

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



---

# Consensus-Based Attitude Maneuver of Multi-spacecraft with Exclusion Constraints

---

Innocent Okoloko

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.71506>

---

## Abstract

Some space missions involve cooperative multi-vehicle teams, for such purposes as interferometry and optimal sensor coverage, for example, NASA Terrestrial Planet Finder Mission. Cooperative navigation introduces extra constraints of exclusion zones between the spacecraft to protect them from damaging each other. This is in addition to external exclusion constraints introduced by damaging or blinding celestial objects. This work presents a quaternion-based attitude consensus protocol, using the communication topology of the team of spacecraft. The resulting distributed Laplacians of their communication graph are applied by semidefinite programming (SDP), to synthesize a series of time-varying optimal stochastic matrices. The matrices are used to generate various cooperative attitude maneuvers from the initial attitudes of the spacecraft. Exclusion constraints are satisfied by quaternion-based quadratically constrained attitude control (Q-CAC), where both static and dynamic exclusion zones are identified every time step, expressed as time-varying linear matrix inequalities (LMI) and solved by semidefinite programming.

**Keywords:** attitude manoeuvre, consensus, exclusion, optimization, LMI

---

## 1. Introduction

Some current space missions already demanded the deployment of teams of spacecraft which cooperate synergistically for such purposes as interferometry and sensor coverage [1, 2]; and many future missions will. Activities such as interferometry and sensor coverage require cooperative attitude control (AC)—the process of making a team of spacecraft, for example, satellites to point toward a specific direction of interest. This makes attitude control an essential part of space missions [3]. Apart from spacecraft, AC is also important in the navigation of aircraft and robots; therefore, it has been studied extensively in the literature, for example [4–11].

Basically, AC is a challenging problem, which becomes more challenging when multiple spacecraft are involved, in highly dynamic environments, and subject to external constraints such as blinding celestial objects such as the sun or some bright stars, which can damage onboard sensitive instruments. In addition, because of the close packing of spacecraft in a team, each of which has protruding appendages (e.g. thrusters and antennae), they must be careful with each other when changing attitude, in order to avoid collision with each other. When there is such a team of networked spacecraft which can communicate, then consensus theory based on graph Laplacians can be applied to achieve cooperation among them [12, 13].

The most common method of representing spacecraft attitude dynamics is by *unit quaternions*, mainly because quaternions do not encounter the singularities associated with other representations such as Euler angles and the *Modified Rodrigues Parameters* (MRP). However, the non-linearity of quaternion dynamics makes it difficult to apply Laplacian-like dynamics directly to quaternions.

We shall now consider some previous work on constrained attitude control (CAC). A brief survey of the main method attitude representation is in [4]. Ref. [5] considers quadratically constrained attitude control (Q-CAC), where the exclusion problems are formulated as a quadratic optimization problem and solved using linear matrix inequalities (LMIs) and semidefinite programming (SDP). It was solved for a single-spacecraft single obstacle in [5] and for two spacecraft in [6]. In [7] an attempt was made to extend [5, 6] to more than two spacecraft and obstacles. In [7–10], was extended to multiple spacecraft multiple obstacles in *different coordinate frames* (as the case of real spacecraft will be). An attempt was made in [11] to reduce the control torques required for effective attitude stabilization from three to two. This is applicable to underactuated spacecraft. [12] applies a consensus-based approach to distributed attitude alignment of a team of communicating spacecraft flying in formation, while [14] applies a Laplacian-based protocol to *leader-follower* attitude control of a team of spacecraft using the modified Rodriguez parameters.

Among the plethora of AC algorithms, only our works [7–10] apply consensus theory directly to quaternions, and only [5–10] tackle the problem of avoidance constraints. In addition, among the works [5–10] only [8–10] were developed for spacecraft in *different coordinate frames*, which has direct practical implementation. The contributions of this chapter are therefore aspects of our previous works [7–10], which include the following: (i) the development of a quaternion consensus protocol, (ii) incorporating dynamic avoidance constraints into the consensus framework using Q-CAC, (iii) mathematical convergence analysis for the quaternion-based consensus framework and (iv) solving the problem for the realistic scenario of multiple spacecraft in different coordinate frames, thus making it more suitable for practical implementation.

Note: the words *obstacle*, *avoidance*, *exclusion* and *exclusion vector* may be used interchangeably in this chapter. **Table 1** lists frequently used notation in this chapter.

## 2. Problem statement

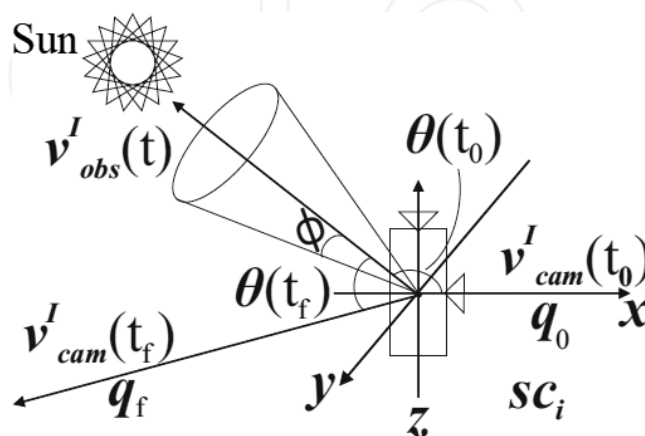
The problem of multi-spacecraft attitude control with avoidance constraints can be stated as follows. Given the initial positions  $x^i(t_0) \in \mathbb{R}^3$   $i=1 \cdots n$ , initial attitudes represented by quaternions

$q^i(t_0)$ , of a set of communicating spacecraft  $SC_i$ , generate a sequence of attitude consensus trajectories that drive the team to a consensus attitude  $q(t_f)$  while satisfying avoidance and norm constraints.

The problem stated above consists of two parts: *consensus* and *avoidance*. For the consensus problem, it is desired to drive the attitudes of all  $SC_i$  to a collective consensus attitude or to various *formation attitudes*. Consensus attitude means that each  $SC_i$  should eventually point to the same direction, which is the average of the initial quaternions. Formation attitudes means  $SC_i$  should finally point to various patterns, for example, each spacecraft can point at  $5^\circ$  away from each other about the z-axis. This we developed by introducing *relative offset quaternions* in the consensus framework. The second problem, *avoidance* constraints, is also important, because  $SC_i$  usually have appendages, for example, some  $SC_i$  have *thrusters* that emit hot plumes (plume impingement), and some have instruments that can be damaged by blinding celestial objects or by the appendage of another team member.

However, the ordinary consensus protocol violates the non-linearity of quaternion kinematics and the quaternion norm preserving requirement and therefore cannot be applied directly with quaternion dynamics. Also, the protocol ordinarily does not solve the problem of collision avoidance in *adversarial* situations. Thus, this chapter consists of aspects of our previous works [7–10], where we developed a consensus theory of quaternions, augmented with Q-CAC-based collision avoidance mechanisms. We employed an optimization approach and cast the problems as a *semidefinite program* (SDP), augmented with some convex quadratic constraints (avoidance), written as *linear matrix inequalities* (LMI). The quaternion consensus protocol computes consensus attitude trajectories each time step, and the Q-CAC avoidance procedure decides which of the computed trajectories are safe to follow or not. Unsafe trajectories are discarded, and a new set of quaternion vectors that avoid collision is generated. The cycle repeats until consensus is achieved.

