# Coordinated rendezvous for visually-guided agents in a nonconvex polygon

Anurag Ganguli Jorge Cortés Francesco Bullo

***Abstract*— This paper presents coordination algorithms for mobile autonomous agents equipped with line-of-sight sensors in a nonconvex polygon. The objective of the proposed algorithms is to achieve rendezvous, that is, agreement over the location of the agents in the network, using only information from the line-of-sight sensors. Two key novel components of the algorithms are the notions of locally-cliqueless visibility graph and of convex continuous constraint set.**

## I. Introduction

Consider a group of robotic agents moving in a nonconvex environment. For simplicity, we model the environment as a simple polygon and the agents as point masses. We assume that each member of the group is equipped with omnidirectional line-of-sight sensors. By a line-of-sight sensor, we mean any device or combination of devices that can be used to determine, in its line-of-sight, (i) the position or state of another agent, and (ii) the distance to the boundary of environment. By omnidirectional, we mean that the field-of-vision for the sensor is $2\pi$ radians. In what follows, we shall use the terms visibility-based sensing and line-of-sight sensing interchangeably. We assume that the algorithm regulating the agents' motion is memoryless, i.e., we consider static feedback laws. Given this model, the goal is to design a provably correct discrete-time algorithm which ensures that the agents converge to a common location within the environment. See Figure 1 for a graphical description of our objective. Ideally, we would want the algorithm to be asynchronous but in this work we confine ourselves to the synchronous case.

This work is motivated by the recent surge of interest in the study of groups of mobile autonomous robots or agents. The "multi-agent rendezvous" problem and the first "circumcenter algorithm" have been introduced by Ando and coworkers in [1]. The algorithm proposed in [1] has been extended to various asynchronous strategies in [2], [3]. A related algorithm, in which connectivity constraints are not imposed, is proposed in [4].

One important difference between this paper and these works, is that we consider visually-guided robots. In fact, technical advancement in the fields of sensor technology and mobile robotics have facilitated the implementation of these algorithms on real systems. Examples of panoramic depth sensors relevant to the current work are (1) omnidirectional cameras, e.g., see [5], and (2) laser scanners with accurate distance measurements at high angular density. We conclude our literature review by mentioning that the problem of rendezvousing at a specified location for visually-guided agents was first introduced in [6]. However, the solution proposed was not distributed in the sense that each agent required the knowledge of the locations of all other agents in the network.

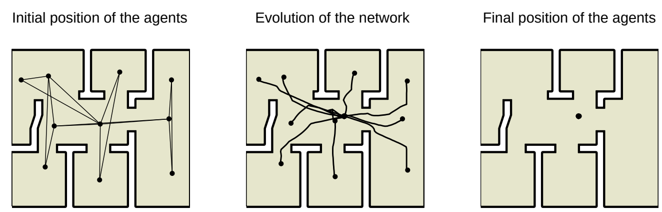


Fig. 1. Simulation results of the Circumcenter Algorithm described in Section IV-C on a network of agents distributed in a polygon shaped like a typical floor plan. The algorithm is run over the visibility graph $\mathcal{G}_{\text{vis},Q}$ (see Section III.

The contribution of this paper is threefold. First, we develop a geometric framework which makes it possible to apply recently developed results on convergence analysis of nonlinear systems, e.g., the LaSalle Invariance Principle for set-valued maps, on a network of visually-guided agents in a nonconvex environment. More explicitly, we constrain the motion of agents to sets that (i) ensure that the visibility between two agents are preserved, and (ii) change continuously as a function of the position of the agents. We call such sets convex continuous constraint sets and characterize their properties. Second, based on a discussion on visibility graphs, we define a new proximity graph, called the locally-cliqueless visibility graph, which contains fewer edges than the visibility graph and has the same connected components as the visibility graph. This construction can be, in general, useful for any problem where the connectivity of the visibility graph is important and fewer constraints on the agents, in terms of number of neighbors, is beneficial. Examples of such problems might include line-of-sight wireless routing and consensus problems over line-of-sight wireless communication networks. Third, we propose an algorithm to solve the rendezvous problem and we provide a partial convergence proof.

The paper is organized as follows. Section II discusses the construction of the motion constraint sets. Section III contains the results on the visibility graph. In Section IV,

Anurag Ganguli is with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 1308 W. Main St., Urbana, IL 61801, USA, aganguli@uiuc.edu, http://motion.mee.ucsb.edu/~aganguli

Jorge Cortés is with the Department of Applied Mathematics and Statistics, University of California at Santa Cruz, Santa Cruz, CA 95064, USA, jcortes@ucsc.edu

Francesco Bullo is with the Department of Mechanical and Environmental Engineering, University of California at Santa Barbara, Santa Barbara, CA 93106, USA, bullo@engineering.ucsb.edu

we model the network under consideration, and propose the algorithm for rendezvous and the corresponding convergence analysis. Section V contains experimental results obtained by simulating the algorithm on a computer. In the interest of space, the proofs of only some results in Section II are included here. The rest can be found in an extended technical report at the first author's website.

## II. Convex continuous constraint sets

In this section, we design motion constraint sets for every pair of agents mutually visible to one another. By constraining the motion, we aim to preserve the connectivity of the network. In addition to this, we also require that these sets change continuously as a function of the position of the agents. As we shall see later in Section IV, this enables us to apply the LaSalle Invariance Principle for set-valued maps for the convergence analysis of the algorithm. We must emphasize here that the construction proposed in this section may be applied to *any distributed algorithm for a network of visually-guided agents in a nonconvex environment*.

We begin by reviewing some notation for standard geometric objects. For $p \in \mathbb{R}^2$, we let $\overline{B}(p, r)$ denote the *closed ball* of radius $r \in \mathbb{R}_+$ centered at $p$, respectively. Here, we let $\mathbb{R}_+$ and $\overline{\mathbb{R}}_+$ denote the positive and the nonnegative real numbers, respectively. For a bounded set $X \subset \mathbb{R}^2$, we let $\mathrm{co}(X)$ denote the convex hull of $X$. For $p, q \in \mathbb{R}^2$, we let $]p, q[= \{\lambda p + (1-\lambda)q \mid 0 < \lambda < 1\}$ and $[p, q] = \mathrm{co}(\{p, q\})$ denote the *open* and *closed segment* with extreme points $p$ and $q$, respectively. For a closed convex set $X \subset \mathbb{R}^2$ and any point $q \in \mathbb{R}^2$, let $\mathrm{proj}_X(q)$ denote the orthogonal projection of $q$ onto $X$. For a bounded set $X \subset \mathbb{R}^2$, we let $\mathrm{CC}(X)$ denote the *circumcenter* of $X$, that is, the center of the smallest-radius circle enclosing $X$. Note that the computation of the circumcenter of a bounded set is a strictly convex problem and in particular a quadratically constrained linear program. Let $|X|$ denote the cardinality of a finite set $X$ in $\mathbb{R}^2$.

We now introduce the notion of continuous set-valued maps followed by an important result about the orthogonal projection onto a convex set. For a discussion on set-valued maps, see [7].

**Definition II.1** *Let $X$ and $Y$ be topological vector spaces (real and Hausdorff). A set-valued map $f : X \to 2^Y$ with non-empty and compact values is continuous at a point $x_0 \in X$ if given any $\epsilon > 0$, there exists a $\delta > 0$ such that for all $x \in \overline{B}(x_0, \delta)$, we have*

$$f(x) \subset \bigcup_{y \in f(x_0)} \overline{B}(y, \epsilon) \quad \textit{and} \quad f(x_0) \subset \bigcup_{y \in f(x)} \overline{B}(y, \epsilon)$$

The following lemma is straightforward to verify.

