

Calculation and Fitting of Boundaries between Elliptic and Hyperbolic Singularities of Pyramid-type Control Moment Gyros

Hirohisa Kojima^{a,*}

^a*Department of Aerospace Engineering, Tokyo Metropolitan University, 6-6 Asahigaoka, Hino, Tokyo 191-0065, Japan*

Abstract

There exists a singularity problem in control moment gyros (CMGs). CMG singularities are classified into two types: hyperbolic and elliptic. Several gimbal steering control methods have been presented to avoid CMG singularities. Hyperbolic singularities can be avoided by null motion, but elliptic singularities cannot. The existing steering control methods are rarely designed by explicitly taking the singularity type into account. In order to effectively avoid elliptic singularities by perturbing gimbal angles, it is desirable to calculate and record the boundaries between elliptic and hyperbolic singularities in advance so that the determined boundaries can be utilized for developing model predictive steering control. To this end, the boundaries between elliptic and hyperbolic singularities of CMGs are calculated and represented in the form of fitted curves. Several numerical examples are presented to determine the perturbation gimbal angles for avoiding elliptic singularities without using singular value decomposition.

Keywords: Control moment gyros; Elliptic and hyperbolic singularities; Curve fitting

Nomenclature

\mathbf{C}	=	Jacobian matrix of the CMG system
\mathbf{C}_i	=	\mathbf{C} with i th column removed
D_i	=	$\det(\mathbf{C}_i)$
E, F, G	=	elements of the first fundamental form
\mathbf{h}	=	angular momentum vector of the CMGs
H	=	angular momentum magnitude of each CMG (= 1 Nms)
\mathbf{H}_i	=	angular momentum vector of the i^{th} CMG
J	=	objective function

*Corresponding author

Email address: hkojima@tmu.ac.jp, tel: +81-42-585-8653, fax: +81-42-583-5119 (Hirohisa Kojima)

K	=	Gaussian radius ($=1/\kappa$)
L, M, N	=	elements of the 2nd fundamental form
\mathcal{N}	=	matrix consisting of null space basis vectors
\mathcal{M}	=	singularity type determinable matrix
$P(u, v)$	=	fitted curve function
$Q_i(u, v)$	=	singular gimbal angles (δ_{s2} or δ_{s4}) determined from (u, v)
\mathbf{S}	=	singular surface
\mathbf{u}_s	=	singular vector
u, v	=	independent variables for singular surface ($=\delta_{s1}, \delta_{s3}$)
β	=	skew angle ($=\tan^{-1} \sqrt{2} = 54.73$ deg)
$\boldsymbol{\delta}$	=	gimbal angle vector ($= (\delta_1, \delta_2, \delta_3, \delta_4)^T$)
δ_i	=	gimbal angle of the i^{th} CMG, rad
δ_{si}	=	singular gimbal angle of the i^{th} CMG
κ	=	Gaussian curvature

1. Introduction

An advantage of control moment gyro (CMG) systems is that they can generate large torques compared to reaction wheels (RWs). There are many single gimbal CMG (SGCMG) array configurations, such as skew type, roof type, symmetric type, and twin type. The pyramid-type CMG system, as shown in Fig. 1, is a typical configuration for a SGCMG system. However, SGCMG systems have a singularity problem. This problem has been studied from various points of view. The singular surface of a pyramid-type CMG system was studied in [1].

The singular surfaces of a CMG system are classified into two types: hyperbolic and elliptic singularities. To overcome the singularity problem of pyramid-type SGCMGs, a number of control schemes have been proposed [2]-[21]. The logic for singularity avoidance can be generally classified into two categories: gimbal angle path planning and real-time feedback control. In the path planning methods, the gimbal angle trajectories are searched in advance so that the CMG systems do not encounter any singularities during attitude maneuvers [4, 5, 6]. However, generally speaking, those methods have high computational cost.

On the other hand, real-time steering logic methods do not have high computational cost but require singularity avoidance techniques because encountering the singularities is not predicted in advance. Real-

time singularity avoidance methods for the SGCMGs can be classified into three types: singularity-robust (SR) inverse steering laws [7, 8, 9, 10], singular direction avoidance (SDA) methods [11], and gradient methods [7, 15, 16]. The gradient methods use null motion, which can change the gimbal angles without generating torque. Therefore, null motion can be used to avoid CMG singularities. However, null motion is not perfect because it still cannot avoid elliptic singularities.