To understand the avoidance (exclusion) problem, let us illustrate with a simpler single- $SC_i$  single-obstacle scenario as shown in **Figure 1**. In the figure, the  $SC_i$  must avoid (exclude) the Sun while rotating a photosensitive instrument from  $q_0$  to  $q_f$ .



**Figure 1.** Constrained attitude control problem for a single-sc single-exclusion scenario.  $SC_i$  must avoid (exclude) the Sun while rotating a photosensitive instrument from  $q_0$  to  $q_f$ .

Let  $v_{cam_i}^I(t)$  denote the unit camera vector in  $\mathcal{F}_{SC_i}^I$  corresponding to the  $SC_i$ 's attitude  $q^i$  (as defined in **Table 1**), and let  $v_{obs_i}^I(t)$  be the attitude quaternion of the obstacle to be avoided (in this case the Sun). Exclusion requires the time evolution of camera vector  $v_{cam_i}^I$  from  $v_{cam_i}^I(t_0)$  to  $v_{cam_i}^I(t_f)$  to avoid  $v_{obs_i}^I(t)$  all times with a minimum angular separation of  $\emptyset$ . The requirement is.

$$\theta(t) \geq \emptyset \quad (1)$$

or

$$\begin{aligned} v_{cam_i}^I(t)^T v_{obs_i}^I(t) &\leq \cos \emptyset, \\ \forall t \in [t_0, t_f] \end{aligned} \quad (2)$$

The constraint is a *non-convex quadratic* constraint; it was *convexified* in [4], which made it possible to be represented as a LMI using the quaternion attitude constraint formulation developed in [3] for a single-spacecraft single-obstacle scenario. In [4],  $v_{obs}^I$  was static, while  $v_{cam_i}^I(t)$  was evolving; both vectors were in the same coordinate frame. Although solving it in the same coordinate frame somewhat simplified the solution, it was not suitable for practical implementation because, in reality, the obstacle and spacecraft operate in different coordinate frames. Next, we present the basic mathematical preliminaries.

Notation	Meaning
$SC_i, SC_i$	Spacecraft $i$
$q^i$	Attitude quaternion vector of $SC_i$ , $q^i = [q_1 \ q_2 \ q_3 \ q_4]^T$
$q^{-i}$ or $q^{i*}$	Conjugate of $q^i$
$\bar{q}^i$	Vector part of $q^i$ , $\bar{q}^i = [q_1 \ q_2 \ q_3]^T$
$\bar{q}^{i \times}$	Antisymmetric of $q^i$
$q$	Stacked vector of more than one quaternion vectors
$q^{off}$	Stacked vector of more than one offset quaternion vectors
$\Omega, \Pi$	Quaternion dynamics plant matrix
$P$	Quaternion dynamics Laplacian-like plant matrix
$\omega$	Angular velocity
$\tau$	Control torque
$J$	Inertia matrix
$L$	Laplacian matrix
$P$	Laplacian-like stochastic matrix
$I_n$	Then $n \times n$ identity matrix
$S^m$	The set of $m \times m$ positive definite matrices

Notation	Meaning
$\tilde{A}$	Cone avoidance constraint matrix
$\mathcal{R}^i$	Rotation matrix corresponding to $q^i$
$\mathcal{F}_{SC_i}^I$	Fixed coordinate (Inertial) frame with origin at $SC_i$ 's center
$\mathcal{F}_{SC_i}^B$	Rotational coordinate (Body) frame with origin at $SC_i$ 's center
$v_{obs_i}^B$	Vector of obstacle in $\mathcal{F}_{SC_i}^B$
$v_{obs_i}^I$	Vector of obstacle in $\mathcal{F}_{SC_i}^I$
$v_{obs_i,j}^I$	Vector of the $j^{th}$ obstacle in $\mathcal{F}_{SC_i}^I$
$v_{cam_i}^B$	Vector of the $SC_i$ 's camera in $\mathcal{F}_{SC_i}^B$
$v_{cam_i}^I$	Vector of the $SC_i$ 's camera in $\mathcal{F}_{SC_i}^I$
$\otimes$	Kronecker multiplication operator
$\odot$	Quaternion multiplication operator
$\ominus$	Quaternion difference operator
$t_0$	Initial time
$t_f$	Final time
$x^i$	Position vector of $SC_i, SC_i$
$x$	Stacked vector of $n$ position vectors
$(x^{ij})^{off}$	Offset vector between $i$ and $j$
$x^{off}$	Stacked vector of $n$ offset vectors
$\mathcal{C}$	The consensus space for $\mathbf{q}, \mathcal{C} = \{\mathbf{q} q^1 = q^2 = \dots = q^n\}$

Table 1. Frequently used notations in this chapter.

3. Mathematical background

In this section, we shall briefly consider the two basic mathematical theories relevant to this chapter. More comprehensive study and analysis are in [10].

3.1. Quaternion-based rotational dynamics

Because quaternions are free from the problems of singularities inherent in Euler angles and most other ways of representing rotations, it is convenient to use unit quaternions to represent the attitude of a rigid body rotating in three-dimensional space (such as spacecraft or satellite) [15]. The quaternion is a four-element vector:

$$q = [q_1 \ q_2 \ q_3 | q_4]^T. \quad (3)$$

Here,  $[q_1 \ q_2 \ q_3]^T$  is the vector part, representing the axis of rotation in the Cartesian  $(x, y, z)$  coordinates, and  $q_4$  is a scalar part, representing the angle of rotation of the quaternion in degrees. The difference between two quaternions  $q^1$  and  $q^2$  can be represented in multiplication terms as.

$$\begin{aligned} q^d &= q^1 \odot q^{-2} = q^1 \odot [-q_1^2 - q_2^2 - q_3^2 - q_4^2]^T \\ &= Q^2 q^1, \end{aligned} \quad (4)$$

where  $q^{-2}$  is the *conjugate* of  $q^2$ . We used  $\odot$  here as a quaternion multiplication operator. And  $Q^2$  is defined as

$$Q^i = \begin{bmatrix} q_4^i & q_3^i & -q_2^i & -q_1^i \\ -q_3^i & q_4^i & q_1^i & -q_2^i \\ q_2^i & -q_1^i & q_4^i & -q_3^i \\ q_1^i & q_2^i & q_3^i & q_4^i \end{bmatrix} \quad (5)$$

Eq. (4) means that  $q^d$  is the rotation quaternion that originally transformed  $q^1$  to  $q^2$  or, alternatively,  $q^d$  is a rotation quaternion that can transform  $q^1$  to  $q^2$ .

The rotational dynamics for the  $i^{th}$  quaternion is.

$$\dot{q}^i = \frac{1}{2} \Omega^i \dot{q}^i = \frac{1}{2} \Pi^i \omega^i \quad (6)$$

where

$$\Omega^i = \begin{bmatrix} 0 & \omega_3^i & -\omega_2^i & \omega_1^i \\ -\omega_3^i & 0 & \omega_1^i & \omega_2^i \\ \omega_2^i & -\omega_1^i & 0 & \omega_3^i \\ -\omega_1^i & -\omega_2^i & -\omega_3^i & 0 \end{bmatrix} \quad (7)$$

$$\Pi^i = \begin{bmatrix} -q_4^i & q_3^i & -q_2^i \\ -q_3^i & -q_4^i & q_1^i \\ q_2^i & -q_1^i & -q_4^i \\ q_1^i & q_2^i & q_3^i \end{bmatrix} \quad (8)$$

are the *plant* matrices of quaternion dynamics.