**Lemma II.2** *Let $X$ and $Y$ be topological vector spaces (real and Hausdorff). Let $X \ni x \mapsto f(x) \subset 2^Y$ be a set-valued map with non-empty, convex and compact values that is continuous . Let $X \ni x \mapsto g(x) \in Y$ be a continuous singleton vector valued map. Then the map $X \ni x \mapsto \mathrm{proj}_{f(x)}\, g(x) \in Y$ is continuous.*

Now let us turn our attention to the polygonal environment. Let $Q$ be a simple polygon, possibly nonconvex. We say that a polygon is simple if the polygon vertices are the only points in the plane common to two polygon edges and every polygon vertex belongs to at most two polygon edges. Such a polygon has a well defined interior and exterior. *Note that a simple polygon can contain holes*. Let $\mathcal{Q}$ refer to the set of all simple polygons. Let $\mathrm{Ve}(Q) = (v_1, \dots, v_n)$ be the list of vertices of $Q$ ordered counterclockwise. The *interior angle of a vertex* $v$ of $Q$ is the angle formed inside $Q$ by the two edges of the boundary of $Q$ incident at $v$. The point $v \in \mathrm{Ve}(Q)$ is a *reflex vertex* if its interior angle is strictly greater than $\pi$ radians. Let $\mathrm{Ve_r}(Q)$ denote the list of reflex vertices of $Q$ ordered counterclockwise. On the other hand a point $v \in \mathrm{Ve}(Q)$ is a *convex vertex* if its interior angle is less than or equal to $\pi$ radians.

A point $q \in Q$ is *visible from* $p \in Q$ if $[p, q] \subset Q$. The *visibility polygon* $S(p) \subset Q$ from a point $p \in Q$ is the set of points in $Q$ visible from $p$. It is convenient to think of $p \mapsto S(p)$ as a map from $Q$ to the set of polygons contained in $Q$. It must be noted that the visibility polygon is not necessarily a simple polygon.

**Definition II.3** *Let $v$ be a reflex vertex of $Q$, and let $w \in \mathrm{Ve}(Q)$ be visible from $v$. The $(v, w)$-*generalized inflection segment *$I(v, w)$ is the set*

$$I(v, w) = \{q \in S(v) \mid q = \lambda v + (1-\lambda)w, \lambda \geq 1\}.$$

*If $w \in \mathrm{Ve_r}(Q)$, then with a slight abuse of terms, we call $I(v, w)$ a* bitangent *of $Q$. A reflex vertex $v$ of $Q$ is an* anchor *of $p \in Q$ if it is visible from $p$ and if $\{q \in S(v) \mid q = \lambda v + (1-\lambda)p, \lambda > 1\}$ is not empty.*

In other words, a reflex vertex is an anchor of $p$ if it occludes a portion of the environment from $p$.

**Lemma II.4** *Let $\{I_\alpha\}_{\alpha \in \mathcal{A}}$ be the set of bitangents of $Q$. There exists a unique cover $\{\overline{D}_\beta\}_{\beta \in \mathcal{B}}$ of $Q$ where $D_\beta$ is a connected component of $Q \setminus \bigcup_{\alpha \in A} I_\alpha$ and $\overline{D}_\beta$ denotes its closure.*

Figure 2 illustrates this partition for the given nonconvex polygon.

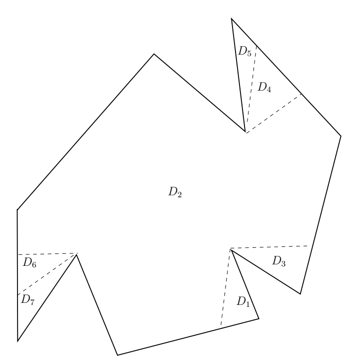


Fig. 2. Partition of $Q$ by the set of bitangents.

**Lemma II.5** *The set-valued map $p \mapsto S(p)$ restricted to $Q \setminus \{\mathrm{Ver_r}(Q) \bigcup (\cup_{\alpha \in \mathcal{A}} I_\alpha)\}$ is continuous.*

Next we define and characterize certain useful convex sets.

**Definition II.6** *Given $Q \in \mathcal{Q}$, let $p, q \in Q$ have the property that $[p, q] \subset Q$. Let $v \in \mathrm{Ver_r}(Q)$. Let $e'_v$ and $e''_v$ be the edges of $Q$ determining $v$. Then we define the set $H_v(p, q) \subset \mathbb{R}^2$ as follows:*

(i) *if $v \notin [p, q]$, then $H_v(p, q)$ is the half-plane with the following properties: (a) the boundary of $H_v(p, q)$ contains $v$ and is perpendicular to the line passing through $v$ and $\mathrm{proj}_{[p,q]} v$, and (b) $p$ and $q$ belong to the interior of $H_v(p, q)$;*

(ii) *if $v = p$ with $p \neq q$, then $H_v(p, q)$ is the half-plane with the following properties: (a) the boundary of $H_v(p, q)$ contains $v$ and is perpendicular to the line passing through $p$ and $q$, and (b) $q$ belongs to the interior of $H_v(p, q)$ (Note: a similar definition holds when we interchange $p$ and $q$);*

(iii) *if $v \in ]p, q[$ with $p \neq q$, then $H_v(p, q)$ is the half-plane with the following properties: (a) the boundary of $H_v(p, q)$ contains the line passing through $p$ and $q$, and (b) the interior of $H_v(p, q)$ intersected with $e'_v$ or with $e''_v$ is empty;*

(iv) *if $v = p = q$, then $H_v(p, q)$ is the set $H'_v \cap H''_v$. $H'_v$ is a half-plane with the following properties: (a) the boundary of $H'_v$ contains the edge $e'_v$, and (b) the interior of $H'_v$ intersected with $e''_v$ is empty. We define $H''_v$ similarly with $e''_v$ interchanged with $e'_v$.*

Figure 3 illustrates the various cases enumerated above.

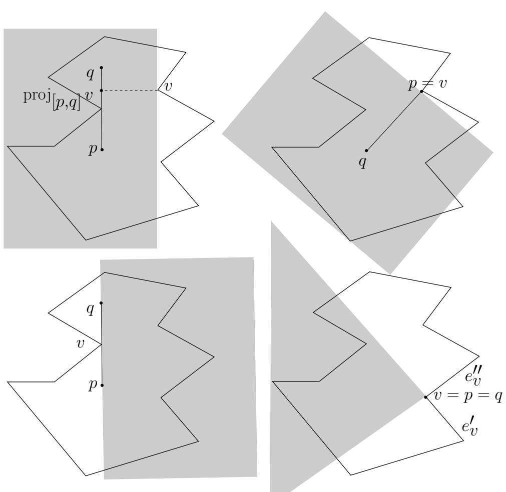


Fig. 3. Definition of the sets $H_v(p, q)$

**Remark II.7** With the above definition, wherever defined, $H_v(p, q)$ is a closed and convex set containing $p$ and $q$. Also, note that if $\mathcal{V}$ is a convex and compact subset of $Q$, then $H_v(p, q)$ is well-defined everywhere in $(\mathcal{V})^2$ and $(p, q) \mapsto H_v(p, q)$ is a set-valued map over the domain $(\mathcal{V})^2$ with range $2^{(\mathbb{R}^2)}$.