Because the singular surfaces of the pyramid-type SGCMG can be obtained in advance, recorded data of CMG singularities can be used to make both real-time singularity avoidance methods and CMG angular momentum path planning more effective. Takada et al. [20] implemented a singularity avoidance method using the surface cost function as a singularity metric calculated from the singular surface data at 3-D mesh points, and they experimentally showed that their proposed method is effective in real-time singularity avoidance. Sato and Takahashi [21] recorded a singular surface in the form of a set of plane surfaces and applied an A* algorithm [22] to the global singularity avoidance path planning of the CMG angular momentum.

Although the above two papers, which are based on recorded singularity data, succeeded in singularity avoidance, the amount of singularity data is considerably large, and the recorded data of the singularities are not precise because recording is limited to the mesh/grid points. Therefore, a more precise and efficient recording method is desired. Moreover, if an attitude tracking maneuver is required, it is desired to use null motion because torque error is not generated by null motion. However, as mentioned above, elliptic singularities cannot be avoided by null motion; thus, torque error is necessary. Even in such a case, a small torque error is still desired for precise attitude tracking maneuvers. In order to compromise between the necessary torque error and precise attitude tracking, a more efficient method of avoiding elliptic singularities is required.

To overcome the above problems, the boundaries between hyperbolic (passable) and elliptic (impassable) singularities are calculated and recorded in the form of a polynomial function of the gimbal angles using curve fitting techniques. The fitted curves of the boundaries can be used to detect the type of singularity (hyperbolic/elliptic) without singular value decomposition (SVD) and to explicitly determine the gimbal angle perturbation to avoid elliptic singularities during maneuvers. In other words, the boundaries recorded in the form of a function will be able to be used to implement model predictive control (MPC) for singularity avoidance. To the author's best knowledge, no previous papers have dealt with the problem by explicitly determining concrete values of the gimbal angle perturbations to avoid elliptic singularities. Furthermore, the amount of recorded singularity data is discussed and compared with those of Refs. [20] and [21], and the simplicity of the boundaries expressed in the form of polynomial functions

is demonstrated by several examples for determining the gimbal angle perturbations to avoid elliptic singularities.

2. Singularities of Pyramid-type SGCMGs

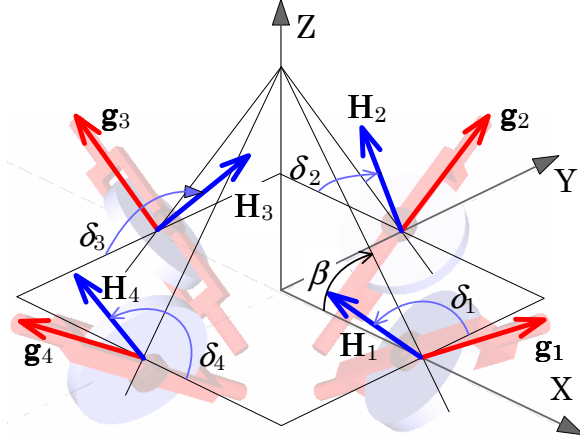


Figure 1: Pyramid-type CMG.

In this paper, a pyramid-type SGCMG (as shown in Fig. 1) is considered. In a traditional pyramid-type CMG system, the skew angle β is fixed at $\beta = \tan^{-1} \sqrt{2}$ rad ($= 54.73$ deg) because for this angle, the momentum envelope, which represents the maximum available angular momentum of the CMG for attitude maneuvers, becomes nearly spherical. The total CMG angular momentum vector for the pyramid mounting of four SGCMGs \mathbf{h} is expressed in the spacecraft reference frame as

$$\mathbf{h} = \sum_{i=1}^4 \mathbf{H}_i = H \begin{bmatrix} -c\beta \sin \delta_1 \\ \cos \delta_1 \\ s\beta \sin \delta_1 \end{bmatrix} + H \begin{bmatrix} -\cos \delta_2 \\ -c\beta \sin \delta_2 \\ s\beta \sin \delta_2 \end{bmatrix} + H \begin{bmatrix} c\beta \sin \delta_3 \\ -\cos \delta_3 \\ s\beta \sin \delta_3 \end{bmatrix} + H \begin{bmatrix} \cos \delta_4 \\ c\beta \sin \delta_4 \\ s\beta \sin \delta_4 \end{bmatrix} \quad (1)$$

where $c\beta = \cos \beta$, and $s\beta = \sin \beta$. Hereafter, the momentum magnitude for each CMG H is assumed to be unity without loss of generality.