Euler's first-order discretization of Eq. (6) yields

$$q^i(k+1) = I_4 q^i(k) + \frac{\Delta t}{2} \Omega^i(k) \dot{q}^i(k) = q^i(k) + \frac{\Delta t}{2} \Pi^i(k) \omega^i(k). \quad (9)$$



The dynamics of the rotational (angular) velocity  $\omega^i$  of  $q^i$  is

$$\begin{aligned} \begin{bmatrix} \dot{\omega}_1^i \\ \dot{\omega}_2^i \\ \dot{\omega}_3^i \end{bmatrix} &= \begin{bmatrix} ((J_2^i - J_3^i)\omega_2^i\omega_3^i + \tau_1^i)/J_1^i \\ ((J_3^i - J_1^i)\omega_3^i\omega_1^i + \tau_2^i)/J_2^i \\ ((J_1^i - J_2^i)\omega_1^i\omega_2^i + \tau_3^i)/J_3^i \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} 0 & \frac{J_2^i}{J_1^i}\omega_3^i & -\frac{J_3^i}{J_1^i}\omega_2^i \\ \frac{J_3^i}{J_2^i}\omega_3^i & 0 & -\frac{J_1^i}{J_2^i}\omega_1^i \\ \frac{J_1^i}{J_3^i}\omega_2^i & -\frac{J_2^i}{J_3^i}\omega_1^i & 0 \end{bmatrix}}_{Y^i} \begin{bmatrix} \omega_1^i \\ \omega_2^i \\ \omega_3^i \end{bmatrix} + \underbrace{\begin{bmatrix} 1/J_1^i & 0 & 0 \\ 0 & 1/J_2^i & 0 \\ 0 & 0 & 1/J_3^i \end{bmatrix}}_{(J^i)^{-1}} \begin{bmatrix} \tau_1^i \\ \tau_2^i \\ \tau_3^i \end{bmatrix}. \end{aligned} \quad (10)$$

Euler's first-order discretization of Eq. (10) is

$$\omega^i(k+1) = (\mathbf{I}_3 + \Delta t Y^i(k))\omega^i(k) + \Delta t (J^i)^{-1} \tau^i(k), \quad (11)$$

where  $\omega_j^i$  is the rotational velocity,  $J_j^i$  is the moment of inertia, and  $\tau_j^i$  is the control torque, of the  $i^{th}$  rigid body along the three principal axes  $j=1, 2, 3$ . Combining Eqs. (9) and (11) in stacked vector form yields.

$$\underbrace{\begin{bmatrix} -\Delta t (J^i)^{-1} & \mathbf{I}_3 & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{4 \times 3} & -\frac{\Delta t}{2} \Pi^i(k+1) & \mathbf{I}_4 \end{bmatrix}}_{F^i(k)} \underbrace{\begin{bmatrix} \tau^i(k) \\ \omega^i(k+1) \\ q^i(k+2) \end{bmatrix}}_{\mathfrak{T}^i(k+1)} = \underbrace{\begin{bmatrix} (\mathbf{I}_3 + \Delta t Y^i(k))\omega^i(k) \\ q^i(k+1) \end{bmatrix}}_{y^i(k)} \quad (12)$$

The typical task of controller synthesis is to determine the torque  $\tau^i$  that stabilizes the system.

### 3.2. Basic consensus theory

The problem of *consensus* theory is to create distributed protocols based on *communication graphs* which can drive the states of a team of communicating agents to a common state or an agreed state. Where the agents  $i$  ( $i=1, \dots, n$ ) are represented by vertices of the communication graph; the edges of the graph are the communication links between them. Let the state of agent (vehicle)  $i$  be  $x^i$ , and  $\mathbf{x}$  is the stacked vector of all the states of the vehicles. For systems modeled



by first-order dynamics, the following first-order consensus protocol (or similar protocols) has been proposed, for example [16, 17]:

$$\dot{\mathbf{x}}(t) = -\mathbf{L}(\mathbf{x}(t) - \mathbf{x}^{off}). \quad (13)$$

We know that consensus has been achieved when  $\|x^i - x^j\| \rightarrow (x^{ij})^{off}$  as  $t \rightarrow \infty$ ,  $\forall i \neq j$ . A more comprehensive analysis of the mathematical basis of graph theoretic consensus theory can be found in [10].

Now we state the limitations of consensus theory that motivates our work. First, the basic consensus protocol Eq. (13) does not admit quaternions directly because quaternion dynamics are highly nonlinear. It violates quaternion unit norm requirements, and therefore we cannot practically apply Eq. (6) with consensus directly. To extend Eq. (13) to attitude quaternions, we proposed the following consensus protocol for quaternions [7–10]:

$$\dot{\mathbf{q}}(t) = -\mathbf{P}(t)(\mathbf{q}(t) \ominus \mathbf{q}^{-off}). \quad (14)$$

Here,  $\mathbf{P}(t)$  is a Laplacian-like stochastic matrix whose values are partially unknown, but a Laplacian-like structure is imposed on it by optimization, and  $\mathbf{q}(t) = [q^1(t), q^2(t) \cdots q^n(t)]^T$ . We present more analysis of  $\mathbf{P}(t)$  in the “Solutions” section.

## 4. Solutions

We present a four-step solution to the problem statement in Section 2 [7–10], listed as follows: (1) development of a consensus protocol for quaternions, (2) development of collision avoidance behavior for quaternion consensus, (3) determining obstacle vectors in different coordinate frames and (4) integration of quaternion consensus with Q-CAC avoidance.

### 4.1. Development of a consensus protocol for quaternions

To handle the difficulty of non-linearity in quaternion kinematics, we develop a consensus protocol especially for quaternions. We adopt an optimization approach and cast the problem as a semidefinite program, which is subject to convex quadratic constraints, stated as linear matrix inequalities (LMI). Based on the current communication graph of any  $SC_i$ , a series of Laplacian-like matrices  $\mathbf{P}^i(t)$  are synthesized each time step to drive  $q^i(t)$  to consensus while satisfying quaternion kinematics:

$$\dot{q}^i(t) - \mathbf{P}^i(t) \left[ q_1^T(t) q_2^T(t) \cdots q_y^T(t) \right], \quad (15)$$

where  $q_1^T(t) q_2^T(t) \cdots q_y^T(t)$  are the quaternions of the  $y$  other neighboring SC which  $SC_i$  can communicate with at time  $t$ . Euler’s first-order discretization of Eq. (15) is

$$q_{k+1}^i = q_k^i - \Delta t \underbrace{\left[ y\Lambda_1^i(t) - \Lambda_2^i(t) \cdots - \Lambda_y^i(t) \right]}_{\mathbf{P}^i(t)} \left[ q_1^T(t) q_2^T(t) \cdots q_y^T(t) \right], \quad (16)$$

