2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Distributed Attitude Synchronization Control

Johan Thunberg, Eduardo Montijano, Xiaoming Hu

Abstract— **In this paper we consider the problem of constructing feedback control laws for a system of n agents that shall synchronize their attitudes in $SO(3)$. We propose distributed controllers for two synchronization problems, in which the objective is the same, to synchronize the orientations, but what the agents can perceive or communicate differs. In the first problem the agents can measure their orientation to a common reference object, and either communicate with the neighbors or estimate the relative orientation to their neighbors. In the second problem the agents can, without communication, only measure the relative orientation to the neighbors. For the first problem we present a controller which will lead to synchronization, provided the neighborhood graph is connected. For the second problem we present a controller that will lead to synchronization provided the neighborhood graph is connected and the agents initially are contained within a geodesic ball of radius $\frac{\pi}{2}$, which is the maximal convex set in $SO(3)$.**

I. INTRODUCTION

There has been an extensive study of cooperative control problems in the past few years in applications such as manipulation, surveillance or exploration of an environment. Solutions based on distributed consensus, *e.g.*, [2], [13] and the references therein, present a lot of properties of great interest for these kind of problems. They do not rely in any specific communication topology, scale well with the number of agents and are robust to the exit or the introduction of new agents to the group.

One of the most challenging problems using distributed consensus is attitude synchronization. The attitude synchronization problem corresponds to the non-linear consensus problem on the manifold $SO(3)$ of all orthogonal matrices in $\mathbb{R}^{3\times 3}$ with determinant equal to 1. Applications of this scenario include flocking or the coordination of teams of UAVs and satellites.

There are several works in the literature that study this problem. Leader-follower schemes are adopted in [1] and [5]. However, with these approaches the interaction topology between the agents is limited to a spanning tree or a ring. The orientation alignment in terms of the velocity of all the agents, commonly referred to as flocking, is studied in [10], [11]. Passivity-based controllers are used in [6], [12]. These approaches either assume the observation of the velocities of the agents [6] or the knowledge of a global frame [12]. Using relative information, in [14] chordal distances are used. In order to compute this distance, based on the arithmetic mean of the rotation matrices, the agents need to communicate their relative orientations.

A problem that is similar to the attitude synchronization can be found in the context of camera networks calibration [15] and pose averaging [16]. Given initial relative orientations between the cameras, the goal is to construct an iterative algorithm that converges to an average orientation, *i.e.*, the Riemannian centre of mass, or the Karcher mean [8]. However, these approaches rely on discrete-time updates and optimization techniques and therefore, they are not suitable for the construction of continuous time distributed controllers for a group of agents leading to synchronization of their attitudes.

In order to solve the attitude synchronization problem we present distributed angular velocity controllers, making use of the axis angle representation to describe the orientation of the agents. Using this representation, which is novel in this context, the analysis of the synchronization problem becomes tractable due to the properties of the the system. In the paper we propose solutions for two different synchronization problems. The objective in the two problems is the same, to synchronize the orientations, however what the agents can perceive or estimate differs between the two problems.

In the first problem each agent can only measure its orientation to a common reference object. We consider two cases for the interaction between neighboring agents. In the first case they communicate with their neighbors and and in the second case, they can estimate the relative orientations between them. This is the case for example when, using a vision sensor, the agents can observe a static object which is known to them but they are not able to directly observe each other. By exchanging these measurements, the proposed controller will lead to synchronization, provided the neighborhood graph is connected.

The second problem is more challenging because the agents are only able to measure the relative orientation to its neighbors in their own body frame, and no communication between the agents are considered. This situation is more realistic, since the availability of a global reference will be difficult in many situations. We present a controller that will lead to synchronization, provided the interaction graph is connected and the agents are initially contained within a geodesic ball of radius $\pi/2$, the largest convex geodesic ball on $SO(3)$. Similar solutions have been presented in the literature for attitude synchronization, but to the the knowledge of the authors the axis-angle representation has not been used, and these works have often considered the localization or estimation problem rather than the control problem.

J. Thunberg and X. Hu are with Optimization and Systems Theory, Royal Institute of Technology (KTH). S-100 44 Stockholm, Sweden (e-mail: johan.thunberg@math.kth.se).

