2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Decentralized Centroid Estimation for Multi-Agent Systems in Absence of any Common Reference Frame

Mauro Franceschelli[†] and Andrea Gasparri[⋆]

[†]Electrical and Electronic Engineering
University of Cagliari
09123 Cagliari, Italy
mauro.franceschelli@diee.unica.it

[⋆]Computer Science and Automation
University of "Roma Tre"
00146 Roma Italy
gasparri@dia.uniroma3.it

Abstract— **In this paper, a novel distributed algorithm to deal with the problem of estimating the network centroid in a multi-agent system is proposed. In this scenario, agents are assumed to be lacking any global reference frame or absolute position information. The proposed algorithm can be thought as a general tool to retrieve information about the centroid of a network of agents. Indeed, this allows to release several simplifying assumptions for a significant family of algorithms dealing with decentralized motion coordination. The convergence properties of the algorithm are carefully investigated in the case of a fully connected network for which a proof of convergence is provided. Successively, simulations to show the effectiveness of the algorithm also for arbitrary undirected connected graphs are given.**

I. Introduction

In recent years multi agent systems have drawn the attention of a huge amount of researchers, for a representative example see [1], [2], [3], [4]. In this framework Laplacian based controllers [5], [6], [7], [8] have been studied in many forms and applications, for instance rendezvous [9], leader following [10], attitude control [11] and many others [12], [13], [14]. The majority of these algorithms, dealing with decentralized motion coordination problems, assume that the agents have access to absolute position information (GPS) and thus have a common global reference that makes it easy to interpret the information passed by other agents. Even when in multi agent systems the agents are not supposed to know their absolute position, many times they are assumed to have a common attitude reference to exchange information that can be achieved by using a compass, gyroscopes and occasionally gravity as common reference for their coordinate system. For space applications another technological solution is to use a frame of fixed stars to have a common reference. In all these instances several technological countermeasures have to be undertaken for the implementation of coordination algorithms increasing the total costs of the single agents. Many coordination algorithms rely on local information taken by the neighbors in the sensing radius of the agents and some general information about the swarm of mobile units, for instance its centroid. In this paper we are interested in developing an asynchronous and decentralized algorithm to estimate the location of the centroid of the network of agents that does not require any common attitude reference nor any absolute position information. We believe that removing such hidden assumption could significantly advance the technological feasibility of mobile swarm of agents, reducing their dependence on the global positioning system in the low level control loops. Furthermore in many space applications, where networks of mobile robots are envisioned in the so not distant future, the absence of the need for absolute position information or a common coordinate system could prove to be an essential robust feature.

The rest of the paper is organized as follows. In Section I-A the scenario along with simplifying assumptions is introduced. In Section II a more formal description of the problem formulation is given. In Section III the proposed algorithm is described and the proof of its convergence in case a fully connected graph is given. In Section IV, the performance of the algorithm for several network topologies is discussed. Finally, in Section V conclusions are drawn and future work is discussed.

A. Paper content

In this paper, a decentralized asynchronous algorithm to estimate the centroid of a network of agents is proposed. It allows to release several simplifying assumption for a large family of algorithms proposed in literature dealing with decentralized motion coordination.

In particular, the following assumptions are made:

- The network of agents can be described by a connected undirected graph.
- Each agent can only communicate with agents directly connected to it.
- Communications are asynchronous, i.e. communication failures or delays do not affect the network.
- Each agent is able to sense the distance between itself and its neighbors.
- Each agent is able to sense the direction in which it sees its neighbors with respect to its local reference frame, arbitrary fixed on it.

Objective of the proposed algorithm is to make available to each agent an estimate of where the network centroid is with respect to the local agents reference frame. The convergence properties of the algorithm are carefully investigated in the case of a fully connected network for which a proof of convergence is provided. Successively, simulations to show the effectiveness of the algorithm also for arbitrary undirected connected graphs are given. Although, the connectedness of the network proved not to be a sufficient condition for the convergence of the proposed algorithm, some promising topologies showing good convergence results are discussed. A characterization of the set of all the graphs in which the proposed algorithm converges is left to future research.