where  $\Lambda^i(t) > 0$  is an unknown positive definite optimization matrix variable, whose components are chosen by the optimization process. For analysis purposes, we shall now reconsider the collective quaternion consensus dynamics Eq. (14). The components of  $\mathbf{P}(t)$  are

$$\mathbf{P}(t) = \underbrace{\begin{bmatrix} \Lambda^1(t) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Lambda^n(t) \end{bmatrix}}_{\Lambda(t)} \underbrace{\begin{bmatrix} l_{11}\mathbf{I}_4 & \cdots & l_{1n}\mathbf{I}_4 \\ \vdots & \ddots & \vdots \\ l_{n1}\mathbf{I}_4 & \cdots & l_{nn}\mathbf{I}_4 \end{bmatrix}}_{\Gamma = \mathbf{L} \otimes \mathbf{I}_4} \quad (17)$$

where  $\Gamma$  is composed of components of the Laplacian  $\mathbf{L} = [l_{ij}]$  ( $i, j = 1, \dots, n$ ), which gives  $\mathbf{P}(t)$  its Laplacian-like behavior, and  $\Lambda^i(t) > 0$  is as previously defined.

We now present the proof of stability of  $\mathbf{P}(t)$ , that is, that Eq. (14) does indeed achieve consensus. Different versions of all the theorems, lemmas and proofs in this section had been presented in [7–10]. Let us begin by recalling the following standard result on a *matrix pencil* [18].

**Theorem 1:** For a *symmetric-definite pencil*  $\mathbf{A} - \lambda\mathbf{B}$ , there exists a nonsingular  $\mathbf{Z} = [z_1, \dots, z_n]$  such that

$$\mathbf{Z}^T \mathbf{A} \mathbf{Z} = \text{diag}(a_1, \dots, a_n) = D_A, \quad (18)$$

$$\mathbf{Z}^T \mathbf{B} \mathbf{Z} = \text{diag}(b_1, \dots, b_n) = D_B. \quad (19)$$

Moreover,  $\mathbf{A}z_i = \lambda_i \mathbf{B}z_i$  for  $i = 1, \dots, n$ , where  $\lambda_i = a_i/b_i$ .

**Lemma 1:** For any time  $t$ , the eigenvalues of  $\mathbf{P}(t)$  are  $\gamma_i \eta_i(t)$ . Here,  $\gamma_i$  are the eigenvalues of  $\Gamma$  and  $\eta_i(t)$  the eigenvalues of  $\Lambda(t)$ . It can therefore be observed that  $\mathbf{P}(t)$  has only four zero eigenvalues; the rest of its eigenvalues are strictly positive.

**Proof:** To find the eigenvalues of  $\mathbf{P}(t)$ , consider a scalar  $\lambda$  such that for some nonzero vector  $z$ :

$$\Gamma z = \lambda \Lambda^{-1}(t) z. \quad (20)$$

Eq. (20) defines a *symmetric-definite generalized eigenvalue problem* (SDGEP), where  $\Gamma - \lambda \Lambda^{-1}(t)$  defines a matrix pencil. Theorem 1 therefore immediately implies that the eigenvalues of  $\mathbf{P}(t)$  are  $\gamma_i \eta_i(t)$ . It is also easy to observe (or show numerically) that due to the property of the Laplacian matrix  $\mathbf{L}$ ,  $\mathbf{P}(t)$  has positive eigenvalues except for four zero eigenvalues. This proves the claim.

**Theorem 2:** The time-varying system Eq. (14) achieves consensus.

**Proof:** For simplicity, we shall assume no offsets are defined, that is,  $\mathbf{q}^{off} = \mathbf{0}$  (or  $(q^{off})^i = [0 \ 0 \ 0 \ 1]^T \forall i$ ). By consensus theory, when  $\mathbf{q}$  has entered the *consensus space*  $\mathcal{C} = \{\mathbf{q} \mid q^1 = q^2 = \dots = q^n\}$ , then  $\dot{\mathbf{q}} = \mathbf{0}$  (i.e. no vehicles are moving anymore).  $\mathcal{C}$  is the *nullspace* of  $\mathbf{P}(t)$ , that is, the set of all  $\mathbf{q}$  such that  $\mathbf{P}(t)\mathbf{q} = \mathbf{0}$ . Therefore,  $\mathbf{q}$  stays in  $\mathcal{C}$  once it enters there.

Suppose that  $\mathbf{q}$  has not entered  $\mathcal{C}$  (i.e.  $\dot{\mathbf{q}} \neq \mathbf{0}$ ), then consider a Lyapunov candidate function  $V = \mathbf{q}^T \Gamma \mathbf{q}$ ;  $V > 0$  unless  $\mathbf{q} \in \mathcal{C}$ . Then:

$$\begin{aligned} \dot{V} &= \mathbf{q}^T \Gamma \dot{\mathbf{q}} + \dot{\mathbf{q}}^T \Gamma \mathbf{q}, \\ &= -\mathbf{q}^T \Gamma \mathbf{P}(t) \mathbf{q} - \mathbf{q}^T \mathbf{P}(t) \Gamma \mathbf{q}, \\ &= -\mathbf{q}^T \Gamma \Lambda(t) \Gamma \mathbf{q} - \mathbf{q}^T \Gamma \Lambda \Gamma \mathbf{q}, \\ &= -2\mathbf{q}^T \Gamma \Lambda(t) \Gamma \mathbf{q}, \\ &= -2s^T \Lambda(t) s, \end{aligned} \tag{21}$$

where  $s = \Gamma \mathbf{q} \neq \mathbf{0}$  for  $\mathbf{q} \notin \mathcal{C}$ , which implies that  $\mathbf{q}$  approaches a point in  $\mathcal{C}$  as  $t \rightarrow \infty$ . This proves the claim. Eq. (21) is true as long as  $\mathbf{L}$  is nonempty, that is, some vehicles can sense, see or communicate with each other all the time.

#### 4.2. Development of collision avoidance behavior for quaternion consensus

Eq. (15) or (16) will indeed generate a consensus  $q^i(t)$  for any  $SC_i$ , but the system still needs to determine whether the trajectory is safe or not. This brings us to the issue of avoidance. Any rigid appendage attached to the body of  $SC_i$ , for example, a camera, whose direction vector is  $v_{cam_i}^I$  in *inertial frame*, can be transformed to the spacecraft fixed *body frame* by the rotation:

$$v_{cam_i}^B(t) = \mathcal{R}_i^{-1}(t) v_{cam_i}^I(t). \tag{22}$$

where

$$\mathcal{R}_i(t) = \left( (2q_4^i(t))^2 - 1 \right) \mathbf{I}_3 + 2\bar{q}^i(t) \bar{q}^i(t)^T - 2q_4^i(t) \bar{q}^i(t)^\times \tag{23}$$

is the rotation matrix corresponding to the  $q^i(t)$  at time  $t$ ;  $\bar{q}^i(t)^\times$  is the *antisymmetric matrix* [19]. For a simpler analysis, let us consider a single  $SC_i$  with a single camera,  $v_{cam_i}^I$ , and  $m$  (possibly, time-varying) obstacles,  $v_{obs_i,j}^I$  ( $j = 1, \dots, m$ ), defined in  $\mathcal{F}_{SC_i}^I$ . We want  $v_{cam_i}^I$  to avoid all  $v_{obs_i,j}^I$  when  $SC_i$  is re-orientating. Then following Eq. (3), the resulting attitude constraint of Eq. (2) can be written as