**Lemma II.8** *Given any $v \in \mathrm{Ver_r}(Q)$ and a convex and compact subset $\mathcal{V}$ of $Q$, the set-valued map $(p, q) \mapsto H_v(p, q) \cap Q$ restricted to $(\mathcal{V} \setminus \mathrm{Ver_r}(Q))^2$ is continuous.*

**Lemma II.9** *Let $V \subset \mathrm{Ver_r}(Q)$ and $\mathcal{V}$ be a convex and compact subset of $Q$. The following statements are true:*

(i) *the set-valued map $(p, q) \mapsto \bigcap_{v \in V} S(p) \cap H_v(p, q)$ restricted to $\left(\mathcal{V} \setminus (\mathrm{Ver_r}(Q) \bigcup (\cup_{\alpha \in \mathcal{A}} I_\alpha))\right)^2$ is continuous, and*

(ii) *the set-valued map $p \mapsto \bigcap_{v \in V} S(p) \cap H_v(p, p)$ restricted to $\mathcal{V} \setminus (\mathrm{Ver_r}(Q) \bigcup (\cup_{\alpha \in \mathcal{A}} I_\alpha))$ is continuous.*

**Definition II.10 (Convex Continuous Constraint Sets)** *Let $p, q \in Q$ have the property that $[p, q] \subset Q$ and let $I_Q(p, q) = \mathrm{Ver_r}(Q) \cap S(p) \cap S(q)$. The convex continuous constraint set between $p$ and $q$ is*

$$\mathcal{C}_Q(p, q) = \bigcap_{v \in I_Q(p,q)} S(p) \cap H_v(p, q).$$

Figure 4 illustrates the constraint set.

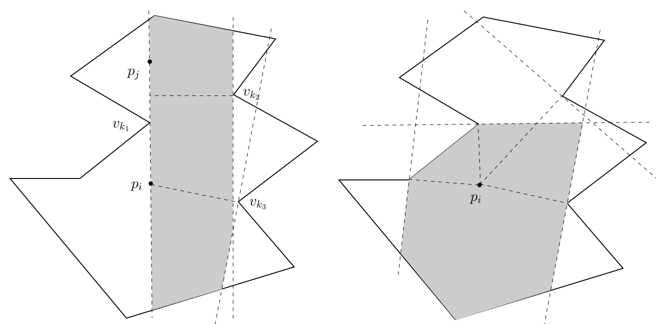


Fig. 4. The figure on the left is an example of the constraint set $\mathcal{C}_Q(p, q)$ where $I_Q(p, q) = \{v_{k_1}, v_{k_2}, v_{k_3}\}$. The figure on the right is an example of $\mathcal{C}_Q(p, p)$ where $I_Q(p, p) = \mathrm{Ver_r}(Q)$.

**Theorem II.11** *Let $\mathcal{V} \subset Q$ be convex and compact. For any two points $p, q \in \mathcal{V}$, the following statements are true:*

(i) *$\mathcal{C}_Q(p, q)$ is convex,*

(ii) *$\mathcal{C}_Q(p, q) = \mathcal{C}_Q(q, p)$, and*

(iii) *the set-valued map $(p, q) \mapsto \mathcal{C}_Q(p, p) \cap \mathcal{C}_Q(p, q)$ restricted to $(\mathcal{V} \setminus \mathrm{Ver_r}(Q))^2$ is continuous.*

*Proof:* We prove fact (i) by induction. Let us first consider a polygon $Q \in \mathcal{Q}$ with $|\mathrm{Ver_r}(Q)| = 1$. Let $\mathrm{Ver_r}(Q) = \{v\}$. It can be easily checked that $C_Q(p, q)$ is convex. Now, let us assume that statement (i) is true for any polygon $Q$ such that $|\mathrm{Ver_r}(Q)| \leq m$. Now let us consider a polygon $Q'$ with $|\mathrm{Ver_r}(Q')| = m + 1$. Since $[p, q] \subset \mathcal{V} \subset Q$, there must exist a vertex $v_{k_1} \in I_{Q'}(p, q)$. Let us take the polygon $H_{v_{k_1}}(p, q) \cap S(p)$. Let us call it $Q''$. Now,

$$\mathcal{C}_{Q'}(p, q) = S(p) \cap H_{v_{k_1}}(p, q) \cap \bigcap_{v \in I_{Q''}(p,q)} H_v(p, q)$$
$$\cap \bigcap_{v \in I_{Q''}(p,q) \setminus \{I_{Q''}(p,q) \cup \{v_{k_1}\}\}} H_v(p, q).$$

The first three terms can be written as $Q'' \cap \mathcal{C}_{Q''}(p,q)$ which is equal to $\mathcal{C}_{Q''}(p,q)$. It suffices to prove that $\mathcal{C}_{Q''}(p,q)$ is convex since the last term in the expression of $\mathcal{C}_{Q'}(p,q)$ is an intersection of half-planes and represents a convex set. Note that we can assume that $Q''$ belongs to $\mathcal{Q}$ (If not, then one can argue that $p$ belongs to a bitangent and in that case we can take $v_{k_1}$ as the nearer of the two vertices defining the bitangent. If $p$ belongs to more than one bitangent, then we can keep on constructing the half-planes till the remaining polygon is simple). Now $|\operatorname{Ve_r}(Q'')| \leq m$ since every time we draw a half-plane $H_v(p,q)$, we remove a reflex vertex. But then by our induction hypothesis, $\mathcal{C}_{Q''}(p,q)$ is convex.

To prove fact (ii), we first claim that $\mathcal{C}_Q(p,q) \subset \mathcal{C}_Q(q,p)$. Let $x \in \mathcal{C}_Q(p,q)$. Then to prove the inclusion, it suffices to show that $x \in S(q)$. Now $x$ and $q$ belong to $\mathcal{C}_Q(p,q)$ which is convex. Therefore $[x,q] \subset \mathcal{C}_Q(p,q) \subset Q$, and hence $x \in S(q)$. The opposite inclusion can be proved identically.

We do not include the proof of fact (iii) here in the interest of space. ■

## III. The locally-cliqueless visibility graph

In Section II we proposed the construction of motion constraint sets to preserve the connectivity of the network. The number of such constraints for an agent is the number of the agents visible to it. It is intuitively clear that the lesser the number of such constraints, the faster will be the convergence of the algorithm. In this section we propose the notion of the locally-cliqueless visibility graph which is a subgraph of the visibility graph. In general it contains fewer number of edges than the visibility graph but has the same number of connected components. In addition we show that this graph can be computed based on the information obtained only from the visibility graph.

We begin by introducing some concepts regarding proximity graphs for point sets in $\mathbb{R}^2$. We assume the reader is familiar with the standard notions of graph theory. We recall that a *clique* of a graph is a complete subgraph of it. A *maximal clique of an edge* is a clique of the graph that (i) contains the edge and (ii) is not a strict subgraph of any other clique of the graph that also contains the edge.

Let us introduce some notation. Given a vector space $\mathbb{V}$, let $\mathbb{F}(\mathbb{V})$ be the collection of finite subsets of $\mathbb{V}$. Accordingly, $\mathbb{F}(\mathbb{R}^2)$ is the collection of finite point sets in $\mathbb{R}^2$; we shall denote an element of $\mathbb{F}(\mathbb{R}^2)$ by $\mathcal{P} = \{p_1, \dots, p_n\} \subset \mathbb{R}^2$, where $p_1, \dots, p_n$ are distinct points in $\mathbb{R}^2$. Let $\mathbb{G}(\mathbb{R}^2)$ be the set of undirected graphs whose vertex set is an element of $\mathbb{F}(\mathbb{R}^2)$.