II. Problem description

Let us consider a network of agents with limited sensing capabilities. Each agent, which is characterized by a position in a d-dimensional space, is able to collaborate with the neighboring agents, i.e., agents that are within its range of sensing. Information exchanged between agents is only the distance between them and the direction of the line of sight with respect to their local reference frame.

From a mathematical standpoint, the graph theory [15] provides a suitable framework for describing such a problem by exploiting the definition of proximity graph as in [6]. Let the network of agents be described by $\mathcal{G} = (V, E)$, where $V = \{1, \ldots, n\}$ is the set of nodes (agents) and $E \subseteq \{V \times V\}$ is the set of edges (connectivity). A position $p_i \in \mathbb{R}^d$ in the d^{th} dimensional space is associated to each node $v_i \in V$, with $i = 1, \ldots, n$. In particular, an edge representing a connection between two agents exists if and only if the distance between these agents is less then or equal to their sensing radius r, namely

$$E = \{(i,j) : \ \|p_i - p_j\| \leq k, \forall i,j = 1,\ldots,n \ i \neq j\},$$

where $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^d$. Therefore, a generic couple of agents i, j is able to sense $\|p_i - p_j\|$ reciprocally. In addition, each agent has a local reference frame defined by an orthonormal basis of vectors in $\mathbb{R}^d$ fixed on it and, is able to determine the direction in which neighbors are sensed, strictly with respect to its own local reference frame.

By introducing a rotation matrix R_i for each agent i, is possible to express the node's network centroid estimate e_i with respect to a global reference frame as follows:

$$e_{gi} = R_i e_i + p_i.$$

Now, by assuming the origin of the arbitrary common global coordinate systems O_g to be at the centroid of the network, the problem formulation can be stated as follows.

Problem Statement: Given the following initial condition for each node $v_i \in V$:

$$e_{gi}(t_0) = R_i e_i(t_0) + p_i = p_i,$$

the following condition must be achieved for each agent under the assumptions I-A:

$$e_{gi}(\infty) = R_i e_i(\infty) + p_i = O_g = \frac{\sum_{i=1}^{n} p_i}{n}$$

Note that, a virtual global reference frame unknown by the agents is used for analysis purposes.

III. Consensus Algorithm with unknown global reference frame

In this section, a decentralized algorithm for the estimation of the network centroid under assumptions I-A is given. First, a convenient terminology is proposed, successively the main concepts of the algorithms are described and finally the prove of its convergence for a fully connected network is given.

Let us define the direction for which agent i is able to sense agent j with respect to its local frame as

$$c_{ij} = R_i^T \frac{(p_j - p_i)}{\|p_j - p_i\|},$$

clearly, the following property holds $R_i c_{ij} = -R_j c_{ji}$. In addition, let the relative distance between two agents i and j be:

$$d_{ij} = d_{ji} = \|p_i - p_j\|_2.$$

Finally, by assuming the network of agents to be deployed in a d-dimensional space, let $\mathbf{0}$ be the d-elements vectors of zeros.

***Algorithm** 1:*

Let $e_i(t_0) = \mathbf{0}$, $\forall \ i = 1, \ldots, n$ with $t_0 = 0$. At each time-step t:

1) Select an edge at random $(i, j) \in E$.
2) Compute $e_i(t)^T c_{ij}$ and $e_j(t)^T c_{ji}$,
3) Update nodes estimate according to:

$$e_i(t+1) = e_i(t) + \Delta_{ij}(t) \cdot c_{ij},$$

$$e_j(t+1) = e_j(t) + \Delta_{ji}(t) \cdot c_{ji}.$$

with

$$\Delta_{ij}(t) = \left(\frac{d_{ij} + e_j(t)^T c_{ji} - e_i(t)^T c_{ij}}{2} + e_i(t)^T c_{ij} \right),$$

$$\Delta_{ji}(t) = \left(\frac{d_{ji} + e_i(t)^T c_{ij} - e_j(t)^T c_{ji}}{2} + e_j(t)^T c_{ji} \right)$$

It is important to notice how this algorithm lends itself to an easy decentralized implementation, where collaboration between couple of agents can be safely performed asynchronously.

