Automatica 45 (2009) 2010–2017

Contents lists available at ScienceDirect

Automatica

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

Practical multiagent rendezvous through modified circumcenter algorithms☆

Sonia Martínez *

Mechanical and Aerospace Engineering Department, University of California at San Diego, 9500 Gilman Dr, La Jolla, CA, 92093-0411, USA

ARTICLE INFO

Article history:
Received 9 November 2007
Received in revised form
5 March 2009
Accepted 13 May 2009
Available online 4 July 2009

Keywords:
Multiagent systems
Mobile robots
Motion coordination
Rendezvous algorithm
Autonomous vehicles

ABSTRACT

We present a class of modified circumcenter algorithms that allow a group of agents to achieve "practical rendezvous" when they are only able to take noisy measurements of their neighbors. Assuming a uniform detection probability in a disk of radius σ about each neighbor's true position, we show how initially connected agents converge to a practical stability ball. More precisely, a deterministic analysis allows us to guarantee convergence to such a ball under r-disk graph connectivity in 1D under the condition that r/σ be sufficiently large. A stochastic analysis leads to a similar convergence result in probability, but for any $r/\sigma > 1$, and under a sequence of switching graphs that contains a connected graph within bounded time intervals. We include several simulations to discuss the performance of the proposed algorithms.

1. Introduction

The topic of distributed algorithms for robotic networks is attracting intense research activity in recent years; see e.g., Kumar, Leonard and Morse (2004). As a consequence of this, a wealth of algorithms are being proposed together with novel analysis tools to evaluate their performance. Clearly, an important aspect to consider is that of robustness to measurement and communication disturbance. If possible, a characterization of what typical degraded behaviors are, and how these are affected by the network size should be provided. When the characterized behavior is not satisfactory, the performed analysis may help find an alternative solution.

Motivated by this, we discuss how the nonlinear Circumcenter Algorithm, see Ando, Oasa, Suzuki, and Yamashita (1999), can be made robust with respect to measurement noise. This complements the work in Ando et al. (1999), which observed good performance of the algorithm in simulation, and the work in Cortés, Martínez, and Bullo (2006), Flocchini, Prencipe, Santoro, and Widmayer (2001) and Lin, Morse, and Anderson (2007a), which respectively considered asynchronous versions of the algorithm, and proved convergence under sequences of switching graphs. Other related papers include Kingston, Ren, and Beard (2005), Schenato and Zampieri (2006) and Xiao, Boyd, and Kim (2007), which study how consensus algorithms are robust to communication, measurement noise and quantization errors. However, the type of algorithms considered in these works are linear, while the Circumcenter Algorithm is nonlinear and agents' motion is constrained. For simplicity, we consider here first-order dynamics for each agent. Rendezvous algorithms for nonholonomic vehicles has been studied in; e.g., Dimarogonas and Kyriakopoulos (2007).

The contributions of this paper can be summarized as follows. First, we propose an alternative to the standard Circumcenter Algorithm. The alternative algorithm, which has been termed as the "1/2 Circumcenter Algorithm", does not require either of the following: (i) the explicit computation of constraint sets so agents maintain connectivity with others within distance $r > 0$, and (ii) knowledge of the absolute positions of neighbors. Second, we propose two possible modifications of the standard Circumcenter Algorithm and the new 1/2 Circumcenter Algorithm to deal with noisy measurements. The assumption is that every measured neighbor's position belongs to a disk centered at the neighbor's true position and radius $\sigma < r$. This noise can make agents lose connectivity when they implement the standard or the 1/2 Circumcenter Algorithms. The two proposed modifications of each algorithm guarantee agent connectivity. The first one restricts further each agent's motion constraint set to guarantee connectivity of the network (variant 1). In the second one, agents filter measurements of neighbors to make sure that they are still within the connectivity radius r (variant 2).

After this, we look at all variants of the modified circumcenter algorithms in 1D and analyze them using a deterministic approach. Being the proofs analogous, we present a detailed account of the modified (standard) Circumcenter Algorithm, variant 1. All

☆ The material in this paper was partially presented at 46th IEEE International Conference on Decision and Control, New Orleans, December 2007. This paper was recommended for publication in revised form by Associate Editor Zhihua Qu under the direction of Editor Andrew R. Teel.

* Tel.: +1 858 822 4243; fax: +1 858 534 4387.
E-mail address: soniamd@ucsd.edu.

0005-1098/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2009.05.013

proofs make use of basic mathematical arguments, require initial agent connectivity under the r-disk graph, and different lower-bounds on r/σ. For example, the modified (standard) Circumcenter Algorithm, variants 1 and 2, require that $r > 7\sigma$. We also show that the practical stability ball where all agents converge under any of the modified algorithms has a diameter upper bounded by 2σ.