A *proximity graph function* $\mathcal{G} : \mathbb{F}(\mathbb{R}^2) \to \mathbb{G}(\mathbb{R}^2)$ associates to a point set $\mathcal{P}$ an undirected graph with vertex set $\mathcal{P}$ and edge set $\mathcal{E}_{\mathcal{G}}(\mathcal{P})$, where $\mathcal{E}_{\mathcal{G}} : \mathbb{F}(\mathbb{R}^2) \to \mathbb{F}(\mathbb{R}^2 \times \mathbb{R}^2)$ has the property that $\mathcal{E}_{\mathcal{G}}(\mathcal{P}) \subseteq \mathcal{P} \times \mathcal{P} \setminus \operatorname{diag}(\mathcal{P} \times \mathcal{P})$ for any $\mathcal{P}$. Here, $\operatorname{diag}(\mathcal{P} \times \mathcal{P}) = \{(p,p) \in \mathcal{P} \times \mathcal{P} \mid p \in \mathcal{P}\}$. In other words, the edge set of a proximity graph depends on the location of its vertices. General properties of proximity graphs are defined in [8], [9]. Here, we define:

(i) a *Euclidean Minimum Spanning Tree* of a proximity graph $\mathcal{G}$, denoted $\mathcal{G}_{\text{EMST},\mathcal{G}}$, assigns to each $\mathcal{P}$ a minimum-length spanning tree of $\mathcal{G}(\mathcal{P})$ whose edge $(p_i, p_j)$ is assigned a length $\|p_i - p_j\|$. If $\mathcal{G}(\mathcal{P})$ is not connected, then $\mathcal{G}_{\text{EMST},\mathcal{G}}(P)$ is simply the union of Euclidean Minimum Spanning Trees of its connected components. For simplicity, when $\mathcal{G}$ is the complete graph $(\mathcal{P}, \mathcal{P} \times \mathcal{P} \setminus \operatorname{diag}(\mathcal{P} \times \mathcal{P}))$, we denote the Euclidean Minimum Spanning Tree by $\mathcal{G}_{\text{EMST}}$;

(ii) the *visibility graph* $\mathcal{G}_{\text{vis},Q}$, for $Q \in \mathcal{Q}$, with $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{vis},Q}}(\mathcal{P})$ if the line segment $[p_i, p_j] \in Q$;

(iii) the *locally-cliqueless visibility graph* $\mathcal{G}_{\text{lc-vis},Q}$, for $Q \in \mathcal{Q}$, with $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{lc-vis},Q}}(\mathcal{P})$ if $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{vis},Q}}(\mathcal{P})$ and $(p_i, p_j)$ belongs to a set $\mathcal{E}_{\mathcal{G}_{\text{EMST}}}(\mathcal{P}')$ for any maximal clique $\mathcal{P}'$ of the edge $(p_i, p_j)$ in $\mathcal{G}_{\text{vis},Q}$.

Figure 5 contains some examples of proximity graphs in a nonconvex polygon $Q$ shaped like a typical floor plan. In general, the inclusions in Theorem III.1(i) are strict. Figure 6 shows an example where $\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}} \subsetneq \mathcal{G}_{\text{lc-vis},Q} \subsetneq \mathcal{G}_{\text{vis},Q}$.

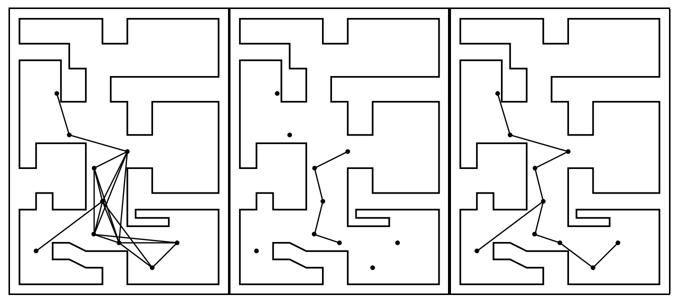

Fig. 5. From left to right, visibility graph, the Euclidean Minimum Spanning Tree for the five agents in the center, and the locally-cliqueless visibility graph.

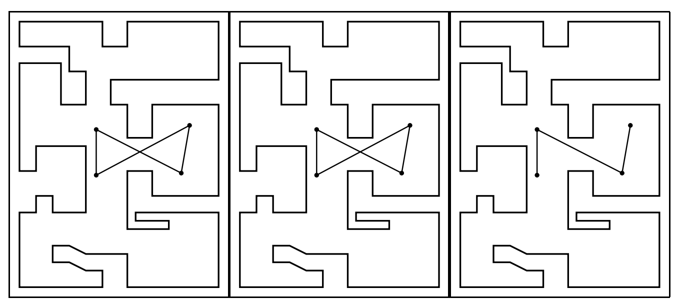

Fig. 6. From left to right, visibility graph, locally-cliqueless visibility graph and Euclidean Minimum Spanning Tree of the visibility graph.

To each proximity graph function $\mathcal{G}$, one can associate the *set of neighbors map* $\mathcal{N}_{\mathcal{G}} : \mathbb{R}^2 \times \mathbb{F}(\mathbb{R}^2) \to \mathbb{F}(\mathbb{R}^2)$, defined by

$$\mathcal{N}_{\mathcal{G}}(p, \mathcal{P}) = \{q \in \mathcal{P} \mid (p,q) \in \mathcal{E}_{\mathcal{G}}(\mathcal{P} \cup \{p\})\}.$$

Typically, $p$ is a point in $\mathcal{P}$, but the definition is well-posed for any $p \in \mathbb{R}^2$. Given $p \in \mathbb{R}^2$, it is convenient to define the map $\mathcal{N}_{\mathcal{G},p} : \mathbb{F}(\mathbb{R}^2) \to \mathbb{F}(\mathbb{R}^2)$ by $\mathcal{N}_{\mathcal{G},p}(\mathcal{P}) = \mathcal{N}_{\mathcal{G}}(p, \mathcal{P})$.

Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be two proximity graph functions. We say that $\mathcal{G}_1$ is *spatially distributed over* $\mathcal{G}_2$ if, for all $p \in \mathcal{P}$,

$$\mathcal{N}_{\mathcal{G}_1,p}(\mathcal{P}) = \mathcal{N}_{\mathcal{G}_1,p}\big(\mathcal{N}_{\mathcal{G}_2,p}(\mathcal{P})\big).$$

It is straightforward to deduce that if $\mathcal{G}_1$ is spatially distributed over $\mathcal{G}_2$, then $\mathcal{G}_1$ is a subgraph of $\mathcal{G}_2$, that is, $\mathcal{G}_1(\mathcal{P}) \subset \mathcal{G}_2(\mathcal{P})$ for all $\mathcal{P} \in \mathbb{F}(\mathbb{R}^2)$.

We say that two proximity graph functions $\mathcal{G}_1$ and $\mathcal{G}_2$ *have the same connected components* if, for all point sets $\mathcal{P}$, the graphs $\mathcal{G}_1(\mathcal{P})$ and $\mathcal{G}_2(\mathcal{P})$ have the same number of connected components consisting of the same vertices.