The time derivative of the angular momentum vector for the CMGs can be obtained as

$$\dot{\mathbf{h}} = \sum_{i=1}^4 \frac{\partial \mathbf{H}_i}{\partial \delta_i} \dot{\delta}_i = \mathbf{f}_1 \dot{\delta}_1 + \mathbf{f}_2 \dot{\delta}_2 + \mathbf{f}_3 \dot{\delta}_3 + \mathbf{f}_4 \dot{\delta}_4 = \mathbf{C} \dot{\boldsymbol{\delta}} \quad (2)$$

where

$$\mathbf{C} = \begin{bmatrix} \mathbf{f}_1 & \mathbf{f}_2 & \mathbf{f}_3 & \mathbf{f}_4 \end{bmatrix} = \begin{bmatrix} -c\beta \cos \delta_1 & \sin \delta_2 & c\beta \cos \delta_3 & -\sin \delta_4 \\ -\sin \delta_1 & -c\beta \cos \delta_2 & \sin \delta_3 & c\beta \cos \delta_4 \\ s\beta \cos \delta_1 & s\beta \cos \delta_2 & s\beta \cos \delta_3 & s\beta \cos \delta_4 \end{bmatrix} \quad (3)$$

When the CMG system is in the singular state, the CMG cannot produce torque along the direction of vector \mathbf{u} perpendicular to the plane of $\dot{\mathbf{h}}$, regardless of the gimbal rate. This situation occurs when $\dot{\mathbf{h}}$ from Eq. (2) lies in a plane for any choice of $\dot{\boldsymbol{\delta}}$. The corresponding gimbal angles, momentum vector, and vector \mathbf{u} are called the singular gimbal angles, singular momentum vector, and singular vector, respectively.

2.1. Singular gimbal angles for SGCMGs

We presented how to determine the singular gimbal angles [23]. In this section, we address the sets of singular gimbal angles by referring to Ref. [23].

The singular vector \mathbf{u}_s of the Jacobian \mathbf{C} is the null space vector of \mathbf{C}^T . Nontrivial solutions for \mathbf{u}_s exist for $(\mathbf{C}\mathbf{C}^T)\mathbf{u}_s = 0$ if and only if $\mathbf{C}\mathbf{C}^T$ is singular. This singular condition can be expressed as

$$\det(\mathbf{C}\mathbf{C}^T) = \sum_{i=1}^4 D_i^2 = 0 \quad (4)$$

where

$$D_i = \det(\mathbf{C}_i) \quad (5)$$

and $\mathbf{C}_i = \mathbf{C}$ with the i^{th} column removed. Eq. (4) is called the Binet-Cauchy theorem, and its necessary and sufficient conditions are

$$D_1 = s\beta [(s_2s_3c_4 + c_2s_3s_4) + c\beta(c_2c_3s_4 - s_2c_3c_4) + 2(c\beta)^2c_2c_3c_4] = 0 \quad (6)$$

$$D_2 = s\beta [(s_3s_4c_1 + c_3s_4s_1) + c\beta(c_3c_4s_1 - s_3c_4c_1) + 2(c\beta)^2c_3c_4c_1] = 0 \quad (7)$$

$$D_3 = s\beta [(s_4s_1c_2 + c_4s_1s_2) + c\beta(c_4c_1s_2 - s_4c_1c_2) + 2(c\beta)^2c_4c_1c_2] = 0 \quad (8)$$

$$D_4 = s\beta [(s_1s_2c_3 + c_1s_1s_3) + c\beta(c_1c_2s_3 - s_1c_2c_3) + 2(c\beta)^2c_1c_2c_3] = 0 \quad (9)$$

where $s_i = \sin \delta_{si}$, and $c_i = \cos \delta_{si}$.