E. Montijano is with Universidad de Zaragoza, María de Luna, 50018 Zaragoza, Spain (e-mail: emonti@unizar.es).

This paper will proceed as follows. In Section II, first the dynamics of the agents is introduced, secondly the two different synchronization problems will be proposed. In Section III the solution to the problem is introduced, where we first propose the controller for the first problem and then propose the controller for the second problem. In section IV an illustrative example is provided where the two controllers are used for the same initial configuration for a set of five agents. Finally the paper is concluded in Section V.

II. Problem Formulation

We want to synchronize the attitude for a system of n rigid body agents. The formalism of [3], [4] will be adapted, where the axis-angle representation of the orientation is used. Let the world frame be denoted by $\mathcal{F}_w$ and the instantaneous body frame of agent i by $\mathcal{F}_i$. The axis-angle representation of the orientation at time t of agent i in the world frame $\mathcal{F}_w$ is denoted by $\theta_i(t)\boldsymbol{u}_i(t)$, where $\boldsymbol{u}_i(t)^T\boldsymbol{u}_i(t) = 1$. The axis-angle representation of the orientation of agent j in the frame of agent i, $\mathcal{F}_i$, is denoted as $\theta_{ij}(t)\boldsymbol{u}_{ij}(t)$. If $\boldsymbol{R}_i$ is the orientation of agent i in the the frame $\mathcal{F}_w$, represented as a rotation matrix, then $\theta_i(t)\hat{\boldsymbol{u}}_i(t) = \mathrm{Log}(\boldsymbol{R}_i)$, and $\theta_{ij}(t)\hat{\boldsymbol{u}}_{ij}(t) = \mathrm{Log}(\boldsymbol{R}_i^T\boldsymbol{R}_j)$ (see *e.g.* [9]), where $\hat{\mathbf{x}}$ denotes the skew symmetric matrix generated by $\mathbf{x}$.

We will for simplicity throughout the text, when possible, abbreviate $\theta_i(t)\boldsymbol{u}_i(t)$ and $\theta_{ij}(t)\boldsymbol{u}_{ij}(t)$ as $\boldsymbol{x}_i$ and $\boldsymbol{x}_{ij}$ respectively, and $\boldsymbol{x} = [\boldsymbol{x}_1^T, \boldsymbol{x}_2^T, ..., \boldsymbol{x}_n^T]^T$. Note that

$$\boldsymbol{x}_{ij} = -\boldsymbol{x}_{ji}, \tag{1}$$

and in general

$$\boldsymbol{x}_j - \boldsymbol{x}_i \neq \boldsymbol{x}_{ij}, \tag{2}$$

which can be easily seen from the Baker-Campbell-Hausdorff formula. Throughout the text, when we refer to the orientation of an agent it will be the axis-angle representation if nothing else is explicitly said.

The Riemannian distance between two points (this time represented as orthogonal matrices) in $SO(3)$ is defined as

$$d_{\boldsymbol{R}}(\boldsymbol{R}_i, \boldsymbol{R}_j) = \frac{1}{\sqrt{2}}||\mathrm{Log}(\boldsymbol{R}_i^T\boldsymbol{R}_j)||_F = |\theta_{ij}|,$$

where θ_{ij} is the angle of the axis-angle representation of the rotation $\boldsymbol{R}_i^T\boldsymbol{R}_j$. Note that from the definition of $\mathbf{x_i}$, it follows that

$$\boldsymbol{x}_i^T\boldsymbol{x}_i = \theta_i^2. \tag{3}$$

Denote the instantaneous angular velocity of $\mathcal{F}_i$ seen in the frame $\mathcal{F}_i$ as $\boldsymbol{\omega}_i$. The dynamics of $\boldsymbol{x}_i$ is given by

$$\dot{\boldsymbol{x}}_i = \boldsymbol{L}_{\boldsymbol{x}_i}\boldsymbol{\omega}_i, \tag{4}$$

and the dynamics of $\boldsymbol{x}_{ij}$ is given by

$$\dot{\boldsymbol{x}}_{ij} = \boldsymbol{L}_{\boldsymbol{x}_{ij}}\boldsymbol{\omega}_j - \boldsymbol{L}_{-\boldsymbol{x}_{ij}}\boldsymbol{\omega}_i, \tag{5}$$

where the transition matrix $\boldsymbol{L}_{\theta\boldsymbol{u}}$ was given in [7] as

$$\boldsymbol{L}_{\theta\boldsymbol{u}} = \boldsymbol{I}_3 - \frac{\theta}{2}\hat{\boldsymbol{u}} + \left(1 - \frac{\mathrm{sinc}(\theta)}{\mathrm{sinc}^2\frac{\theta}{2}}\right)\hat{\boldsymbol{u}}^2. \tag{6}$$

The function $\mathrm{sinc}(x)$ is defined so that $x\mathrm{sinc}(x) = \sin(x)$ and $\mathrm{sinc}(0) = 1$. It was also shown in [7], that $\boldsymbol{L}_{\theta\boldsymbol{u}}$ is invertible for $\theta \in (-2\pi, 2\pi)$.