**Theorem III.1** *For $Q \in \mathcal{Q}$, the following statements hold:*

(i) $\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}} \subset \mathcal{G}_{\text{lc-vis},Q} \subset \mathcal{G}_{\text{vis},Q}$;

(ii) $\mathcal{G}_{\text{lc-vis},Q}$ *is spatially distributed over* $\mathcal{G}_{\text{vis},Q}$;

(iii) $\mathcal{G}_{\text{lc-vis},Q}$ *and* $\mathcal{G}_{\text{vis},Q}$ *have the same connected components.*

*Proof:* The second inclusion in fact (i) is a direct consequence of the definition of $\mathcal{G}_{\text{lc-vis},Q}$. We prove now that $\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}} \subset \mathcal{G}_{\text{lc-vis},Q}$ by contradiction. Let $\mathcal{P} \in \mathbb{F}(Q)$ and assume, without loss of generality, that $\mathcal{G}_{\text{vis},Q}(\mathcal{P})$ is connected (otherwise, the same reasoning carries over for each connected component of $\mathcal{G}_{\text{vis},Q}(\mathcal{P})$). For simplicity, further assume that the distances $\|p_k - p_l\|$, $k, l \in \{1, \ldots, n\}$, $k \neq l$ are all distinct. This ensures that there is a unique Euclidean Minimum Spanning Tree of $\mathcal{G}_{\text{vis},Q}(\mathcal{P})$. If this is not the case, the same reasoning exposed here carries through for each Euclidean Minimum Spanning Tree associated with $\mathcal{G}_{\text{vis},Q}(\mathcal{P})$. Let $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}}}(\mathcal{P})$ and $(p_i, p_j) \notin \mathcal{E}_{\mathcal{G}_{\text{lc-vis},Q}}(\mathcal{P})$. Since necessarily $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{vis},Q}}(\mathcal{P})$, the latter implies that there exists a maximal clique $\mathcal{P}'$ of the edge $(p_i, p_j)$ in $\mathcal{G}_{\text{vis},Q}$ such that $(p_i, p_j) \notin \mathcal{E}_{\mathcal{G}_{\text{EMST}}}(\mathcal{P}')$. If we remove the edge $(p_i, p_j)$ from $\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}}(\mathcal{P})$, the tree becomes disconnected into two connected components $T_1$ and $T_2$, with $p_i \in T_1$ and $p_j \in T_2$. Now, there must exist an edge $e \in \mathcal{E}_{\mathcal{G}_{\text{EMST}}}(\mathcal{P}')$ with one vertex in $T_1$ and the other vertex in $T_2$ and with length strictly less than $\|p_i - p_j\|$. To see this, let $\{e_1, \ldots, e_d\}$ be the edges of $\mathcal{G}_{\text{EMST}}(\mathcal{P}')$ obtained in incremental order by running Prim's algorithm (e.g., see [10]) starting from the vertex $p_i$. Because $p_i$ is in $T_1$ and $p_j$ is in $T_2$, there must exist at least an edge in $\{e_1, \ldots, e_d\}$ with one vertex in $T_1$ and the other vertex in $T_2$. Let $s \in \{1, \ldots, d\}$ be such that $e_s$ is the first edge having one vertex in $T_1$ and another vertex in $T_2$. Since $(p_i, p_j) \notin \mathcal{E}_{\mathcal{G}_{\text{EMST}}}(\mathcal{P}')$, according to Prim's algorithm, the length of $e_s$ must be strictly less than $\|p_i - p_j\|$ (otherwise, the edge $(p_i, p_j)$ will be part of the Euclidean Minimum Spanning Tree of $\mathcal{P}'$). If we add the edge $e_s$ to the set of edges of $T_1 \cup T_2$, the obtained graph $G$ is acyclic, connected and contains all the vertices $\mathcal{P}$, i.e., $G$ is a spanning tree. Moreover, since the length of $e_s$ is strictly less than $\|p_i - p_j\|$ and $T_1$ and $T_2$ are induced subgraphs of $\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}}(\mathcal{P})$, we conclude that $G$ has shorter length than $\mathcal{G}_{\text{EMST},\mathcal{G}_{\text{vis},Q}}(\mathcal{P})$, which is a contradiction. Fact (ii) follows from noticing that, given $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{lc-vis},Q}}(\mathcal{P})$, both agents $i$ and $j$ can compute the maximal cliques of the edge $(p_i, p_j)$ knowing the location of its neighbors in $\mathcal{G}_{\text{vis},Q}$. With this information, each agent can decide if the edge $(p_i, p_j)$ belongs or not to $\mathcal{G}_{\text{lc-vis},Q}$. Finally, fact (iii) is a consequence of fact (i). ■

**Remark III.2** It is a well known fact of combinatorial optimization that finding the maximum clique of a graph is an NP complete problem. However, for a given point set $\mathcal{P}$ and a polygon $Q$, $\mathcal{G}_{\text{lc-vis},Q}(\mathcal{P})$ is easily computable by exploiting the underlying structure of the visibility graph and the Euclidean Minimum Spanning Tree. In fact, it can be shown that for any point $p_i$, $\mathcal{N}_{\mathcal{G}_{\text{lc-vis},Q},p_i}(\mathcal{P})$ can be computed via an $O(m^2)$ algorithm where $m$ is the number of neighbors of $p_i$ in $\mathcal{G}_{\text{vis},Q}(\mathcal{P})$. The algorithm and the corresponding proof will appear in future submissions.

## IV. Rendezvous via proximity graphs

In this section we state the model, the control objective, the motion coordination algorithm, and the properties of the resulting closed-loop system.

### A. A synchronous network of visually-guided agents

We begin by introducing the notions of *visually-guided agent* and of *synchronous network of visually-guided agents*. Let $n$ be the number of agents in the network. Each agent has the following sensing, computation, communication, and motion control capabilities. The $i$th agent has a processor with the ability of allocating continuous and discrete states and performing operations on them. The $i$th agent occupies a location $p_i \in Q$, $Q \in \mathcal{Q}$, and it is capable of moving at any time $m \in \mathbb{N}$, for any unit period of time, according to the discrete-time control system

$$p_i(m+1) = p_i(m) + u_i. \tag{1}$$

We assume that there is a maximum step size $s_{\text{max}} \in \mathbb{R}_+$ common to all agents, that is, $\|u_i\| \leq s_{\text{max}}$, for all $i \in \{1, \ldots, n\}$. The sensing and communication model is the following. Each agent is capable of measuring the relative position of every other agent visible to it, i.e., within line-of-sight. In addition to this, it can also sense the boundary of $Q$. Note that as a consequence, the processor has the capability to answer the query as to whether two agents visible to it are mutually visible to one another.

### B. The rendezvous motion coordination problem

We now state the control design problem for the network of visually-guided agents. The *rendezvous objective* is to achieve agreement over the location of the agents in the network, that is, to steer each agent to a common location. This objective is to be achieved with the limited information flow described in the model above.

Typically, it will be impossible to solve the rendezvous problem if the agents are placed in such a way that they do not form a connected graph. Arguably, a good property of any algorithm to rendezvous is that of maintaining some form of connectivity between agents.

### C. The Circumcenter Algorithm

Here is an informal description of what we shall refer to as the *Circumcenter Algorithm over a proximity graph* $\mathcal{G}$:

> Each agent performs the following tasks: (i) it detects its neighbors according to $\mathcal{G}$; (ii) it computes the circumcenter of the point set comprised of its neighbors and of itself, and (iii) it moves toward this circumcenter while maintaining connectivity with its neighbors.

This algorithm is inspired by the one introduced in [1]. Let us clarify which proximity graphs are allowable and how connectivity is maintained. Firstly, we are allowed to design motion coordination algorithms that are spatially distributed over the visibility graph $\mathcal{G}_{\text{vis},Q}$, or more generally, over any proximity graph $\mathcal{G}$ that is spatially distributed over $\mathcal{G}_{\text{vis},Q}$. This is a direct consequence of our modeling assumption that each agent can acquire the location of every other agent visible to it. Secondly, we maintain connectivity by restricting the allowable motion of each agent. In particular, we will show that it suffices to restrict the motion of each agent as follows. If agents $p_i$ and $p_j$ are neighbors in the proximity graph $\mathcal{G}$, then their subsequent positions are required to belong to $\mathcal{C}_Q(p_i, p_j)$ as defined in Theorem II.11.