In other words, after an initialization step where each agent assumes its own location to be the network centroid, namely $e_i = 0$ and thus $e_{gi} = p_i$ in the global reference frame, the algorithm iterates as follows at each time-step:

- A communication link between two nodes is activated. The edge selection process deeply affects the convergence rate of the algorithm and the network topology affects both the convergence rate and the convergence to a common estimate. In this paper, each edge has a strictly positive probability to be chosen at each instant of time. The following edge selection process has been exploited during simulations: at each time step, an agent is selected at random using a uniform probability

distribution, it then communicates with all its neighbors in random order.
- The two "activated" agents estimate the relative position between each other, namely their distance and the line of sight both in their own local reference frame. They then compute the projection of their current estimate of the network centroid with respect to the line of sight between them and transmit this scalar value to their companion.
- The two agents then update their estimates independently by averaging between their projections on the line of sight and updating their estimate of the network centroid along the direction of the line of sight.
- Another edge is selected and the process repeats itself, after some time each node will have a good estimate with respect to its local reference frame.

Note that, such iterative algorithm makes all nodes converge to the correct estimate of the centroid if some assumptions on the topology hold. It should be also noticed that, for sake of clarity, the description of the algorithm was referred only to a single couple of agents. Indeed, in a real-context different couples of agents might perform the algorithm at the same time in different portion of the network. This not only drastically affects the convergence time but it also highlights the inherent parallelism of the proposed algorithm. Moreover, in a real implementation a distributed edge selection process can be naturally obtained by letting each node individually schedule its transmitting time according to any deterministic or random schema.

It will now be proved that, if the graph representing the network is fully connected, and each edge has a strictly positive probability to be chosen at each instant of time, then the probability that all the agents agree on where is the network centroid goes to one as time goes to infinity.

***Theorem** 1:* If the network of agents is fully connected then

$$\forall i \in V,\ Pr\left(\lim_{t\to\infty} e_{gi}(t) = \frac{\sum_{i=1}^{n} p_i}{n}\right) = 1$$

Proof:

Given a suitable Lyapunov-like function of the state of the network, the proof relies on a probabilistic argument to show that such a function converges to zero as time goes to infinity almost surely, depending on the edge selection process.

By considering two generic agents i and j, the generic configuration at time t given in Figure 1 is used as support for the proof.

Let O_g be the coordinates of the network centroid in the common global reference frame. Then if the estimates of the agents are updated using Algorithm 1 and we choose

$$V(t) = \sum_{i=1}^{n} \|e_{gi}(t) - O_g\|_2^2$$

as a candidate Lyapunov function. $V(t)$ is a quadratic function and so is positive for any $t \geq 0$. The following manipulations show that $V(t+1) \leq V(t)$.

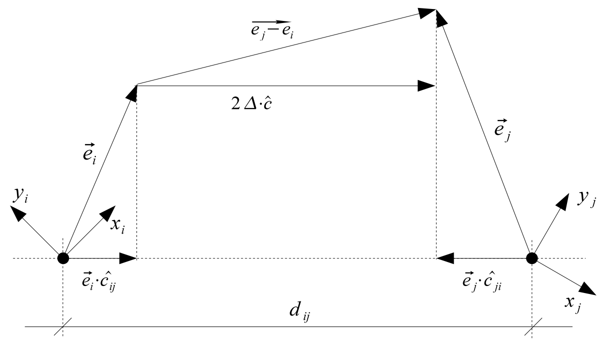

Fig. 1. Example of algorithm iteration involving two nodes.

At time t only nodes i, j change their estimation, thus possibly changing the value of $V(t+1)$. Thus we have

$$\begin{aligned} V(t+1) - V(t) = {} & \|e_{gi}(t+1) - O_g\|^2 + \|e_{gj}(t+1) - O_g\|^2 \\ & - \quad \|e_{gi}(t) - O_g\|^2 - \|e_{gj}(t) - O_g\|^2 \end{aligned}$$

Since the global reference frame is arbitrary we chose $O_g = \mathbf{0}$ for sake of clarity. Thus

$$\begin{aligned} & V(t+1) - V(t) = \\ & = \|e_{gi}(t+1)\|^2 + \|e_{gj}(t+1)\|^2 - \|e_{gi}(t)\|^2 - \|e_{gj}(t)\|^2 \end{aligned}$$