The connectivity between the agents will be represented by the undirected connectivity (or neighborhood) graph G, which is assumed to be connected. Let the indices of the agents in the group constitute the vertexes in the graph. Agent i will be connected to agents with index in the set $\mathcal{N}_i$, *i.e.*, there will be an edge (i,j) in the graph. Since G is undirected $j \in \mathcal{N}_i$ if and only if $i \in \mathcal{N}_j$. We will also refer to $\mathcal{N}_i$ as the neighborhood of agent i. The connectivity will either be based on estimation or communication, and will be clarified in the problems below.

In this setting we want to construct a controller $\boldsymbol{\omega}_i$ for each agent i so that the orientations of all agents become synchronized as the time goes to infinity, *i.e.*,

$$\boldsymbol{x}_i - \boldsymbol{x}_j \to \boldsymbol{0}, \tag{7}$$

for all pairs (i, j), as $t \to \infty$. A controller solving this problem will be referred to as a *synchronization controller*. We now formulate the two synchronization problems.

Problem 1 - Synchronization using global information

Construct a *synchronization controller* using global information. Here by global information we mean that the measurements $\{\boldsymbol{x}_j - \boldsymbol{x}_i,\ j \in N_i\}$ is available for agent i and used in the controller (8). In practice this implies that agent i can can measure $\boldsymbol{x}_i$, which is the case if for example the agents are using a vision sensor and are observing some common static object which is known to the agents. This object will then serve as the global frame $\mathcal{F}_w$. The vector $\boldsymbol{x}_j$, where $j \in \mathcal{N}_i$, is obtained by one of the following two means.

1) Using communication. Here it is assumed that agent i transmits its measurement of $\boldsymbol{x}_i$ to agent j if they are neighbors.
2) without communication. Here it is assumed that agent i can estimate the relative orientation $\boldsymbol{x}_{ij}$ to agent j. Thus agent i can calculate the rotation $\boldsymbol{x}_j$ using the following relations

$$\begin{aligned}
\boldsymbol{R}_{ij} &= \exp(\hat{\boldsymbol{x}}_{ij}), \\
\boldsymbol{R}_i &= \exp(\hat{\boldsymbol{x}}_i), \\
\boldsymbol{R}_j &= \boldsymbol{R}_i\boldsymbol{R}_{ij}, \\
\hat{\boldsymbol{x}}_j &= \log(\boldsymbol{R}_{\boldsymbol{j}}).
\end{aligned}$$

The two methods complement each other. If it is not possible to measure the relative orientations between neighboring agents, communication can be used. On the other hand, in situations when communication is not possible the second alternative can be used.

One might argue that if a measurement of $\boldsymbol{x}_i$ is available for each agent i, then each agent can *e.g.* use the controller

$$\boldsymbol{\omega}_i = -\boldsymbol{x}_i,$$

in order to reach synchronization at the origin of $\mathcal{F}_w$. Thus a distributed controller is not necessary. However, there are reasons to prefer a distributed controller in which each agent i uses the information $\{\boldsymbol{x}_j - \boldsymbol{x}_i,\ j \in \mathcal{N}_i\}$.

We can guarantee a better synchronization during the transient phase when the rotations of the agents are converging to their finial synchronized orientation. The synchronization becomes more robust and less sensitive to measurement errors. If agent i is only using the measurement $\boldsymbol{x}_i$, then this could be regarded as an open-loop controller in terms of synchronization (not in terms of state feedback).

We require that $\theta_i \in (-\pi, \pi)$ for each pair of agents (i,j), where $j \in \mathcal{N}_i$, so that there is a bijective map from the axis angle representation ($so(3)$) to $SO(3)$.