$$q^i(t)^T \tilde{A}_j^i(t) q^i(t) \leq 0. \tag{24}$$

Its LMI equivalent [5] is

$$\begin{bmatrix} \mu & q^i(t)^T \\ q^i(t) & (\mu \mathbf{I}_4 + \tilde{A}_j^i(t))^{-1} \end{bmatrix} \geq 0. \quad (25)$$

where

$$\tilde{A}_j^i(t) = \begin{bmatrix} A_j(t) & b_j(t) \\ b_j(t)^T & d_j(t) \end{bmatrix} \in \mathbb{R}^{4 \times 4}, \quad (26)$$

and

$$A_j(t) = v_{cam_i}^B(t) v_{obs_i,j}^I(t)^T + v_{obs_i,j}^I(t) v_{cam_i}^B(t)^T - (v_{cam_i}^B(t)^T v_{obs_i,j}^I(t) + \cos \theta) \mathbf{I}_3, \quad (27)$$

$$b_j(t) = -v_{cam_i}^B(t) \times v_{obs_i,j}^I(t), \quad (28)$$

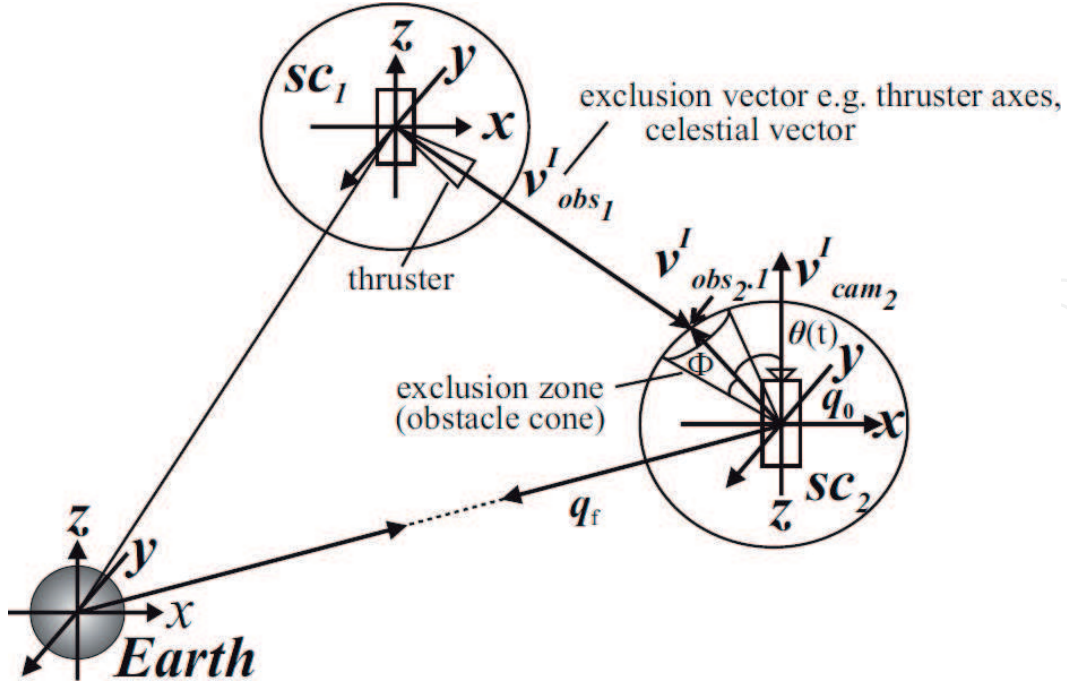
$$d_j(t) = v_{cam_i}^B(t)^T v_{obs_i,j}^I(t), \quad (29)$$

for  $j=1, \dots, m$ .

Eq. (24) defines the set of attitude quaternions  $q^i(t)$  to satisfy the constraint  $v_{cam_i}^I(t)^T v_{obs_i,j}^I(t) \geq \cos \theta \forall t \in [t_0, t_f]$ , so it is used to find a collision-free  $v_{cam_i}^I(t)$ . In Eq. (25),  $\mu$  is chosen to ensure that  $\mu \mathbf{I}_4 + \tilde{A}_j^i(t)$  is positive definite.

However, the solution presented above assumes that  $v_{cam_i}^B(t)^T$  and  $v_{obs_i,j}^I(t)$  are in the same coordinate frame and that  $v_{obs_i,j}^I(t)$  is static, so  $t$  is constant. In reality, this is not so. To address such a practical issue, we present a mechanism to calculate  $v_{obs_i,j}^I$  (defined in  $\mathcal{F}_{SC_i}^I$ ) corresponding to  $v_{obs_j}^I$  (defined in  $\mathcal{F}_{SC_j}^I$ ) ( $v_{obs_i,j}^I$  means the obstacle vector originated from the rotating frame of  $SC_j$  but defined in  $\mathcal{F}_{SC_i}^I$ ). This is essentially a mechanism to determine the intersection point of  $v_{obs_j}^I(t)$  with the sphere of radius  $r$ , centered on  $SC_i$ . If indeed such an intersection exists, it defines  $v_{obs_i,j}^I$  which can be used to define an attitude constraint represented as Eq. (24) to be avoided by  $SC_i$ .

The scenario is illustrated in **Figure 2**, where  $SC_1$  and  $SC_2$  are shown in their different coordinate frames relative to Earth. A thruster attached to  $SC_1$  body frame is at  $v_{obs_1}^I$ , while the circles around  $SC_1$  and  $SC_2$  are spheres representing the coordinate frames from which their attitude evolves. If both spacecraft are close enough, then vector  $v_{obs_1}^I$  may intersect a point on the sphere of  $SC_2$ , whereby the intersection defines  $v_{obs_2,1}^I$  in the frame of  $SC_2$ . The requirement is that as  $SC_2$  changes its attitude from  $q_0$  to  $q_f$ ,  $v_{cam_2}^I$  must avoid the cone created around  $v_{obs_2,1}^I \forall t \in [t_0, t_f]$ .



**Figure 2.** Q-CAC problem in different frames.  $SC_2$  must maneuver from  $q_0$  to  $q_f$ , while  $v_{cam_2}^I$  must avoid  $v_{obs_{2,1}}^I$  by at least  $\emptyset \forall t \in [t_0, t_f]$ .