As shown in simulations, the same type of multi-agent behavior does not hold for other graphs. To deal with those cases and motion in higher dimensions, we make use of stochastic analysis tools. By using an argument similar to that of the LaSalle invariance principle, we can characterize the behavior of the modified 1/2 Circumcenter Algorithm. More precisely, we can prove that the expectation of the position of every agent tends to the same point as time goes to infinity. This holds for any $r/\sigma > 1$, and under a periodic strong-connectivity assumption detailed in the paper. In general, we see that the diameter of the practical stability ball is upper bounded by $r - \sigma$ or r.

We also look at executions of the algorithms in simulation. In all cases, agents reach a practical stability ball with diameter much smaller than 2σ when using the r-disk graph. Simulations also show that convergence holds for small ratios r/σ at the expense of longer convergence times. In general, the convergence gets worse as connectivity becomes sparser and the number of agents increases. With respect to Martínez (2007), here we present the 1/2 Circumcenter Algorithm and extend the stochastic analysis to 2D.

The paper is organized as follows. Section 2 introduces preliminary notions, circumcenter algorithms and modifications. Section 3 includes a deterministic analysis of the modified circumcenter algorithms when implemented in 1D and over the r-disk graph. Section 4 includes a stochastic analysis of the algorithms in 2D. Finally, Section 5 illustrates the performance of the algorithms in simulations and Section 6 presents some concluding remarks.

2. Preliminaries

Here, we review some notation for standard geometric objects; for additional information the reader is referred to Bullo, Cortés, and Martínez (2009) and de Berg, van Kreveld, Overmars, and Schwarzkopf (2000). We then recall the circumcenter and parallel circumcenter algorithms as discussed in Ando et al. (1999), Lin, Morse, and Anderson (2007b) and Martínez, Bullo, Cortés, and Frazzoli (2007). The section concludes introducing the new class of modified circumcenter algorithms.

2.1. Basic geometric notions and notation

In what follows, $\mathbb{R}^d$ will refer to either $\mathbb{R}$ or $\mathbb{R}^2$. For a bounded set $S \subset \mathbb{R}^d$, we let $\mathrm{co}(S)$ denote the convex hull of S and $\mathrm{diam}(S) = \mathrm{diam}(\mathrm{co}(S)) = \max_{q_1,q_2 \in S} \|q_1 - q_2\|$ its diameter. For $p, q \in \mathbb{R}^d$, we let $(p, q) = \{\lambda p + (1-\lambda)q \mid \lambda \in (0, 1)\}$ and $[p, q] = \mathrm{co}(\{p, q\})$ denote the *open* and *closed segment* with extreme points p and q, respectively. For a bounded set $S \subset \mathbb{R}^d$, we let $\mathrm{CC}(S)$ and $\mathrm{CR}(S)$ denote the *circumcenter* and *circumradius* of S, respectively, that is, the center and radius of the smallest-radius d-sphere enclosing S. Let $S = \{q_1, \ldots, q_k\} \subseteq \mathbb{R}^2$, then it can be proved that $\mathrm{CC}(S) \in \mathrm{co}(S) \setminus S$. The computation of the circumcenter and circumradius of a bounded set is a strictly convex problem and in particular a quadratically constrained linear program. In particular, the circumcenter of a set of points $\mathrm{CC}(q_1, \ldots, q_k)$ becomes a continuous function of $q_1, \ldots, q_k$. For $p \in \mathbb{R}^d$ let $D(p, r)$ denote the *closed disk* of center p and radius $r \in \mathbb{R}_{>0}$.

In the sequel we will use tuples $P = (p_1, \ldots, p_n) \in \mathbb{R}^{dn}$ to refer to the positions of a group of n robots in space. The algorithms we consider are implemented in discrete time over a time schedule $m = 0, 1, 2, 3, \ldots$, and give rise to point sequences $\{P_m = (p_{1,m}, \ldots, p_{n,m}) \in \mathbb{R}^{dn}\}_{m \geq 0}$.