Problem 2 - Synchronization using local information

Construct a *synchronization controller* using local information. By local information we mean the following. Assume that agent i can directly measure or observe the relative orientation $\boldsymbol{x}_{ij}$ for all agents j in the set $\mathcal{N}_i$. Agent i can not measure the relative orientation to agents not in $\mathcal{N}_i$. The relative orientation can *e.g.* be measured by using vision see [15], [5], [3], [4].

No communication is assumed between the agents and similar to Problem 1, we also require that $\theta_{ij} \in (-\pi, \pi)$ for each pair of agents (i,j), where $j \in \mathcal{N}_i$.

III. Solution

In this section we propose two distributed controllers that solve Problem 1 and Problem 2 respectively. The section contains two subsections. In Subsection III-A the controller that solves *Problem 1* is presented, and in Subsection III-B the controller that solves *Problem 2* is presented.

A. Solution to problem 1

The candidate controller is just the standard consensus protocol

$$\boldsymbol{\omega}_i = k \sum_{j \in \mathcal{N}_i} (\boldsymbol{x}_j - \boldsymbol{x}_i), \tag{8}$$

where $k > 0$. The constant k will be omitted in the following analysis without loss of generality.

Proposition 3.1: Controller (8) solves *Problem 1*.

Proof. We have as a requirement that $|\theta_i(t)| < \pi, \forall t \geq 0$, in order to keep the bijective map between the axis-angle representation and $SO(3)$. In order to show that the requirement is fulfilled by using the controller, we show that the open ball $\boldsymbol{B}_r(I)$ around identity in $\mathcal{F}_w$ with radius $r = \pi$ is invariant.

The distance from identity to each orientation $\boldsymbol{x}_i$ is $|\theta_i|$. Suppose $|\theta_i| \geq |\theta_j|\, \forall j = 1, 2, ..., n$, we want to see how θ_i^2 changes. We have that

$$\begin{aligned} \frac{(\dot{\theta_i^2})}{2} = \frac{(\dot{\boldsymbol{x}_i^T \boldsymbol{x}_i})}{2} \quad &= \qquad (9)\\ \boldsymbol{x}_i^T \boldsymbol{L}_{\boldsymbol{x}_i} \sum_{j \in \mathcal{N}_i} (\boldsymbol{x}_j - \boldsymbol{x}_i) \quad &\leq \qquad (10)\\ -n_i \theta_i^2 + \sum_{j \in \mathcal{N}_i} \theta_i \theta_j \quad &\leq \quad 0, \qquad (11) \end{aligned}$$

where $n_i = |\mathcal{N}_i|$ and where we have used the fact that $\boldsymbol{x}_i^T \boldsymbol{L}_{\boldsymbol{x}_i} = \boldsymbol{x}_i^T$. So the largest distance between the orientation of any agent and the identity does not increase, implying that the open ball will be invariant.

Now consider the Lyapunov function candidate

$$\gamma(t) = \sum_{i=1}^{n} \theta_i^2 = \sum_{i=1}^{n} \boldsymbol{x}_i^T \boldsymbol{x}_i. \tag{12}$$

We calculate the time derivative of γ as

$$\begin{aligned} \dot{\gamma} \quad &= \quad \sum_{i=1}^{n} \boldsymbol{x}_i^T \boldsymbol{L}_{\boldsymbol{x}_i} \sum_{j \in \mathcal{N}_i} (\boldsymbol{x}_j - \boldsymbol{x}_i) \qquad (13)\\ &= \quad \sum_{i=1}^{n} \boldsymbol{x}_i^T \sum_{j \in \mathcal{N}_i} (\boldsymbol{x}_j - \boldsymbol{x}_i) \qquad (14)\\ &= \quad -\boldsymbol{x}^T \boldsymbol{L} \boldsymbol{x}, \qquad (15) \end{aligned}$$

where $\boldsymbol{L}$ is the graph Laplacian of G (not to mix up with $\boldsymbol{L}_{\boldsymbol{x}_i}$ which is a Jacobian matrix). Provided G is connected, which is an assumption, $\dot{\gamma}$ is negative unless $\boldsymbol{x}_i = \boldsymbol{x}_j\ \forall i, j$, and by the invariance theorem this implies that the system will be synchronized.

Proposition 3.1 states that the orientations of the agents will be synchronized as $t \to \infty$. A natural question to ask now is to what orientation the agents will converge in the global frame $\mathcal{F}_w$. In the Euclidean case the convergence point would be the arithmetic mean of the initial positions, assuming a system of agents with single integrator dynamics. One could assume something similar in this case. However this can not be true in general since the mean is not always time invariant. We hope the following discussion can shed some light on this question.