#### 4.3. Determination of obstacle vectors in different coordinate frames

Pursuing the issue of practicality further, given  $SC_i$  in  $\mathcal{F}_{SC_i}^I$  and  $SC_j$  in  $\mathcal{F}_{SC_j}^I$  with emanating vectors, an intersection between vectors emanating from  $\mathcal{F}_{SC_j}^I$  with the sphere centered on  $\mathcal{F}_{SC_i}^I$  can be determined, either by using onboard sensors or by application of computational geometry. Given a line segment  $[p_1, p_2]$ , originating at  $p_1$  and terminating at  $p_2$ , a point  $p = [p_x \ p_y \ p_z]^T$  on  $[p_1, p_2]$  can be tested for intersection with a sphere centered at an external point  $p_3$  with radius  $r$  [20]. Therefore, for any  $v_{obs_j}^I(t)$  in  $\mathcal{F}_{SC_j}^I$ , if an intersection point  $p(t)$  exists at time  $t$  with the sphere centered on  $\mathcal{F}_{SC_i}^I$  with radius  $r$ , then  $v_{obs_{i,j}}^I(t) = p(t)$ ; otherwise, one can set  $v_{obs_{i,j}}^I(t) = -v_{cam_i}^I(t)$  to show that no constraint violation has occurred. The value of  $r$  will thus depend on the current application but must be proportional to the urgency of avoiding obstacle vectors originating from other spacecraft. The above formulation effectively completes the decentralization of the avoidance problem which has already been partly decentralized by Eq. (16). Eq. (16) will be written in a semidefinite optimization program, which gives us the privilege to apply further constraints. Therefore, the *norm constraints* required by quaternion kinematics can be enforced as follows:

$$q_k^{iT} (q_{k+1}^i - q_k^i) = 0 \quad (30)$$

Essentially, Eq. (30) is the discrete time version of  $\dot{q}^i(t)^T \dot{q}^i(t) = 0$  or  $\mathbf{q}(t)^T \dot{\mathbf{q}}(t) = 0$ . This guarantees that  $\dot{q}^i(t)^T \dot{q}^i(t) = 1$  or  $\mathbf{q}(t)^T \dot{\mathbf{q}}(t) = n$  for  $nSC$ , iff  $\|q^i(0)\| = 1 \forall i$ .

#### 4.4. Integration of quaternion consensus with Q-CAC avoidance

The integration of the quaternion consensus protocol with the Q-CAC collision avoidance in different coordinate frames is a two-stage process. First, the quaternion consensus protocol generates a set of consensus quaternion trajectories using Eq. (15) or (16). Then Eq. (25) tests whether the generated sequence is safe or not. If the next safe quaternion trajectory  $q_{safe}^i$  has been determined, the control torque  $\tau^i$  and angular velocity  $\omega^i$  to rotate the  $SC_i$  optimally to  $q_{safe}^i$  can be determined by using the normal quaternion dynamics Eq. (12). Otherwise, Eq. (25) adjusts the  $q_{unsafe}^i$  to generate a  $q_{safe}^i$ , which will be close to but not be exactly  $q_{safe}^i$ . The cycle repeats until consensus is achieved.

Using semidefinite programming, the solutions presented previously are cast as an optimization problem, augmented with a set of LMI constraints and solved for collision-free consensus quaternion trajectories. We consider the algorithm in discrete time. Given the initial attitude  $q^i(0)$  of  $SC_i$ , ( $i=1, \dots, n$ ), find a sequence of consensus quaternion trajectories that satisfies the following constraints:

$$\begin{aligned} q_{k+1}^i &= q_k^i - \Delta t \mathbf{P}^i(t) \mathbf{q}_k^i, \\ \mathbf{q}_k^i T(\mathbf{q}_{k+1}^i - q_k^i) &= 0, \\ \begin{bmatrix} \mu & q_i(t)^T \\ q_i(t) & (\mu \mathbf{I}_4 + \tilde{A}_j^i(t))^{-1} \end{bmatrix} &\geq 0. \end{aligned} \quad (31)$$

$$\underbrace{\begin{bmatrix} -\Delta t (f^i)^{-1} & \mathbf{I}_3 & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{4 \times 3} & -\frac{\Delta t}{2} \mathbf{\Pi}_{k+1}^i & \mathbf{I}_4 \end{bmatrix}}_{F_k^i} \underbrace{\begin{bmatrix} \tau_k^i \\ \omega_{k+1}^i \\ q_{k+2}^i \end{bmatrix}}_{\mathfrak{T}_{k+1}^i} = \underbrace{\begin{bmatrix} (\mathbf{I}_3 + \Delta t \mathbf{Y}_k^i) \omega_k^i \\ q_{k+1}^i \end{bmatrix}}_{y_k^i}$$

## 5. Simulation results

We shall present only three results for attitude multi-path planning in different coordinate frames due to limitation of space. These results will partly be found in [7–10]. For the SDP programming and simulation, we used the available optimization software tools SeDuMi [21] and YALMIP [22] running inside Matlab®.

### 5.1. Q-CAC avoidance in different coordinate frames without consensus

In this experiment  $SC_1$  and  $SC_2$  are changing their orientation to point an instrument to Earth. They are close to each other, and their thrusters can cause plume impingements to damage each other. Their initial quaternions are  $q_0^1 = q_0^2 = [0 \ 0 \ 0 \ 1]^T$ . The desired final quaternions are.

$$\begin{aligned} q_f^1 &= [0.2269 \ 0.0421 \ 0.9567 \ 0.1776]^T \\ q_f^2 &= [0 \ 0 \ 0.9903 \ 0.1387]^T. \end{aligned} \quad (32)$$



Three thrusters of  $SC_1$  in  $\mathcal{F}_{SC_1}^B$  are

$$\begin{aligned} v_{obs1.1}^B &= [-0.2132 \ -0.0181 \ 0.9768]^T \\ v_{obs1.2}^B &= [0.314 \ 0.283 \ -0.906]^T \\ v_{obs1.3}^B &= [-0.112 \ -0.133 \ -0.985]^T. \end{aligned} \quad (33)$$

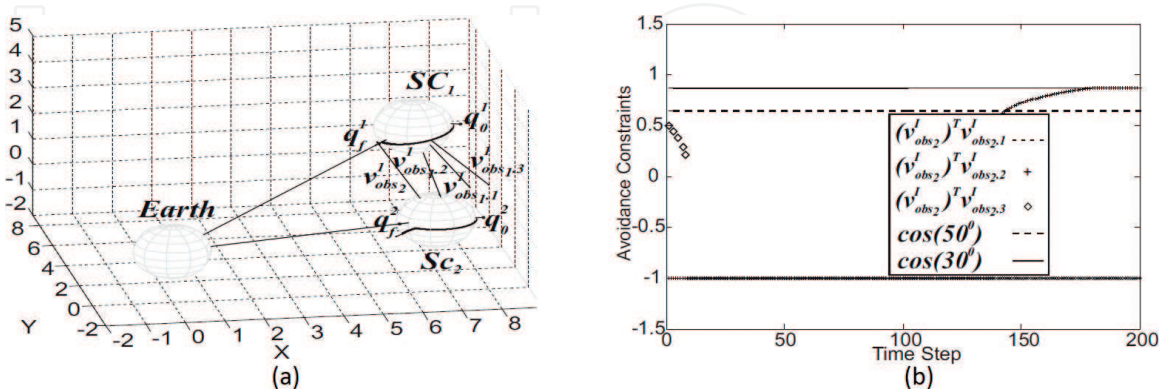
A single thruster of  $SC_2$  in  $\mathcal{F}_{SC_2}^B$  is

$$v_{obs2}^B = [0.02981 \ 0.0819 \ 0.9962]^T. \quad (34)$$

It is desired that  $v_{obs2}^I$  avoid  $v_{obs1.1}^I$  by  $50^\circ$  and avoid  $v_{obs1.2}^I$  and  $v_{obs1.3}^I$  by  $30^\circ$ , while both are maneuvering to their desired final attitudes. The trajectories obtained are shown in **Figure 3 (a)** and **(b)**. This experiment demonstrates that when both constraints are in conflict, the avoidance constraint is superior to the desired final quaternion constraint. As seen from (a),  $SC_2$  cannot reconfigure exactly to the desired  $q_f^2$  due to the satisfaction of the avoidance constraints. This can be resolved by changing either the position of  $SC_2$  or  $SC_1$ .