If an agent $p_i$ has its neighbors at locations $\{q_1, \dots, q_l\}$, then define $\mathcal{M}_i = \{q_1, \dots, q_l\} \cup \{p_i\}$. We define the *constraint set* $C_{p_i,Q}(\mathcal{M}_i)$ by

$$C_{p_i,Q}(\mathcal{M}_i) = \bigcap_{q \in \mathcal{M}_i} \mathcal{C}_Q(p_i, q).$$

The following remark informally describes what we have accomplished by thus designing the constraint sets.

**Remark IV.1**
- $C_{p_i,Q}(\mathcal{M}_i)$ is a convex subset of $Q$ containing $p_i$. This follows from the definition of $C_{p_i,Q}(\mathcal{M}_i)$ and Theorem II.11 (i).
- If $\mathcal{M}_i \cap \text{Ve}_\text{r}(Q)$ is empty and the set of neighbors of $p_i$ is fixed, then $C_{p_i,Q}(\mathcal{M}_i)$ changes continuously as a function of $p_i$ and of the positions of its neighbors. This follows from the fact that for each $p_j \in \mathcal{M}_i$, $p_i$ is constrained to remain in $\mathcal{C}_Q(p_i, p_j)$ which is a convex and compact subset of $Q$. The statement is then a consequence of Theorem II.11 (iii) and the fact that $C_{p_i,Q}(\mathcal{M}_i)$ is an intersection of continuous maps.

With this, we are ready to formally describe the algorithm.

| | |
|---|---|
| **Name:** | Circumcenter Algorithm over $\mathcal{G}$ |
| **Goal:** | Solve the rendezvous problem |
| **Assumes:** | (i) $s_{\max} \in \mathbb{R}_+$ is maximum step size<br>(ii) $Q \in \mathcal{Q}$<br>(iii) $\mathcal{G}$ is a spatially distributed proximity graph over $\mathcal{G}_{\text{vis},Q}$ |

For $i \in \{1, \dots, n\}$, agent $i$ executes the following at each time instant in $\mathbb{N}$:

1: acquire $\{q_1, \dots, q_k\} := \mathcal{N}_{\mathcal{G}_{\text{vis},Q}, p_i}(\mathcal{P})$
2: compute $\mathcal{M}_i := \mathcal{N}_{\mathcal{G},p_i}(\{q_1, \dots, q_k\}) \cup \{p_i\}$
3: compute $X_i := C_{p_i,Q}(\mathcal{M}_i) \cap \text{co}(\mathcal{M}_i)$
4: compute $q_i^* := \text{proj}_{X_i}(\text{CC}(\mathcal{M}_i))$
5: $u_i := \frac{\min(s_{\max}, \|q_i^* - p_i\|)}{\|q_i^* - p_i\|}(q_i^* - p_i)$

See Figure 7 for examples of the constraint sets $C_{p_i,Q}(\mathcal{M}_i)$ defined above.

In what follows we shall refer to the Circumcenter Algorithm over the proximity graph $\mathcal{G}$ as the map $T_{\mathcal{G}} : Q^n \to Q^n$.

### D. Asymptotic correctness of the Circumcenter Algorithm

In what follows, $P$ shall refer to tuples of elements in $\mathbb{R}^2$ of the form $(p_1, \dots, p_n)$. With a slight abuse of notation, henceforth we shall use $P$ interchangeably with a point set $\mathcal{P}$ of the form $\{p_1, \dots, p_n\}$. Before proceeding to analyze the convergence properties of the Circumcenter Algorithm, let us first define a Lyapunov function. We define the function $V_{\text{diam}} : Q^n \to \overline{\mathbb{R}}_+$, by

$$V_{\text{diam}}(P) = \max\{\|p - q\| \mid p, q \in \text{co}(P)\}.$$

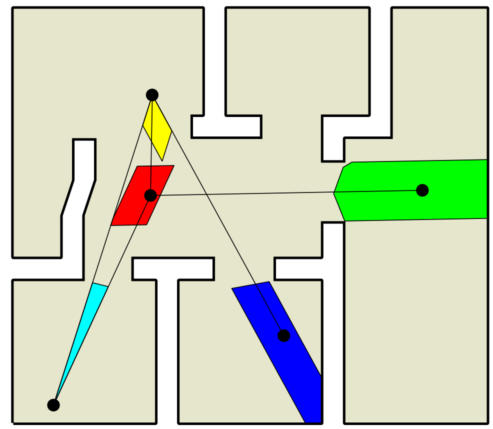

Fig. 7. Constraint sets $C_{p_i,Q}(\mathcal{M}_i)$ generated by the algorithm encoded as described in Section V

We shall also require, at some times, to make the following assumption on a sequence $\{P_m\}_{m \in \mathbb{N} \cup \{0\}} \subset Q^n$:

(A) There exists a compact set $\mathcal{X} \subset (Q \setminus \text{Ve}_\text{r}(Q))$ such that $\{P_m\}_{m \in \mathbb{N} \cup \{0\}} \subset \mathcal{X}^n$.

We are now ready to state the following convergence result.

**Theorem IV.2** *Let $p_1, \dots, p_n$ be a network of visually-guided agents in $Q \in \mathcal{Q}$, with maximum step size $s_{\max} \in \mathbb{R}_+$. Assume that $Q$ does not contain any holes, and that the proximity graph $\mathcal{G}$ is spatially distributed over $\mathcal{G}_{\text{vis},Q}$ and has the same connected components as $\mathcal{G}_{\text{vis},Q}$. Given $P \in Q^n$, let $P_{\mathcal{C}}$ refer to the locations of agents in any connected component $\mathcal{C}$ of $\mathcal{G}_{\text{vis},Q}(P)$. Then, any trajectory $\{P_m\}_{m \in \mathbb{N} \cup \{0\}}$ of $T_{\mathcal{G}}$ has the following properties:*

(i) *if the locations of two agents belong to the same connected component of $\mathcal{G}_{\text{vis},Q}(P_k)$ for some $k \in \mathbb{N} \cup \{0\}$, then they remain in the same connected component of $\mathcal{G}_{\text{vis},Q}(P_m)$ for all $m \geq k$,*

(ii) *$\text{co}(P_m) \subseteq \text{co}(P_{m-1})$, for all $m \in \mathbb{N}$, and*

(iii) *if $\{P_m\}_{m \in \mathbb{N} \cup \{0\}}$ satisfies (A), then $\{P_m\}_{m \in \mathbb{N} \cup \{0\}}$ converges to the largest weakly invariant set contained in*

$$\mathcal{X}^n \cap \{P \in Q^n \mid \exists P' \in T_{\mathcal{G}}(P) \text{ such that } V_{\text{diam}}(P'_{\mathcal{C}}) = V_{\text{diam}}(P_{\mathcal{C}}) = a_{\mathcal{C}} \text{ for all connected components } C \text{ of } \mathcal{G}_{\text{vis},Q}(P)\},$$

*for some constants $a_{\mathcal{C}} \in \overline{\mathbb{R}}_+$.*

**Remark IV.3** For a network that has only one connected component, extensive computer simulations (see Section V) have shown that the network converges to a point of the form $(p^*, \dots, p^*) \in \mathcal{X}^n$. The complete theoretical proof of this observation is yet to be completed and will be included in future submissions.

Before presenting the proof for Theorem IV.2, let us introduce some useful results. The technical approach in what follows is similar to the one in [9].