Introduce the vector $\boldsymbol{\theta} = [\theta_1, ..., \theta_n]$. From the proof of Proposition 3.1 and especially equations (13) to (15), we see that $\boldsymbol{\theta}^T \boldsymbol{\theta}$ is decreasing as long as there exists a pair of agents (i,j) such that $\theta_i \neq \theta_j$ or there exists a pair of agents (i,j) such that $\boldsymbol{u}_i \neq \boldsymbol{u}_j$.

The dynamics of $\boldsymbol{x}_i$ can be seen as the sum of two orthogonal parts

$$\dot{\boldsymbol{x}}_i = \dot{\theta}_i \boldsymbol{u}_i + \theta_i \dot{\boldsymbol{u}}_i, \tag{16}$$

where the first part is the change of $\boldsymbol{x}_i$ in the direction of $\boldsymbol{u}_i$ and the second part is the change of $\boldsymbol{x}_i$ in the direction orthogonal to $\boldsymbol{u}_i$. In order to get the dynamics of θ we multiply (16) with $\boldsymbol{u}_i^T$ from the left, and use the fact that $\boldsymbol{u}_i^T \boldsymbol{u}_i = 1$ and $\boldsymbol{u}_i^T \dot{\boldsymbol{u}} = 0$. We arive at the following dynamical equation

$$\dot{\theta}_i = \boldsymbol{u}_i^T \sum_{j \in \mathcal{N}_i} (\boldsymbol{x}_j - \boldsymbol{x}_i) = \sum_{j \in \mathcal{N}_i} (\boldsymbol{u}_i^T \boldsymbol{u}_j \theta_j - \theta_i). \tag{17}$$

In order to get the dynamics of $\boldsymbol{u}_i$ we use the fact that $\boldsymbol{u}_i$ and $\dot{\boldsymbol{u}}_i$ are orthogonal and thus $-\hat{\boldsymbol{u}}^2 \dot{\boldsymbol{u}} = \dot{\boldsymbol{u}}$. Hence

$$\dot{\boldsymbol{u}}_i = -\frac{1}{\theta_i} \hat{\boldsymbol{u}}_i^2 \boldsymbol{x}_i = -\frac{1}{\theta_i} (1 - k(\theta_i)) \hat{\boldsymbol{u}}_i^2 \sum_{j \in \mathcal{N}_i} \boldsymbol{x}_j + \hat{\boldsymbol{u}}_i \sum_{j \in \mathcal{N}_i} \boldsymbol{x}_j, \tag{18}$$

where we have used the fact that $\hat{\boldsymbol{u}}^3 = -\hat{\boldsymbol{u}}$.

Consider the case of two agents where $\boldsymbol{u}_1 \neq \boldsymbol{u}_2$ initially. One can see that the motion of $\boldsymbol{u}_1$ consists of two orthogonal components: the first component draws $\boldsymbol{u}_1$ closer to $\boldsymbol{u}_2$ while the second component generates a motion on the normal direction of the plane formed by $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ as long as $\theta_2 \neq 0$. This indicates that $\boldsymbol{u}_1$ would reach synchronization "spirally". Simulations seem to indicate that θ^* is zero, see Figure 1.

B. Solution to problem 2

The candidate controller is

$$\boldsymbol{\omega}_i = k \sum_{j \in \mathcal{N}_i} \boldsymbol{x}_{ij}, \tag{19}$$

where $k > 0$.

Proposition 3.2: If there is a $\boldsymbol{Q} \in SO(3)$, such that the orientations of the agents initially are contained within an open ball $B_r(\boldsymbol{Q})$ of radius r less than or equal to $\frac{\pi}{2}$ centered around $\boldsymbol{Q}$ in the frame $\mathcal{F}_w$, *i.e.*,

$$|\theta_{ij}| < \frac{\pi}{2} \quad \forall i, j, \tag{20}$$

the controller (19) solves *Problem 2*.

Proof. We will divide the proof into two parts. In the first part we show invariance and in the second part we will show convergence.