## 5.2. Consensus with Q-CAC avoidance in different coordinate frames

In this experiment  $SC_i$  ( $i=1, 2, 3$ ) will maneuver to a consensus attitude. Each carries a sensitive instrument  $v_{cam_i}^I$  pointing in the direction  $SC_i$ 's initial attitude quaternion. In addition, each  $SC_i$  has only one thruster pointing to the opposite (rear) of  $SC_i$ 's initial attitude. It is desired that the time evolution of the attitude trajectory of the sensitive instrument avoids the thruster plumes emanating from each of the two other SC by  $30^\circ$ . From the generated initial quaternions, there is possibility of intersection of the thrusters of  $SC_1$  and  $SC_3$ , with  $SC_2$ , and the thruster of  $SC_2$  may impinge on  $SC_1$  or  $SC_3$  at any time  $k$ .



**Figure 3.** (a) shows the avoidance between thrusters of  $SC_1$  and  $SC_2$  during reorientation to Earth:  $SC_2$  cannot reconfigure to the desired  $q_f^2$  due to the avoidance constraints. Note that  $v_{obs1.1}^I, v_{obs1.2}^I$  and  $v_{obs1.3}^I$  are the points of intersections of  $v_{obs1.1}^I, v_{obs1.2}^I$  and  $v_{obs1.3}^I$  with  $SC_2$ . (b) Satisfaction of avoidance constraints: the sudden jumps to and from  $-1$  indicate times when any of  $v_{obs1.1}^I, v_{obs1.2}^I$  and  $v_{obs1.3}^I$  lost intersection with the sphere of  $SC_2$  and therefore was replaced with  $-v_{obs1.i}^I, i = 1, \dots, 3$ .



The initial positions are

$$\begin{aligned}\mathcal{F}_{SC_1}^I &= [-2 \ 0 \ 2]^T \\ \mathcal{F}_{SC_2}^I &= [0.5 \ 0 \ 2]^T \\ \mathcal{F}_{SC_3}^I &= [3 \ 0 \ 2]^T.\end{aligned}\quad (35)$$

A set of initial quaternions were randomly generated, with the following data:

$$\begin{aligned}q_0^I &= [-0.5101 \ 0.6112 \ -0.3187 \ -0.5145]^T \\ q_0^2 &= [-0.9369 \ 0.2704 \ -0.1836 \ -0.124]^T \\ q_0^3 &= [0.1448 \ -0.1151 \ 0.1203 \ 0.9753]^T.\end{aligned}\quad (36)$$

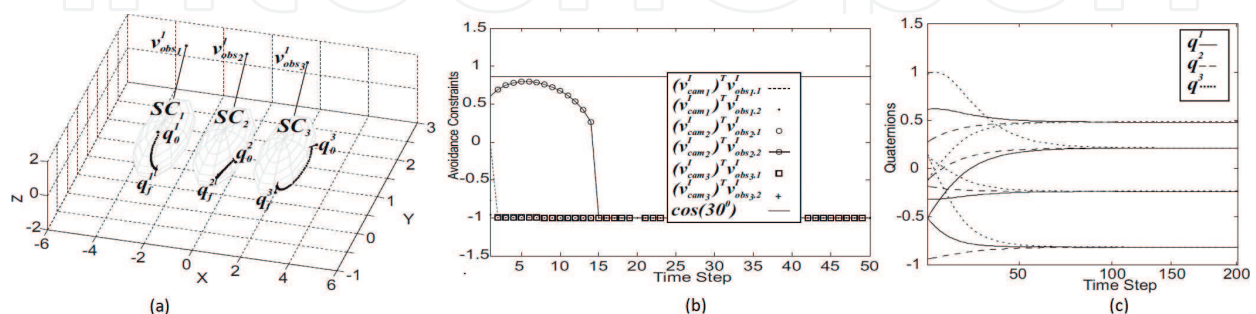
**Figure 4** (a) shows the solution trajectories while (b) shows the avoidance graph; no constraints are not violated; (c) shows the consensus graph. The final consensus quaternion is  $q_f = [-0.8167 \ 0.4807 \ -0.2396 \ 0.2112]^T$ , which is the normalized average of the initial attitude quaternions. This proves that consensus is indeed achieved by Eq. (16).

### 5.3. Consensus-based attitude formation acquisition with avoidance

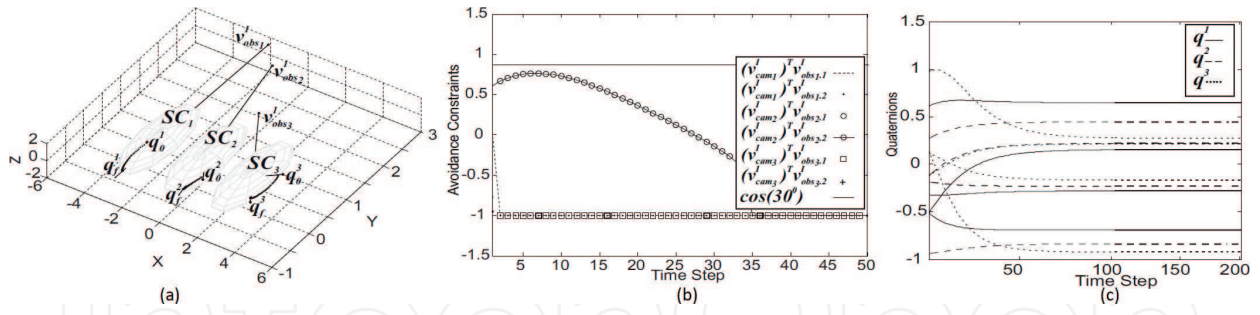
This experiment is to test the capability of the quaternion consensus algorithm in attitude formation acquisition.  $SC_i$  ( $i=1,2,3$ ) will maneuver to a consensus formation attitude, with relative offset quaternions defined to enable the sensitive instruments to point at  $30^\circ$  offsets from each other about the z-axis. The previous set of initial data for  $q_0^i$  and  $\mathcal{F}_{SC_i}^I$  were used. Like the previous experiment, it is desired that the sensitive instruments avoid the thruster plumes emanating from each of the two other SC by an angle of  $30^\circ$ .

The relative offsets are defined as

$$\begin{aligned}q_1^{off} &= [0 \ 0 \ 0 \ 1]^T \\ q_2^{off} &= [0 \ 0 \ 0.2588 \ 0.9659]^T \\ q_3^{off} &= [0 \ 0 \ 0.5 \ 0.866]^T.\end{aligned}\quad (37)$$



**Figure 4.** (a) Reorientation to consensus attitude with intervehicle thruster plume avoidance, (b) avoidance constraints graph and (c) attitude consensus graph.