When $c_i \neq 0$, the singular conditions of Eqs. (6) to (9) can be simplified as

$$\tan \delta_{s3}(\tan \delta_{s2} + \tan \delta_{s4}) + c\beta(\tan \delta_{s4} - \tan \delta_{s2}) = -2(c\beta)^2 \quad (10)$$

$$\tan \delta_{s4}(\tan \delta_{s3} + \tan \delta_{s1}) + c\beta(\tan \delta_{s1} - \tan \delta_{s3}) = -2(c\beta)^2 \quad (11)$$

$$\tan \delta_{s1}(\tan \delta_{s4} + \tan \delta_{s2}) + c\beta(\tan \delta_{s2} - \tan \delta_{s4}) = -2(c\beta)^2 \quad (12)$$

$$\tan \delta_{s2}(\tan \delta_{s1} + \tan \delta_{s3}) + c\beta(\tan \delta_{s3} - \tan \delta_{s1}) = -2(c\beta)^2 \quad (13)$$

Because the minimum rank of the Jacobian matrix \mathbf{C} is 2 in the case of $0 < \beta < \pi/2$ rad, only two of the above four conditions are independent. Therefore, Eqs. (10) to (13) yield six singular gimbal angle cases. Here, we pick only one of the cases (Case 1) from Ref. [23] with $c_i \neq 0$. The other five cases can be easily obtained by cyclically changing the subscript of the gimbal angles and by adding or subtracting π from the gimbal angles because the configuration of a pyramid-type CMG is symmetric, and the gimbal angles in the opposite directions are also singular angles.

Case 1: For all $(\delta_{s1}, \delta_{s3})$, determine $(\delta_{s2}, \delta_{s4})$ using Eqs. (11) and (13).

$$\delta_{s2} = \arctan2(-2(c\beta)^2 - c\beta \tan \delta_{s3} + c\beta \tan \delta_{s1}, \tan \delta_{s1} + \tan \delta_{s3}) := Q_1(\delta_1, \delta_3) \quad (14)$$

$$\delta_{s4} = \arctan2(-2(c\beta)^2 - c\beta \tan \delta_{s1} + c\beta \tan \delta_{s3}, \tan \delta_{s1} + \tan \delta_{s3}) := Q_2(\delta_1, \delta_3) \quad (15)$$

Because the gimbal angles opposite to the above angles are also singular gimbal angles,

$$\delta_{s2} = \arctan2(-2(c\beta)^2 - c\beta \tan \delta_{s3} + c\beta \tan \delta_{s1}, \tan \delta_{s1} + \tan \delta_{s3}) + \pi := Q_3(\delta_1, \delta_3) \quad (16)$$

$$\delta_{s4} = \arctan2(-2(c\beta)^2 - c\beta \tan \delta_{s1} + c\beta \tan \delta_{s3}, \tan \delta_{s1} + \tan \delta_{s3}) + \pi := Q_4(\delta_1, \delta_3) \quad (17)$$

In addition, there are six singular gimbal angle cases with $c_i = 0$. Here, we pick two cases in which a pair of gimbal angles $(\delta_{s2}, \delta_{s4})$ are determined from another pair of gimbal angles $(\delta_{s1}, \delta_{s3})$.

Case 7: When $\cos \delta_{s1} = \cos \delta_{s3} = 0$,

$$\sin(\delta_{s2} + \delta_{s4}) = 0 \quad (18)$$

Case 9: When $\cos \delta_{s1} = 0$ for all δ_{s3} ,

$$\tan \delta_{s2} = -\tan \delta_{s4} = c\beta \quad (19)$$

2.2. Elliptic and hyperbolic singularities of SGCMGs [23]

The singularities of an SGCMG can be classified into two types: elliptic and hyperbolic. The hyperbolic singularities can be avoided through null motion. However, the elliptic singularities cannot be avoided through null motion. The null motions of CMGs are defined as gimbal motions that generate no torque. In this section, by referring to Ref. [23], it is explained how to determine the type of singularity at the given singular gimbal angles.