Suppose that the condition

$$|\theta_{ij}| < \frac{\pi}{2} \quad \forall i, j,$$

is fulfilled, or that there is a $\boldsymbol{Q}$ in $\mathcal{F}_w$ such that all the orientations of the agents are contained in the open ball $B_r(\boldsymbol{Q})$, where $r \leq \frac{\pi}{2}$. The ball $B_r(\boldsymbol{Q})$ is a convex set, see [8], [9].

Let $\boldsymbol{X}(t) = \{\boldsymbol{x}_i(t)\}$, $i = 1, ..., n$ be the set of all rotations from each frame $\mathcal{F}_i$ to the frame $\mathcal{F}_w$. Let $\boldsymbol{X}_i(t) = \{\boldsymbol{x}_j(t)\}$, $j \in \mathcal{N}_i$ be set of the rotations from the frames $\mathcal{F}_i$ of each agent i to the frame $\mathcal{F}_w$. The following relations must hold

$$Conv(\boldsymbol{X}_i(t)) \subset Conv(\boldsymbol{X}(t)) \subset B_r(\boldsymbol{Q}),$$

where $Conv(\boldsymbol{X}_i(t))$ and $Conv(\boldsymbol{X}(t))$ denotes the convex hull of the elements in $\boldsymbol{X}_i(t)$ and $\boldsymbol{X}(t)$ respectively. Let $\bar{\boldsymbol{X}}(t)$ denote the set of all rotations of the neighbors of agent i except agent i, *i.e.*, $\bar{\boldsymbol{X}}(t) = \boldsymbol{X}_i(t) - \{\boldsymbol{x}_i\}$.

The Karcher mean of the elements in the set $\bar{\boldsymbol{X}}_i(t)$ at time t is defined as the rotation $\boldsymbol{R}_i^*(t)$ which minimizes

$$\gamma(\boldsymbol{R}_i^*) = \sum_{j \in \mathcal{N}_i} d^2_{\boldsymbol{R}}(\boldsymbol{R}_j(t), \boldsymbol{R}_i^*(t)).$$

The rotation matrix we define as $\boldsymbol{R}_i = \exp(\hat{\boldsymbol{x}}_i)$ and

$$\hat{\boldsymbol{x}}_{ij} = \mathrm{Log}(\boldsymbol{R}_i^T \boldsymbol{R}_j).$$

The dynamics of $\boldsymbol{R}_i$ is thus given by

$$\dot{\boldsymbol{R}}_i = \boldsymbol{R}_i \hat{\boldsymbol{\omega}}_i = \boldsymbol{R}_i \sum_{j \in \mathcal{N}_i} \hat{\boldsymbol{x}}_{ij}. \tag{21}$$

The Karcher mean $\boldsymbol{R}_i^*(t) \in Conv(\boldsymbol{X}_i(t))$. The function $\gamma(\boldsymbol{R}_i^*)$ is convex, see [9], and is defined on the convex set $Conv(\boldsymbol{X}_i(t))$. Thus the shortest path between $\boldsymbol{R}_i$ and $\boldsymbol{R}_i^*$ is contained in $Conv(\boldsymbol{X}_i(t))$ and the direction to move in order to follow this path at the rotation $\boldsymbol{R}_i$, is given by the covariant derivative of $\gamma(\boldsymbol{R}_i)$, see [9],

$$\nabla \gamma(\boldsymbol{R}_i) = \boldsymbol{R}_i \sum_{j \in \mathcal{N}_i} \mathrm{Log}(\boldsymbol{R}_j^T \boldsymbol{R}_i) = -\boldsymbol{R}_i \sum_{j \in \mathcal{N}_i} \hat{\boldsymbol{x}}_{ij}. \tag{22}$$

Since (21) and (22) are opposite in sign, this implies that $\boldsymbol{R}_i(t)$ will move in the gradient descent direction of $\gamma(t)$ at time t, a direction that points into $Conv(\boldsymbol{X}_i(t))$. Since this is true for all agents, especially the ones on the border of $Conv(\boldsymbol{X}(t))$, the set $B_{\boldsymbol{r}}(\boldsymbol{Q})$ is invariant.