Now from the description given in Algorithm 1, we know that

$$e_i(t+1) = e_i(t) + \Delta_{ij}(t) \cdot c_{ij},$$

$$e_j(t+1) = e_j(t) + \Delta_{ji}(t) \cdot c_{ji}.$$

where:

$$\Delta_{ij}(t) = \left(\frac{d_{ij} + e_j(t)^T c_{ji} - e_i(t)^T c_{ij}}{2} + e_i(t)^T c_{ij}\right),$$

$$\Delta_{ji}(t) = \left(\frac{d_{ji} + e_i(t)^T c_{ij} - e_j(t)^T c_{ji}}{2} + e_j(t)^T c_{ji}\right)$$

and noticing that $\Delta_{ij}(t) = \Delta_{ji}(t) = \Delta(t)$, and by definition $c_{ij} = -c_{ji} = \hat{c}$ with $\hat{c}^T\hat{c} = 1$, the previous equation can be re-written as

$$\begin{aligned} V(t+1) - V(t) &= \|e_{gi} + \Delta\hat{c}\|^2 + \|e_{gj} - \Delta\hat{c}\|^2 \\ & \quad -\|e_{gi}\|^2 - \|e_{gj}\|^2 \\ &= e_{gi}^T e_{gi} + 2\Delta\hat{c}^T e_{gi} + \Delta^2 + e_{gj}^T e_{gj} \\ & \quad -2\Delta\hat{c}^T e_{gj} + \Delta^2 - e_{gi}^T e_{gi} - e_{gj}^T e_{gj} \\ &= 2\Delta^2 + 2\Delta\hat{c}^T(e_{gi} - e_{gj}) \end{aligned}$$

where the temporal index has been omitted for sake of clarity.

Now, by observing the Fig. 1 can be noticed that Δ is nothing more than the projection over $\hat{c}$ at time t of the vector $e_{gj} - e_{gi}$ scaled by a factor of two,

$$\Delta = \|e_{gj} - e_{gi}\| \frac{\cos(\alpha)}{2}$$

where α is the angle between e_{gj} and e_{gi}. Now, by substituting for Δ:

$$\begin{aligned} & V(t+1) - V(t) = \\ & = \quad 2\Delta^2 + 2\Delta\hat{c}^T(e_{gi} - e_{gj}) \\ & = \quad \frac{\|e_{gj} - e_{gi}\|^2 \cos(\alpha)^2}{2} - \|e_{gj} - e_{gi}\|^2 \cos(\alpha)^2 \leq 0 \end{aligned}$$

where

$$\begin{aligned} \hat{c}^T(e_{gi} - e_{gj}) &= -\hat{c}^T(e_{gj} - e_{gi}) \\ &= -\|e_{gi} - e_{gj}\| \cos(\alpha). \end{aligned}$$

Thus proving that $V(t)$ is a non-increasing function of time. $V(t+1) = V(t)$ each time an edge connecting two nodes who have exactly the same estimate of the network centroid are chosen.

If $V(t) \neq 0$ necessarily there exist at least two nodes in the network such that $e_{gi} \neq e_{gj} \neq O_g$. Since we assume that the network is fully connected, i.e. each node has the possibility to communicate with any other node, and given that each communication link is assumed to have a strictly positive probability to be activated at each instant of time, then we have that the probability

$$Pr\left(\lim_{t\to\infty} V(t'+t) - V(t') < 0\right) = 1$$

since we are sampling a finite set of elements an infinite amounts of times.

Since for $V(0)$ we have that

$$\forall i, e_{gi} = \frac{\sum_{i=1}^{n} p_i}{n},$$

then

$$\forall i \in V,\ Pr\left(\lim_{t\to\infty} e_{gi}(t) = \frac{\sum_{i=1}^{n} p_i}{n}\right) = 1,$$

thus proving the statement. ■

IV. Simulations

In order to corroborate the mathematical analysis, several simulations have been performed by exploiting a framework developed by the authors using Matlab. Different network topologies such as fully connected, arbitrary or tetrahedrons-based have been investigated. The following edge selection process has been exploited for the simulations: at each time step, an agent is selected randomly using a uniform probability distribution, it then communicates with all its neighbors in random order. Such a selection process has been chosen to mimic the expected behavior of the algorithm in a real application where communications are sequential and asynchronous.