The condition for null motion of a pyramid-type CMG can be expressed using a Taylor series as follows:

$$\delta \mathbf{h} = \sum_{i=1}^4 \left[\frac{\partial \mathbf{H}_i}{\partial \delta_i} \Delta \delta_i + \frac{1}{2} \frac{\partial^2 \mathbf{H}_i}{\partial \delta_i^2} \Delta \delta_i^2 + O(\delta_i^3) \right]_{\delta_{si}} = \mathbf{0} \quad (20)$$

When \mathbf{u} is the singular vector \mathbf{u}_s , the inner product of \mathbf{u}_s and the first term in the right-hand side of Eq. (20) becomes zero. In addition, by recalling that

$$\frac{\partial^2 \mathbf{H}_i}{\partial \delta_i^2} = -\mathbf{H}_i \quad (21)$$

and ignoring the terms higher than the second order, the inner product of \mathbf{u}_s and Eq. (20) becomes

$$\mathbf{u}_s^T \delta \mathbf{h} = \mathbf{u}_s^T \sum_{i=1}^4 \left[-\frac{1}{2} \mathbf{H}_i \Delta \delta_i^2 \right]_{\delta_{si}} = 0 \quad (22)$$

This constraint equation can be rewritten as

$$\Delta \boldsymbol{\delta}^T \mathcal{E} \Delta \boldsymbol{\delta} = 0 \quad (23)$$

where

$$\mathcal{E} = \text{diag}(e_1, e_2, e_3, e_4) \quad (24)$$

$$e_i = \mathbf{H}_i \cdot \mathbf{u}_s, \quad (i = 1, 2, 3, 4) \quad (25)$$

When the gimbal angles are in the singular configuration, the dimension of the null space is two. Therefore, the null motion of the gimbal angles can be expressed in terms of the null space basis vectors of the Jacobian matrix \mathbf{n}_i as follows:

$$\Delta \boldsymbol{\delta} = \sum_{i=1}^2 \lambda_i \mathbf{n}_i = \mathcal{N} \boldsymbol{\lambda} \quad (26)$$

where λ_i is the i^{th} weighting coefficient on the null space basis, and $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)^T$. Substituting Eq. (26) into Eq. (23) yields

$$\boldsymbol{\lambda}^T \mathcal{M} \boldsymbol{\lambda} = 0 \quad (27)$$

where

$$\mathcal{M} = \mathcal{N}^T \mathcal{E} \mathcal{N} \quad (28)$$

If \mathcal{M} is sign-definite, then there is no solution to Eq. (27) other than $\boldsymbol{\lambda} = \mathbf{0}$. In this case, null motion is impossible at the corresponding singularity. This type of singularity is referred to as an elliptic singularity. On the other hand, if \mathcal{M} is indefinite or semidefinite, then there is a solution to Eq. (27) that satisfies $\boldsymbol{\lambda} \neq \mathbf{0}$ (i.e., null motion is possible at the singularity). This type of singularity is referred to as a hyperbolic singularity. Note that there are degenerate hyperbolic singularities that cannot be avoided by null motion.

Fig. 2(a), (b), and (c) shows the 4H, passable (hyperbolic), and impassable (elliptic) singular surfaces, respectively. The equations for the singular gimbal angles are cyclic with respect to the subscript of the gimbal number, and the singular surfaces are symmetric with respect to the $x = 0$, $y = 0$, and $z = 0$ planes. Besides, along the directions of the gimbal axis vectors, trumpet-like shapes are observed in the internal singular surfaces, and holes are observed in the momentum envelopes. These trumpet-like shapes and holes are related to the gimbal axis vectors and angle $\tan^{-1}(\cos \beta)$.

By referring to the method in [23], the singular gimbal angles (δ_2, δ_4) are determined from the pair of gimbal angles (δ_1, δ_3) . Then, the passability for the singular gimbal angles is determined, recorded, and illustrated in the passability region map.

Kojima [24] illustrated examples of passability region maps. Here, all the maps for all four cases are shown in Fig. 3, where the blue and red regions correspond to the passable and impassable singular gimbal angles, respectively. Note that these maps are calculated for every region of $\pi/4$ [rad]. Hereinafter, the region with the range of $\pi/2$ [rad] \times $\pi/2$ [rad] is referred to as a sub-domain.