Now we will prove that the rotations will be synchronized as $t \to \infty$. This can be easily shown using (6). We first define the sum of all the rotations of the neighbors of agent i as

$$\boldsymbol{x}_{s_i} = \sum_{j \in \mathcal{N}_i} \boldsymbol{x}_{ij}, \tag{23}$$

and define the candidate Lyapunov function

$$\phi(t) = \frac{1}{2} \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \boldsymbol{x}_{ij}^T \boldsymbol{x}_{ij} = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} d^2_{\boldsymbol{R}}(\boldsymbol{R}_i, \boldsymbol{R}_j). \tag{24}$$

Since the graph is connected, ϕ will be zero if and only if the orientations are aligned. The derivative is given by

$$\begin{aligned}
\dot{\phi}(t) &= \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \boldsymbol{x}_{ij}^T (\boldsymbol{L}_{x_{ij}} \boldsymbol{\omega}_j - \boldsymbol{L}_{-x_{ij}} \boldsymbol{\omega}_i) \\
&= \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \sum_{k \in \mathcal{N}_j} \boldsymbol{x}_{ij}^T \boldsymbol{x}_{jk} - \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \sum_{k \in \mathcal{N}_i} \boldsymbol{x}_{ij}^T \boldsymbol{x}_{ik} \\
&= \sum_{j=1}^{n} \sum_{i \in \mathcal{N}_j} \sum_{k \in \mathcal{N}_j} \boldsymbol{x}_{ij}^T \boldsymbol{x}_{jk} - \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \sum_{k \in \mathcal{N}_i} \boldsymbol{x}_{ij}^T \boldsymbol{x}_{ik} \\
&= -\sum_{j=1}^{n} \sum_{i \in \mathcal{N}_j} \sum_{k \in \mathcal{N}_j} \boldsymbol{x}_{ji}^T \boldsymbol{x}_{jk} - \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \sum_{k \in \mathcal{N}_i} \boldsymbol{x}_{ij}^T \boldsymbol{x}_{ik} \\
&= -2 \sum_{i=1}^{n} \boldsymbol{x}_{s_i}^T \boldsymbol{x}_{s_i},
\end{aligned}$$

This implies that $\dot{\phi}(t)$ is negative unless $\boldsymbol{x}_{s_i} = 0$, $i = 1, ..., n$. Thus $\boldsymbol{x}_{ij}(t)$ will converge to the largest invariant set contained in $\boldsymbol{x}_{s_i} = 0$, $i = 1, ..., n$. Since

$$\nabla \gamma(\boldsymbol{R}_i) = \boldsymbol{R}_i \boldsymbol{x}_{s_i}, \; i = 1, ..., n,$$

$\boldsymbol{x}_{s_i} = 0$ implies each $\boldsymbol{R}_i$ is in the Karcher mean of all its neighbors. If the rotations in the neighborhood are not synchronized at this point, then there must be some rotations forming the boundary of the convex hull $Conv(\boldsymbol{X}_i)$. Provided G is connected, for any of those rotations the rotation would not be in the Karcher mean of its neighbors, which is a contradiction. This implies that if the agents are in the largest invariant set of $\boldsymbol{x}_{s_i} = 0$, $i = 1, ..., n.$, the rotations have to be synchronized.

IV. Illustrative Example

We will now show the performance of the controllers for an illustrative example with 5 agents, whose initial orientations are uniformly distributed in the geodesic ball $B_r(\boldsymbol{Q})$ where $r = \frac{\pi}{2}$. The simulation was conducted in Matlab Simulink. The graph adjaceny matrix E in G is given by

$$E = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix},$$

the time horizon was chosen as 10 sec and $k = 5$. In Figure 1 the controller (8) is used and one can see how the agents converge from their initial orientations (blue squares) to the final orientation where they reach synchronization. The synchronization point (or orientation) is at the origin of of the global frame $\mathcal{F}_w$ in this case.

In Figure 2 the controller (19) is used. one can see how the agents converge from their initial orientations (blue squares) the the final orientation where they reach synchronization.

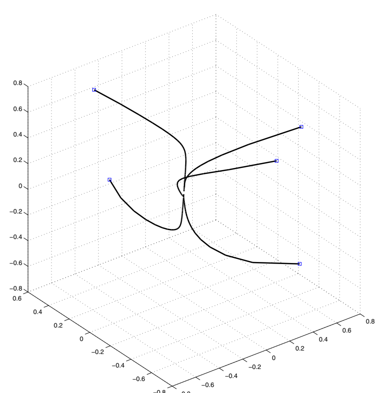

Fig. 1. The rotations of five agents converge to a synchronized rotation as time goes to infinity and the agents use controller (8). The initial orientations are marked by blue squares. In this example the constant k was chosen as 5, and the time horizon was 10 sec. The agents are synchronized at the rotation $[0, 0, 0]^T$ or at the identity of $\mathcal{F}_w$, which differs from the synchronization point in Figure 2.