To a proximity graph function $\mathcal{G}$ that is spatially distributed over $\mathcal{G}_{\text{vis},Q}$, and a configuration $P \in Q^n$, one may associate a graph $G_{\mathcal{G}(P)} = (\{1,\dots,n\}, E)$ by defining $(i,j) \in E$ if $(p_i, p_j)$ is an edge of $\mathcal{G}(P)$. Clearly, for each $P \in Q^n$, $P(\mathcal{N}_{G_{\mathcal{G}(P)}}(i))$ is equal to the set of neighbors of $p_i$ with respect to the graph $\mathcal{G}(P)$.

Given an undirected graph $G = (\{1,\dots,n\}, E)$, define the *Circumcenter Algorithm at Fixed Topology* $T_G : Q^n \to Q^n$ whose $i$th component is

$$(T_G)_i(p_1,\dots,p_n) = (T_{\mathcal{G}})_i(p_1,\dots,p_n).$$

**Lemma IV.4** *For $G = (\{1,\dots,n\}, E)$, the map $T_G : Q^n \to Q^n$ has the following properties:*

(i) *The map $P \mapsto T_G(P)$ restricted to $(Q \setminus \mathrm{Ve}_{\mathrm{r}}(Q))^n$ is continuous, and*
(ii) $\mathrm{co}(T_G(P)) \subseteq \mathrm{co}(P)$, *for $P \in Q^n$.*

*Proof:* Statement (i) is a consequence of Lemma II.2 and Theorem II.11 (iii). From the description of the algorithm in Section IV-C, we have that $(T_G)_i(P) \in [p_i, q_i^*]$. Now $q_i^* \in X_i$. Also $p_i \in X_i$ since $p_i \in C_{p_i,Q}(\mathcal{M}_i)$ (see Remark IV.1) and trivially $p_i \in \mathrm{co}(\mathcal{M}_i)$. But $X_i$ is convex since it is an intersection of two convex sets. Hence, $[p_i, q_i^*] \subset X_i$ and in particular $(T_G)_i(P) \subset X_i$. The following chain of inclusions is again trivially true: $X_i \subset \mathrm{co}(\mathcal{M}_i) \subset \mathrm{co}(P)$. Thus, we have that $(T_G)_i(P) \subset \mathrm{co}(P)$ for all $i \in \{1,\dots,n\}$ and hence $\mathrm{co}(T_G(P)) \subset \mathrm{co}(P)$. ■

Given $Q \in \mathcal{Q}$, define the *Circumcenter Algorithm at All Connected Topologies* $T : Q^n \to 2^{(Q^n)}$ by

$$T(P) = \{T_G(P) \in Q^n \mid G = (\{1,\dots,n\}, E) \text{ is connected}\}.$$

**Proposition IV.5** *For $Q \in \mathcal{Q}$, the map $T : Q^n \to 2^{(Q^n)}$ has the following properties:*

(i) *the map $P \mapsto T(P)$ restricted to $\mathcal{X}$, a compact subset of $(Q \setminus \mathrm{Ve}_{\mathrm{r}}(Q))^n$, is upper semicontinuous, and*
(ii) $\mathrm{co}(T(P)) \subseteq \mathrm{co}(P)$, *for $P \in Q^n$.*

*Proof:* We first prove statement (i). Since $\mathcal{X}$ is compact and $T$ is bounded on a neighborhood of $\mathcal{X}$, then from [7] [Page 66, Lemma 14] it suffices to prove that $T$ is closed on $\mathcal{X}$. The proof now follows on the same lines as Proposition 4.3 (ii) in [9]. Statement (ii) follows from Lemma IV.4 (ii). ■

Now that we have analyzed the smoothness of algorithm $T$, let us now study the properties of the Lyapunov function $V_{\text{diam}} : Q^n \to \overline{\mathbb{R}}_+$.

**Lemma IV.6** *The function $V_{\text{diam}} : Q^n \to \overline{\mathbb{R}}_+$ has the following properties:*

(i) *$V_{\text{diam}}$ is continuous, and is invariant under permutations of its arguments;*
(ii) *$V_{\text{diam}}(P) = 0$ if and only if $P = (p_1,\dots,p_n) \in Q^n$ is such that $p_i = p_j$ for all $i,\ j \in \{1,\dots,n\}$;*
(iii) *$V_{\text{diam}}$ is non-increasing along $T$.*

*Proof:* Facts (i) and (ii) are straightforward to verify. Proposition IV.5 (ii) implies fact (iii). ■

We now present the asymptotic convergence properties of the algorithm $T$. The proof of this relies on a discrete-time LaSalle Invariance Principle for set-valued maps; see [9].

**Lemma IV.7** *Let $Q \in \mathcal{Q}$. Assume that $Q$ does not contain any holes, and that the proximity graph $\mathcal{G}$ is spatially distributed over $\mathcal{G}_{\text{vis},Q}$ and has the same connected components as $\mathcal{G}_{\text{vis},Q}$. Then, any sequence $\{P_m\}_{m\in\mathbb{N}\cup\{0\}}$, defined by $P_{m+1} \in T(P_m)$ and satisfying Assumption (A), converges to the largest weakly invariant set contained in $\mathcal{X} \cap \{P \in Q^n \mid \exists P' \in T(P)$ such that $V_{\text{diam}}(P') = V_{\text{diam}}(P) = a\}$ where $a \in \overline{\mathbb{R}}_+$.*

*Proof:* From (A), we have that any sequence $\{P_m\}_{m\in\mathbb{N}\cup\{0\}}$ belongs to $\mathcal{X}^n$. From Proposition IV.5, we know that the algorithm $T$ is upper semicontinuous on $\mathcal{X}^n$ (with non-empty and compact values) and hence upper semicontinuous on $\mathcal{X}^n$. We now use $V_{\text{diam}} : Q^n \to \overline{\mathbb{R}}_+$ as a candidate Lyapunov function (see Lemma IV.6). Then the result follows from LaSalle's Invariance Principle. ■

We are now ready to state the proof of Theorem IV.2.

*Proof of Theorem IV.2:* We start by proving fact (i). Let $k \in \mathbb{N}\cup\{0\}$ and take $\mathcal{C}$ a connected component of $\mathcal{G}_{\text{vis},Q}(P_k)$. By assumption, $\mathcal{G}$ and $\mathcal{G}_{\text{vis},Q}$ have the same connected components, and therefore $\mathcal{C}$ is also a connected component of $\mathcal{G}(P_k)$. By definition of $T_{\mathcal{G}}$, if agents $i$ and $j$ are neighbors according to the graph $\mathcal{G}(P_k)$, then $(p_i)_{k+1}, (p_j)_{k+1} \in C_{p_i,Q}(\mathcal{M}_i)$. Since $C_{p_i,Q}(\mathcal{M}_i)$ is a convex subset of $Q$ which in particular implies that $[(p_i)_{k+1}, (p_j)_{k+1}] \subset Q$, we have that the agents in $\mathcal{C}$ remain connected in the visibility graph at step $k+1$, i.e., the agents in $\mathcal{C}$ are contained in the same connected component of $\mathcal{G}_{\text{vis},Q}(P_{k+1})$.

Now, let $P'_k$ refer to the location of the agents in $\mathcal{C}$ for $k \in \mathbb{N}$. Fact (ii) is an extension of the observation that $\mathrm{co}(P'_k) \subseteq \mathrm{co}(P'_{k-1})$ which in turn follows from Proposition IV.5 (ii).