It can be seen from Fig. 3 that the regions of impassable singular gimbals radiate towards four directions from some sets of gimbal angles, i.e., $(\delta_1, \delta_3) = (\pi/2, \pi/2)$, $(\delta_1, \delta_3) = (3\pi/2, 3\pi/2)$, $(\delta_1, \delta_3) = (\pi/2, 3\pi/2)$, and $(\delta_1, \delta_3) = (3\pi/2, \pi/2)$. It can also be seen that the regions of impassable singular gimbals radiate from a gimbal angle of $\tan^{-1}(\cos \beta)$, which corresponds to the center of the trumpet-like shaped parts. The impassable regions occupy several sub-domains, and such sub-domains correspond to the 4H singularities.

By comparing Fig. 3(a) with Fig. 3(d) and Fig. 3(b) with Fig. 3(c), these two region maps are line symmetric with each other with respect to line $\delta_1 + \delta_3 = 2\pi$. The maps in Fig. 3(a) are line symmetric with respect to line $\delta_1 = \delta_3$. This is because gimbals 1 and 3 are located in symmetric positions. In addition, several passable/impassable regions in one map are point symmetric. For this reason, as mentioned later, it is not always necessary to calculate all the boundaries between elliptic and hyperbolic singularities. In this paper, basic boundary curves between elliptic and hyperbolic singularities, which are needed to represent all the boundaries by rotating and transferring, will be calculated and approximated in the form of a single value function.

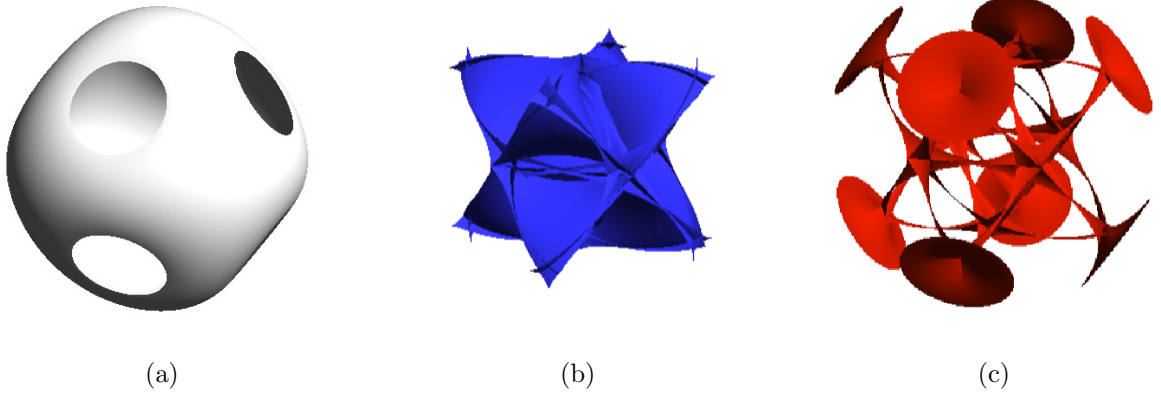


Figure 2: Singular surfaces: (a) 4H singular surface, (b) passable (hyperbolic) singular surface, and (c) impassable (elliptic) singular surface.

3. Boundaries Between Elliptic and Hyperbolic Singularities

3.1. Boundary condition of elliptic and hyperbolic singularities

At the boundaries between hyperbolic (passable) and elliptic (impassable) singular surfaces, as shown in Fig. 4, the radius of the surface is zero; in other words, the Gaussian curvature of the singular surface κ is infinity. This condition can be expressed as

$$K(\delta_{s1}, \delta_{s3}) := 1/\kappa = 0 \quad (29)$$

Because the singular surfaces are two-dimensional, they can be expressed in terms of two independent parameters. Here, gimbal angles δ_1 and δ_3 are treated as independent parameters to represent the singular surfaces. In this paper, for simplicity and referring to the parameters that are usually used in the surface geometry theory, gimbal angles δ_1 and δ_3 are hereafter denoted as u and v , respectively.

