ii} = 0$ for all i. The Laplacian matrix $L = [l_{ij}]_{n\times n}$ associated with the adjacency matrix A is defined as

$$l_{ij} = \begin{cases} -a_{ij} & i \neq j, \\ \sum_{j=1, j\neq i}^{n} a_{ij} & i = j. \end{cases} \tag{1}$$

A directed path is a sequence of edges in the directed graph $\mathcal{G}$ with distinct nodes. A directed graph is strongly connected if, for any two distinct nodes j and i, there exists a directed path from node j to node i. A digraph with n nodes is called a directed tree if it has $n - 1$ edges and there exists a root node with directed paths to every other node. A directed spanning tree is a directed tree that includes all the nodes of the digraph. A digraph is said to have a directed spanning tree if there is at least one node having a directed path to every other node.

Lemma 1. *The Laplacian matrix L has a simple zero eigenvalue with an associated eigenvector 1_n, and all the other eigenvalues have positive real parts if and only if the digraph associated with has a directed spanning tree.*

Definition 1. Given a matrix $S = [s_{ij}]_{n\times n}$, the associated digraph $\mathcal{G}(S)$ of matrix S is a directed graph with n vertices indexed by $1, 2, \ldots, n$ such that there is an edge in $\mathcal{G}(S)$ from j to i if and only if $s_{ij} \neq 0$.