**Figure 5.** (a) Reorientation to consensus formation attitude with intervehicle thruster plume avoidance, (b) avoidance constraints graph and (c) attitude consensus graph.

**Figure 5** (a) shows the trajectories, while (b) shows the avoidance graph; no constraints are violated. Finally, (c) shows the consensus graph. The final consensus quaternions are.

$$\begin{aligned}
 q_f^1 &= [-0.6926 \ 0.6468 \ -0.2798 \ 0.1541]^T \\
 q_f^2 &= [-0.8364 \ 0.4455 \ -0.2303 \ 0.2212]^T \\
 q_f^3 &= [-0.9232 \ 0.2138 \ -0.1652 \ 0.2733]^T.
 \end{aligned} \tag{38}$$

The differences of these quaternions are  $30^\circ$  apart about the same axis. Clearly, the algorithm is capable of attitude formation acquisition with avoidance.

## 6. Conclusion

In this chapter, a method of consensus with quaternion-based attitude maneuver with avoidance, of multiple networked communicating spacecraft, was presented. The presentation is composed of aspects of solutions we previously developed, by combining consensus theory and Q-CAC optimization theory. The solutions enable a team of spacecraft to point to the same direction or to various formation patterns, while they avoid an arbitrary number of attitude obstacles or exclusion zones in any coordinate frames. The proof of stability of the Laplacian-like dynamics was also presented. Simulation results also demonstrated the effectiveness of the algorithm. We hope to implement the algorithms using rotorcraft and specialized hardware.

## Author details

Innocent Okoloko

Address all correspondence to: [okoloko@ieee.org](mailto:okoloko@ieee.org)

Department of Electrical Engineering, Universidad de Ingenieria y Tecnologia, Lima, Peru

## References

- [1] Blackwood G, Lay O, Deininger B, Gudim M, Ahmed A, Duren R, Noeckerb C, Barden B. The StarLight mission: A formation-flying stellar interferometer. In: SPIE 4852, Interferometry in Space; 22 August; Waikoloa, Hawaii. SPIE Digital Library; 2002. DOI: 10.1117/12.460942
- [2] Beichman CA. NASA's terrestrial planet finder. In: Darwin and Astronomy: The Infrared Space Interferometer; 17–19 November; Stockholm, Sweden. The Netherlands: Noordwijk; 2000. p. 29-30. DOI: ISSN/ISBN:03796566
- [3] Ahmed A, Alexander J, Boussalis D, Breckenridge W, Macala G, Mesbahi M, Martin MS, Singh G, Wong E. Cassini Control Analysis Book. Pasadena, CA: Jet Propulsion Laboratory, CALTECH technical report; 1998 NA p. DOI: NA
- [4] Wen JT, Kreutz-Delgado K. The attitude control problem. IEEE Transactions on Automatic Control. 1991;**36**(10):1148-1162. DOI: 10.1109/9.90228
- [5] Kim Y, Mesbahi M. Quadratically constrained attitude control via semidefinite programming. IEEE Transactions on Automatic Control. 2004;**49**:731-735. DOI: 10.1109/TAC.2004.825959
- [6] Kim Y, Mesbahi M, Singh G, Hadaegh FY. On the convex parameterization of constrained spacecraft reorientation. IEEE Transactions on Aerospace and Electronic Systems. 2010; **46**(3):1097-1109. DOI: 10.1109/TAES.2010.5545176
- [7] Okoloko I, Kim Y. Distributed constrained attitude and position control using graph laplacians. In: ASME Dynamic Systems and Control Conference; 13–15 September; Cambridge, Massachusetts. ASME; 2010. p. 377-383. DOI: 10.1115/DSCC2010-4036
- [8] Okoloko I, Kim Y. Attitude synchronization of multiple spacecraft with cone avoidance constraints. In: IEEE Aerospace Conference; 3–10 March; Big Sky, Montana. IEEEExplore; 2012. p. 1-10. DOI: 10.1109/AERO.2012.6187119
- [9] Okoloko I, Kim Y. Attitude synchronization of multiple spacecraft with cone avoidance constraints. Systems & Control Letters. 2014;**69**:73-79. DOI: <https://doi.org/10.1016/j.sysconle.2014.04.008>
- [10] Okoloko I. Multi-Path Planning and Multi-Body Constrained Attitude Control [dissertation]. Stellenbosch, South Africa: PhD Thesis: Stellenbosch University; 2012. p. 185. Available from: <http://hdl.handle.net/10019.1/71905>
- [11] Bullo F, Murray R M, Sarti A. Control on the sphere and reduced attitude stabilization. In: IFAC symposium on nonlinear control systems; 25-28 June; Tahoe City, CA. Elsevier; 1995. p. 495-501. DOI: [https://doi.org/10.1016/S1474-6670\(17\)46878-9](https://doi.org/10.1016/S1474-6670(17)46878-9)
- [12] Ren W. Distributed attitude alignment in spacecraft formation flying. International Journal of Adaptive Control and Signal Processing. 2006;**21**(2–3):95-113. DOI: 10.1002/acs.916

- [13] Fax AJ. Optimal and Cooperative Control of Vehicle Formations [thesis]. Pasadena, CA: PhD Thesis: CALTECH; 2002. p. 135. Available from: [thesis.library.caltech.edu/4230/1/Fax\\_ja\\_2002.pdf](http://thesis.library.caltech.edu/4230/1/Fax_ja_2002.pdf)
- [14] Dimarogonas DV, Tsiotras P, Kyriakopoulos KJ. Leader-follower cooperative attitude control of multiple rigid bodies. *Systems and Control Letters*, DOI. 2009;**58**(6):429-435 <https://doi.org/10.1016/j.sysconle.2009.02.002>
- [15] Kuipers JB. Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace and Virtual Reality. 1st ed. Princeton, NJ: Princeton University Press; 2002. p. 371. ISBN-13:978-0691102986
- [16] Peng L, Zhao Y, Tian B, Zhang J, Bing-Hong W, Hai-Tao Z, Zhou T. Consensus of self-driven agents with avoidance of collisions. *Physical Review*. 2009;**79**(E). DOI: 10.1103/PhysRevE.79.026113
- [17] Olfati-Saber R. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*. 2006;**51**(3):401-420. DOI: 10.1109/TAC.2005.864190
- [18] Golub GH, Van Loan CF. *Matrix Computations*. 3rd ed. Baltimore, MD: Johns Hopkins University Press; 1996. p. 699. ISBN-13: 978-0801854149
- [19] Hughes PC. *Spacecraft Attitude Dynamics*. 2nd ed. Mineola, NY: Dover Publications Inc; 2004. p. 592. ISBN13:9780486439259
- [20] Eberly DH. *3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics*. 2nd ed. London, UK: Taylor & Francis; 2012. p. 1015. DOI: ISBN: 978-0-12-229063-3
- [21] Sturm JF. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. *Optimization methods and software*. 1998;**11**(12):625-653. DOI: <http://dx.doi.org/10.1080/10556789908805766>
- [22] Lofberg J. Yalmip: A toolbox for modelling and optimization in Matlab. In: *IEEE CACSD Conference*; 2-4 Sept.; Taipei, Taiwan. *IEEEExplore*; 2004. p. 284-289. DOI: 10.1109/CACSD.2004.1393890