V. Conclusions

In this paper we have proposed two distributed controllers for the attitude synchronization problem. The first controller is based on differences between the orientations in a global frame, and the second controller is based on the relative orientations between the agents. The axis-angle representation has been used in order to represent the orientations, and its properties makes the convergence analysis easy. The first controller leads to synchronization, provided the neighborhood graph is connected. When the second controller is used, the orientations will be synchronized, provided the interaction graph is connected and the agents initially are contained within a geodesic ball of radius $\frac{\pi}{2}$, which is the maximal convex set in $SO(3)$.

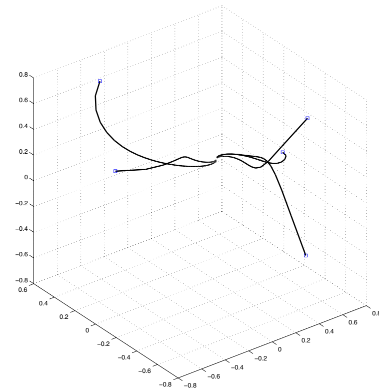

Fig. 2. The rotations of five agents converge to a synchronized rotation as time goes to infinity and the agents use controller (19). In this example the constant k was chosen as 5, and the time horizon was 10 sec. Blue squares indicate the initial orientations of the agents.

Acknowledgments

The first and the third author gratefully acknowledge the financial support from the Swedish National Space Technology Research Programme (NRFP) and the Swedish Research Council (VR). The second author is supported by the project DPI2009-08126 Ministerio de Ciencia e Innovacin/Union Europea and grant AP2007-03282 Ministerio de Educacion y Ciencia.

References

[1] A.K. Bondhus, K.Y. Pettersen, and J.T. Gravdahl. Leader/follower synchronization of satellite attitude without angular velocity measurements. In *44th IEEE Conference on Decision and Control*, pages 7270–7277, 2006.

[2] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009. Electronically available at http://coordinationbook.info.

[3] F. Chaumette and S. Hutchinson. Visual servo control. I. Basic approaches. *IEEE Robotics & Automation Magazine*, 13(4):82–90, 2006.

[4] F. Chaumette and S. Hutchinson. Visual servo control. II. Advanced approaches [Tutorial]. *IEEE Robotics & Automation Magazine*, 14(1):109–118, 2007.

[5] T. Ibuki, T. Hatanaka, M. Fujita, and M.W. Spong. Visual Feedback Attitude Synchronization in Leader-follower Type Visibility Structures. In *49th IEEE Conference on Decision and Control*, 2010.

[6] J.R. Lawton and R.W. Beard. Synchronized multiple spacecraft rotations. *Automatica*, 38(8):1359–1364, 2002.

[7] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 D visual servoing. *IEEE Transactions on Robotics and Automation*, 15(2):238–250, 1999.

[8] J.H. Manton. A globally convergent numerical algorithm for computing the centre of mass on compact Lie groups. In *IEEE int. Conf. on Control, Automation, Robotics and Vision*, pages 2211–2216, 2005.

[9] M. Moakher. Means and averaging in the group of rotations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):1–16, 2003.

[10] N. Moshtagh and A. Jadbabaie. Distributed geodesic control laws for flocking of nonholonomic agents. *IEEE Transaction on Automatic Control*, 52(4):681–686, 2007.

[11] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

[12] W. Ren. Distributed Cooperative Attitude Synchronization and Tracking for Multiple Rigid Bodies. *IEEE Transactions on Control Systems Technology*, 18(2):383–392, 2010.

[13] W. Ren and R. W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Communications and Control Engineering. Springer-Verlag, London, 2008.

[14] A. Sarlette, R. Sepulchre, and N.E. Leonard. Autonomous rigid body attitude synchronization. *Automatica*, 45(2):572–577, 2009.

[15] R. Tron and R. Vidal. Distributed 3-D localization in camera networks. In *Conference on Decision and Control*. Citeseer, 2009.

[16] R. Tron, R. Vidal, and A. Terzis. Distributed Pose Averaging in Camera Sensor Networks via Consensus on SE(3). In *International Conference on Distributed Smart Cameras*, pages 1–10, 2008.