A. Fully connected Graph

In the case of a fully connected graph the algorithms always converges, as proven in Theorem 1. In Fig. 2, a configuration where a network is deployed according to the vertexes of a cube (8 agents) is shown. In particular, each node converges to the right estimate of the network centroid after only few iterations. Note that, this particular embedding was chosen only for sake of visualization and it does not affect the convergence of the algorithm.

As far as the convergence time is concerned, several simulations considering a varying number of vertexes ranging from 10 nodes to 100 nodes have been performed. Results are given in Table I. In detail, 10 different configurations were considered, each one was run 50 times and at each single iteration a deployment was randomly generated. Table I shows the average number of iteration required to the algorithm in order to converge for each configuration. Although the number of iterations increases with the number of vertexes, it is important to recall the inherent parallelism of the algorithm previously discussed which is not revealed by this table. Indeed, at each iteration in a real context there might be several couples of nodes performing the algorithm at the same time, while the code running in Matlab is sequential.

TABLE I

Convergence Rate - Fully Connected Graph

Number of Vertixes	Number of Iterations
10	57
20	65
30	77
40	82
50	90
60	98
70	106
80	116
90	127
100	136

B. Tetrahedrons-based Graphs

In this section, a conjecture that has been validated only by simulation so far is proposed. It involves a special class of connectivity graphs, namely graphs obtained by opportunely interconnecting tetrahedrons. An inductive argument is given to convince the reader of the correctness of the proposed conjecture.

Conjecture *2:* A sufficient condition for Algorithm 1 to converge is that the network of agents is represented by a graph composed by tetrahedrons sharing one face.

This conjecture can be supported by a constructive argument. As shown in Fig. 4, let us build a new connectivity graph by "attaching" two tetrahedrons along a face and then several others in at different spots. The algorithm can then be thought as acting locally on each single tetrahedron and finally composing the result along the segment connecting the centroids of the two tetrahedrons. Now if a connectivity graph characterized by n tetrahedrons is considered, this process works if the algorithm is assumed to be executed at each step only in a couple of adjacent tetrahedrons. However, as the algorithm convergence is not affected by the edge selection order, this argumentation holds for any given edge selection sequence.

C. Arbitrary Connected Graphs

A general necessary condition for the convergence of Algorithm 1 for arbitrary connected graphs is now given.

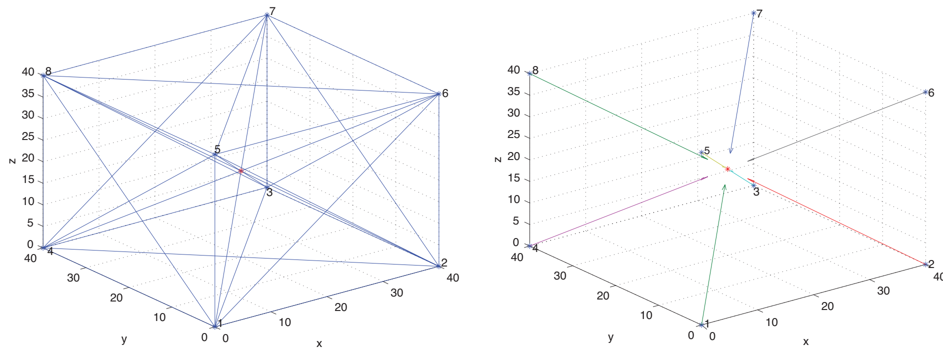

Fig. 2. Fully connected graph with 8 nodes.