A *proximity graph function* $\mathcal{G}(\mathcal{P})$ associates to a point set $\mathcal{P} = \{p_1, \ldots, p_n\} \subset \mathbb{R}^d$ an undirected graph with vertex set $\mathcal{P}$ and edge set $\mathcal{E}_{\mathcal{G}}(\mathcal{P}) \subseteq \mathcal{P} \times \mathcal{P} \setminus \mathrm{diag}(\mathcal{P} \times \mathcal{P})$. In other words, the edge set of a proximity graph may depend on the location of its vertices. General properties of proximity graphs, basics on graph theory and examples can be found in Jaromczyk and Toussaint (1992), de Berg et al. (2000) and Bullo et al. (2009). In particular, we will make use of the r-disk proximity graph $\mathcal{G}_{\mathrm{disk}}(r)$, for $r \in \mathbb{R}_{>0}$, over a set of vertices $\mathcal{P}$. In this graph, two agents $p_i, p_j \in \mathcal{P}$ are neighbors iff $\|p_i - p_j\| \leq r$. We denote the set of neighbors of agent p_i in $\mathcal{G}(\mathcal{P})$ by:

$$\mathcal{N}_i(\mathcal{G}) = \{j \in \{1, \ldots, n\} \setminus \{i\} \mid (p_i, p_j) \in \mathcal{E}_{\mathcal{G}}(\mathcal{P})\},$$

and the cardinality of $\mathcal{N}_i(\mathcal{G})$ will be denoted as $n_i = |\mathcal{N}_i(\mathcal{G})|$. A sequence of tuples $\{P_m\}_{m\geq 0}$ (or associated finite point sets $\{\mathcal{P}_m\}_{m\geq 0}$) and a given $\mathcal{G}$ induce a sequence of graphs that we denote as $\mathcal{G}(m)$, $m \geq 0$, when it is clear from the context that $\mathcal{G}(m) \equiv \mathcal{G}(\mathcal{P}_m)$. We will also consider proximity graphs subject to link failures, $\mathcal{G}_{\mathcal{F}}(\mathcal{P})$. These are graphs on $\mathcal{P}$ with an edge set that may also be dependent on the location of the vertices. However, given (p_i, p_j) in $\mathcal{G}_{\mathcal{F}}(\mathcal{P})$, the reversed edge (p_j, p_i) may not be in $\mathcal{G}_{\mathcal{F}}(\mathcal{P})$. In other words, $\mathcal{G}_{\mathcal{F}}(\mathcal{P})$ is a directed graph. We will use these graphs to capture sensing or communication failures. When elements in the set $\mathcal{P}$ are indexed by $i \in \{1, \ldots, n\} = V$, the graph $\mathcal{G}_{\mathcal{F}}(\mathcal{P})$ can be associated with a graph G over V in a natural way. With a slight abuse of notation, we will sometimes identify these two objects.

For q_0 and q_1 in $\mathbb{R}^d$, and for a convex closed set $Q \subset \mathbb{R}^d$ with $q_0 \in Q$, let $\lambda(q_0, q_1, Q)$ denote the solution of the strictly convex problem:

$$\begin{aligned} &\text{maximize } \lambda \\ &\text{subject to } \lambda \leq 1,\ (1-\lambda)q_0 + \lambda q_1 \in Q. \end{aligned} \tag{1}$$

Note that this convex optimization problem has the following interpretation: move along the segment from q_0 to q_1 the maximum possible distance while remaining in the constraint set Q. Under the stated assumptions the solution exists and is unique. We will make explicit use of the constraint set in the circumcenter algorithms that follow.

2.2. Circumcenter algorithms

In the following, we present an informal description of one execution of the Circumcenter Algorithm for a group of agents $\mathcal{P}$. It is defined for any graph $\mathcal{G}_{\mathcal{F}}(\mathcal{P}) \subseteq \mathcal{G}_{\mathrm{disk}}(r)(\mathcal{P})$, with $r \in \mathbb{R}_{>0}$. For a formal description of this algorithm written in pseudocode, the reader is referred to Bullo et al. (2009).

> *(Standard) Circumcenter Algorithm* (Ando et al., 1999; Lin et al., 2007a; Cortés et al., 2006) Each agent performs the following actions: (i) it detects its neighbors according to the connectivity graph; (ii) it computes the circumcenter of the point set comprised of its neighbors and of itself; (iii) it moves to the closest point to the circumcenter while remaining in a constraint set $Q_i = \bigcap_{j \in \mathcal{N}_i(\mathcal{G}) \cup \{i\}} D\left(\frac{p_i + p_j}{2}, \frac{r}{2}\right)$. The constraint set guarantees connectivity with the group of previous neighbors in $\mathcal{G}_{\mathrm{disk}}(r)(\mathcal{P})$.

This algorithm can be implemented by each agent with knowledge of neighbors' positions in a local frame. When implemented over a proximity graph with failures, $\mathcal{G}_{\mathcal{F}} \subset \mathcal{G}_{\mathrm{disk}}$, convergence of the algorithm can be guaranteed as long as $\mathcal{G}_{\mathcal{F}}$ is periodically strongly connected (Cortés et al., 2006). The asynchronous behavior of the algorithm for was studied in Flocchini et al. (2001) and