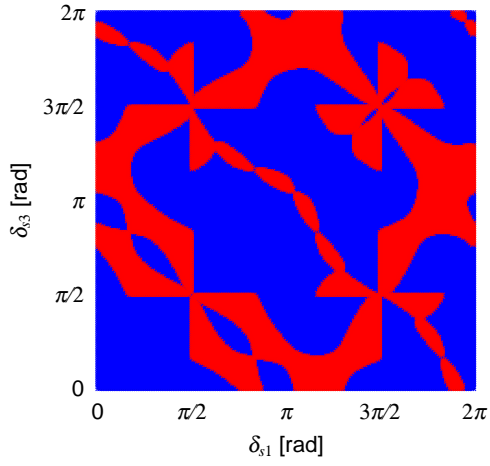
The normal vector and the elements of the first and second fundamental forms for any kind of surface $\mathcal{S}(u, v)$ can be expressed as:

$$E = \mathbf{S}_u \cdot \mathbf{S}_u \quad (30a)$$

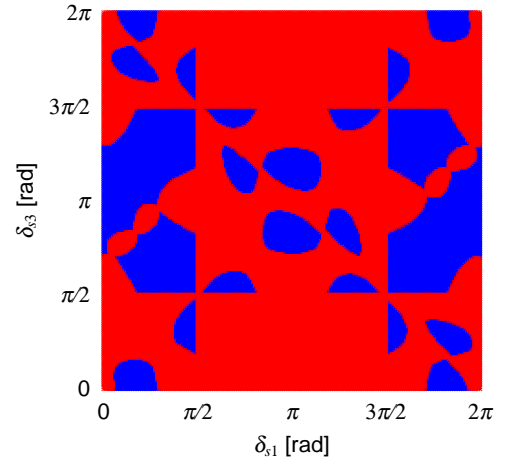
$$F = \mathbf{S}_u \cdot \mathbf{S}_v \quad (30b)$$

$$G = \mathbf{S}_v \cdot \mathbf{S}_v \quad (30c)$$

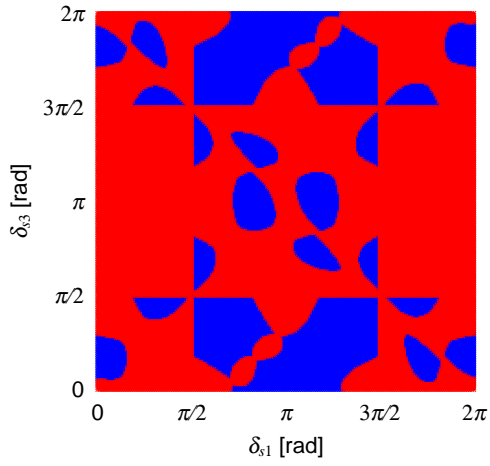
$$L = \mathbf{S}_{uu} \cdot \mathbf{n} \quad (31a)$$



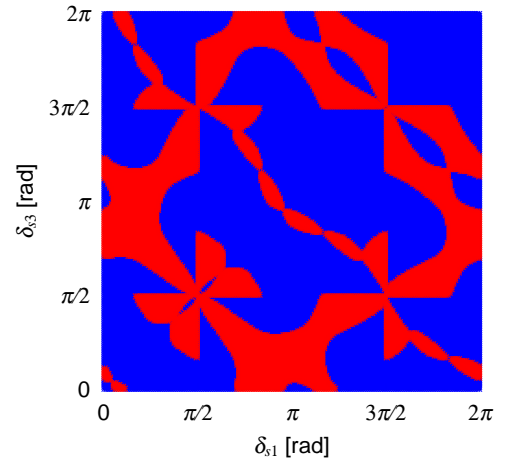
(a) case of $\delta_{s2} = Q_1(\delta_{s1}, \delta_{s3}), \delta_{4s} = Q_2(\delta_{s1}, \delta_{s3})$



(b) case of $\delta_{s2} = Q_3(\delta_{s1}, \delta_{s3}), \delta_{4s} = Q_2(\delta_{s1}, \delta_{s3})$



(c) case of $\delta_{s2} = Q_1(\delta_{s1}, \delta_{s3}), \delta_{4s} = Q_4(\delta_{s1}, \delta_{s3})$



(d) case of $\delta_{s2} = Q_3(\delta_{s1}, \delta_{s3}), \delta_{4s} = Q_4(\delta_{s1}, \delta_{s3})$

Figure 3: Mapping of passable (blue) and impassable (red) regions of the singular gimbal angles.