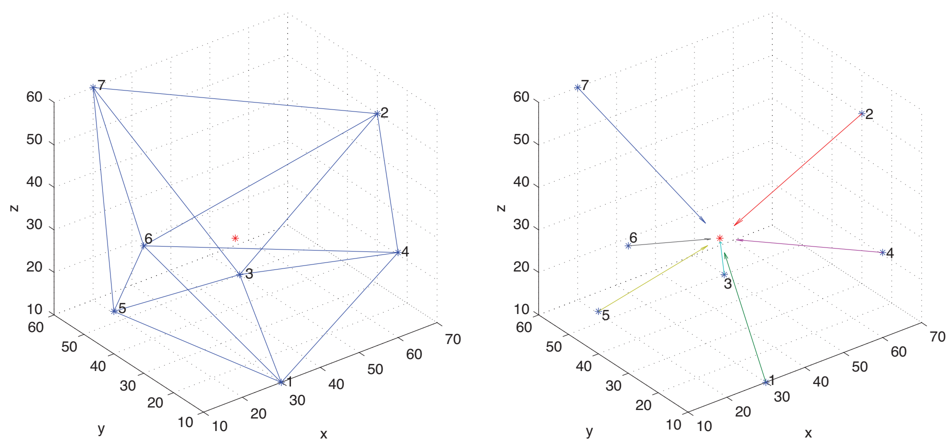

Fig. 3. Arbitrary graph with 7 nodes.

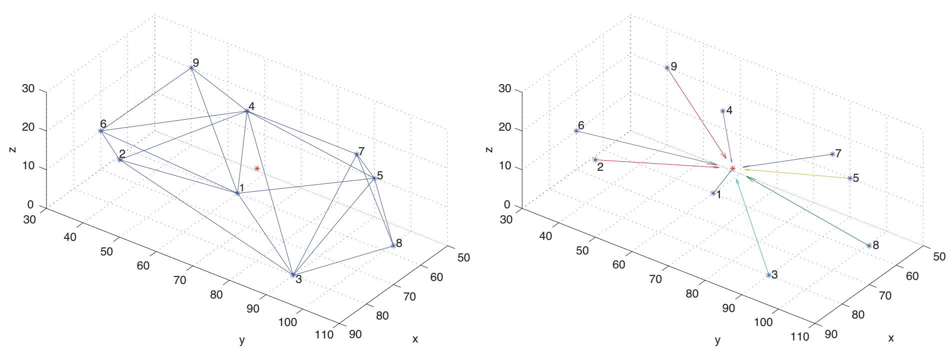

Fig. 4. Tetrahedron-based graph with 9 nodes.

***Theorem** 3:* If a network of agents with $p_i \in \mathbb{R}^d$ executes Algorithm 1, a necessary condition for the agents to have the same common estimate of the network centroid is that each agent has at least d neighbors.

Proof: The agents perform estimation updates along the line of sight between them, at each iteration they can adjust their estimate along only one direction. Thus to be able to adjust their estimate in a $\mathbb{R}^d$ space they need at least d independent directions over which perform their update. This condition is not sufficient, a counter-example in Fig.6 is provided. ■

Although the algorithm has proven to perform well if the graph is sufficiently connected, convergence cannot be achieved for any arbitrary connected graph.

A reason why this happens is that the algorithm involves a projection of the agents' estimate along the line of sight with their neighbors, this may fail to propagate enough information about the agent's estimate if the graph is not sufficiently connected.

V. Conclusions

In this paper, a novel algorithm to deal with the problem of estimating the network centroid in a multi agent system

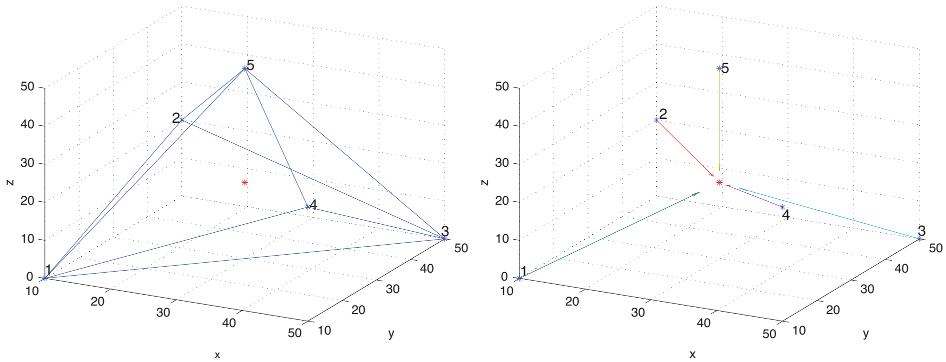

Fig. 5. Arbitrary graph with 5 nodes.