Now, let us prove fact (iii). From (i), we deduce that the number of vertices in each of the connected components of $\mathcal{G}(P_m)$ is non-decreasing. Since there is a finite number of agents, there must exist $m_0$ such that the identity of the agents in each connected component is fixed for all $m \geq m_0$ (i.e., no more agents are added to the connected component afterward). Let $\mathcal{C} = \{p_{i_1},\dots,p_{i_K}\}$ be any of these connected components. The result follows by noting that Lemma IV.7 is applicable to the agents in $\mathcal{C}$ (since their evolution under $T_{\mathcal{G}}$ is one of the many possible evolutions under the algorithm $T$, see the definition of $T$). ■

### *E. A variant of the Circumcenter Algorithm*

In Section IV-D, we conjecture that the Circumcenter Algorithm solves the rendezvous problem for visually-guided agents if the network evolves in a compact subset of $Q \setminus \mathrm{Ve}_{\mathrm{r}}(Q)$. In what follows we describe an algorithm that we conjecture guarantees convergence without this assumption.

**Name:** Modified Circumcenter Algorithm over $\mathcal{G}$
**Goal:** Solve the rendezvous problem
**Assumes:** (i) $s_{\max} \in \mathbb{R}_+$ is maximum step size
(ii) $Q \in \mathcal{Q}$
(iii) $\mathcal{G}$ is a spatially distributed proximity graph over $\mathcal{G}_{\text{vis},Q}$ with the property that two agents at the same location have identical sets of neighbors.

For $i \in \{1, \dots, n\}$, agent $i$ executes the following at each time instant in $\mathbb{N}$:

acquire $\{q_1, \dots, q_k\} := \mathcal{N}_{\mathcal{G}_{\text{vis},Q}, p_i}(\mathcal{P})$
compute $\mathcal{W}_i := \{q_j \mid q_j = p_i,\ j \in \{1, \dots, n\}\}$
compute $\mathcal{B}_i := (\mathcal{N}_{\mathcal{G}, p_i}(\{q_1, \dots, q_k\}) \setminus \mathcal{W}_i)$
compute $\mathcal{M}_i := \mathcal{B}_i \cup \{p_i\}$
**if** $\mathcal{B}_i = \{v\}$, for $v \in \text{Ve}_{\text{r}}(Q)$, and $p_i \notin \text{Ve}_{\text{r}}(Q)$ **then**
    compute $q_i^* := v$
**else**
    compute $X_i := C_{p_i,Q}(\mathcal{M}_i) \cap \text{co}(\mathcal{M}_i)$
    compute $q_i^* := \text{proj}_{X_i}(\text{CC}(\mathcal{M}_i))$
**end if**
$u_i := \frac{\min(s_{\max}, \|q_i^* - p_i\|)}{\|q_i^* - p_i\|}(q_i^* - p_i)$

**Remark IV.8** The graph $\mathcal{G}_{\text{lc-vis},Q}$ fulfills assumption (iii) in the statement of the Modified Circumcenter Algorithm.

## V. Simulation results

To conduct experiments, a simulation environment has been developed in `Matlab` ®. The code is organized in two layers. The lower layer consists of a library containing routines to answer queries such as whether two points in a two dimensional polygonal environment are visible to each other. The higher layer utilizes these routines. The controller which computes the goal for each agent at every time instant is implemented in the higher layer. Note that other visibility based algorithms for single or multiple agents can be easily implemented within this framework. This can be done by extracting the appropriate information using the low level functions and implementing the desired controller.

Figures 1, 8 and 9 illustrate the performance of the Circumcenter Algorithm in Section IV-C.

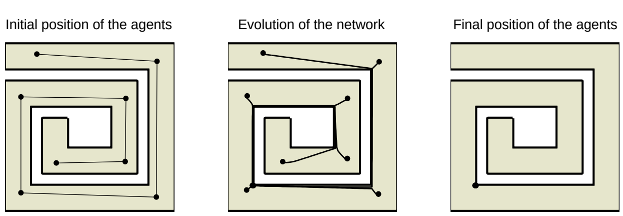


Fig. 8. Simulation results of the Circumcenter Algorithm on a network of agents distributed in a spiral polygon. The locations of the agents, at all times, do not belong to reflex vertices. However, at some instants, reflex vertices are approached very closely. The algorithm is run over $\mathcal{G}_{\text{vis},Q}$.

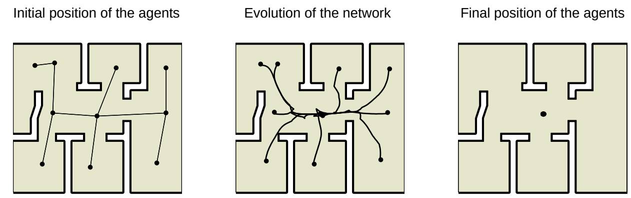


Fig. 9. Simulation results of the Circumcenter Algorithm on a network of agents distributed in a polygon shaped like a typical floor plan. The algorithm is run over $\mathcal{G}_{\text{lc-vis},Q}$.

## VI. Conclusions

This paper focuses on the distributed control of synchronous networks of visually-guided robotic agents. We have defined some useful geometric quantities, such as continuous constraint sets and generalized visibility graphs, and studied circumcenter algorithms for rendezvous. We have provided a partial convergence proof as well as successful numerical simulations. Future work will involve completion of the proofs of the results in this paper, as well as coordination algorithms for deployment and search.

*Acknowledgments:* This material is based upon work supported in part by AFOSR through Award F49620-02-1-0325, by ONR through YIP Award N00014-03-1-0512, and by faculty research funds granted by the University of California, Santa Cruz. The authors thank Prof. Jana Kosěcká for an early inspiring discussion on the subject of this paper.

## References

[1] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita, "Distributed memoryless point convergence algorithm for mobile robots with limited visibility," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 5, pp. 818–828, 1999.

[2] J. Lin, A. S. Morse, and B. D. O. Anderson, "The multi-agent rendezvous problem: an extended summary," in *Proceedings of the 2003 Block Island Workshop on Cooperative Control*, ser. Lecture Notes in Control and Information Sciences, V. Kumar, N. E. Leonard, and A. S. Morse, Eds. New York: Springer Verlag, 2004, vol. 309, pp. 257–282.

[3] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, "Gathering of asynchronous oblivious robots with limited visibility," in *STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science (Dresden, Germany)*, ser. Lecture Notes in Computer Science, A. Ferreira and H. Reichel, Eds. New York: Springer Verlag, 2001, vol. 2010, pp. 247–258.

[4] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 622–629, 2004.

[5] C. Geyer and K. Daniilidis, "Mirrors in motion: Epipolar geometry and motion estimation," in *IEEE International Conference on Computer Vision*, Nice, France, Oct. 2003, pp. 766–773.

[6] J. Hayes, M. McJunkin, and J. Kosěcká, "Communication enhanced navigation strategies for teams of mobile agents," in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, Las Vegas, NV, Oct. 2003, pp. 2285–2290.

[7] A. F. Filippov, *Differential Equations with Discontinuous Righthand Sides*, ser. Mathematics and Its Applications. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1988, vol. 18.

[8] J. W. Jaromczyk and G. T. Toussaint, "Relative neighborhood graphs and their relatives," *Proceedings of the IEEE*, vol. 80, no. 9, pp. 1502–1517, 1992.

[9] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in $d$ dimensions," *IEEE Transactions on Automatic Control*, June 2004, submitted. Electronic version available at `http://www.me.ucsb.edu/bullo`.

[10] N. A. Lynch, *Distributed Algorithms*. San Mateo, CA: Morgan Kaufmann Publishers, 1997.