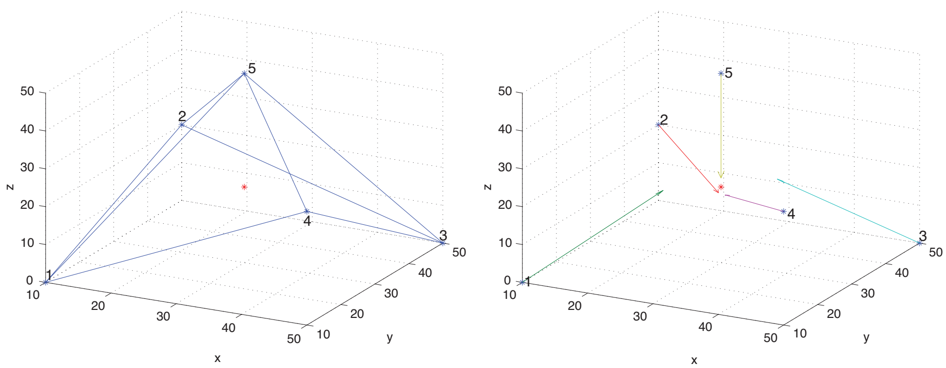

Fig. 6. Arbitrary graph with 5 nodes.

has been provided. In this framework, agents are assumed to be lacking any global reference frame or absolute position information. Collaboration among agents, which involves the computation of the relative distance and the direction of their line of sight with respect to the local reference frame of each agent, was limited to the exchange of the projection of the actual estimates along the direction of the line of sight.

The proposed algorithm can be thought as a general tool to retrieve information about the centroid of a network of agents. Indeed, this information turns out to be crucial for a large family of algorithms dealing with decentralized motion control, as many simplifying assumption can be reduced.

A proof of convergence for a fully connected network, i.e., each agent correctly estimates the location of the network centroid with respect to its own local reference frame, is given. A formal characterization of all the graph structures over which the proposed algorithm converges is left for future research.

References

[1] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *Automatic Control, IEEE Transactions on*, vol. 48, no. 6, pp. 988–1001, June 2003.

[2] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Automat. Contr.*, vol. 51, no. 3, pp. 401 – 420, 2006.

[3] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[4] H. G. Tanner and A. Jadbabaie, "Stable flocking of mobile agents, part ii: Dynamic topology," in *In IEEE Conference on Decision and Control*, 2003, pp. 2016–2021.

[5] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1520 – 1533, 2004.

[6] J.Cortes, S.Martinez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in d dimensions," *IEEE Trans. Robot. Automat.*, vol. 51, no. 8, pp. 1289 – 1298, 2006.

[7] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[8] G. Ferrari-Trecate, M. Egerstedt, A. Buffa, and M. Ji, "Laplacian sheep: A hybrid, stop-go policy for leader-based containment control." in *HSCC*, vol. 3927, 2006, pp. 212–226.

[9] D. Dimarogonas and K. Kyriakopoulos, "On the rendezvous problem for multiple nonholonomic agents," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 916–922, 2007.

[10] J. Rodrigues, D. Figueira, C. Neves, and M. I. Ribeiro, "Leader-following graph-based distributed formation control," in *Proc. of Robotica 2008 - 8th Conference on Autonomous Robot Systems and Competitions*, 2008.

[11] D. Dimarogonas, P. Tsiotras, and K. Kyriakopoulos, "Laplacian cooperative attitude control of multiple rigid bodies," *Intelligent Control, 2006. IEEE International Symposium on*, pp. 3064–3069, Oct. 2006.

[12] M. Ji and M. Egerstedt, "Distributed coordination control of multi-agent systems while preserving connectedness," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693–703, August 2007.

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

[14] W. Ren, R. Beard, and E. Atkins, "A survey of consensus problems in multi-agent coordination," *American Control Conference*, vol. 3, pp. 1859–1864, 2005.

[15] C. Godsil and G. Royle, *Algebraic graph theory*. Springer, 2001.