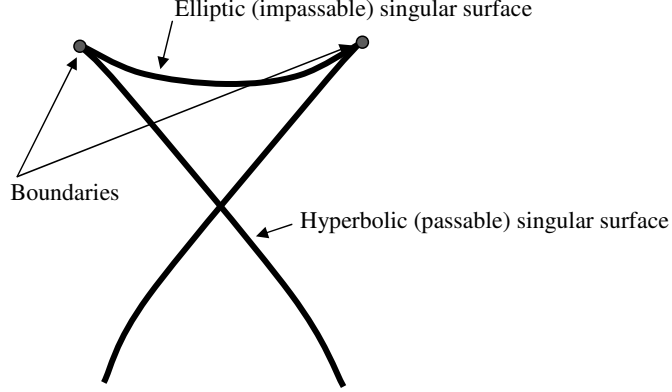


Figure 4: Schematic shape of the singular surface around the boundary between elliptic and hyperbolic singularity.

$$M = \mathbf{S}_{uv} \cdot \mathbf{n} \quad (31b)$$

$$N = \mathbf{S}_{vv} \cdot \mathbf{n} \quad (31c)$$

where \mathbf{n} is the normal vector of surfaces $\mathbf{S}(u, v)$ given by

$$\mathbf{n} = \frac{\mathbf{S}_u \times \mathbf{S}_v}{|\mathbf{S}_u \times \mathbf{S}_v|} \quad (32)$$

Using these parameters, the Gaussian curvature is given by

$$\kappa = \frac{LN - M^2}{EG - F^2} \quad (33)$$

When $EG - F^2 = 0$ or $LN - M^2 = \infty$, the condition for the boundaries between the elliptic and hyperbolic singularities is satisfied. From a practical point of view, it is unnecessary to consider $LN - M^2 = \infty$; thus, the practical condition for the boundaries is $EG - F^2 = 0$. In order to calculate E , F , and G , it is sufficient to calculate \mathbf{S}_u and \mathbf{S}_v . In addition, by taking into consideration that δ_1 and δ_3 are treated as independent parameters to represent the singular surfaces of the pyramid-type SGCMG and that δ_2 and δ_4 are parameters dependent on δ_1 and δ_3 , as shown in Eqs. (14) and (15) or Eqs. (16) and (17), the first derivatives of the singular surfaces with respect to u and v can be expressed as

$$\mathbf{S}_u = \mathbf{f}_1 + \mathbf{f}_2 \left(\frac{\partial \delta_{s2}(\delta_{s1}, \delta_{s3})}{\partial \delta_{s1}} \right) + \mathbf{f}_4 \left(\frac{\partial \delta_{s4}(\delta_1, \delta_3)}{\partial \delta_{s1}} \right) \quad (34)$$

$$\mathbf{S}_v = \mathbf{f}_2 \left(\frac{\partial \delta_{s2}(\delta_{s1}, \delta_{s3})}{\partial \delta_{s3}} \right) + \mathbf{f}_3 + \mathbf{f}_4 \left(\frac{\partial \delta_{s4}(\delta_{s1}, \delta_{s3})}{\partial \delta_{s3}} \right) \quad (35)$$

3.2. Boundary calculation by intersection tracking method

As mentioned earlier, it is not necessary to calculate all the boundaries between the elliptic and hyperbolic singularities. The minimum number of boundaries to calculate is eight. In this paper, by taking into account the results presented in Fig. 3, those curves are selected. Two of them are closed curves, and it is difficult to represent them in the form of a single value function. Therefore, in this paper, these two curves are divided into two open curves, which are treated as individual curves. In order to obtain these curves, the geometric section tracking method [25] is employed. Fig. 5 shows a flow chart for the boundary tracking method subject to one sub-domain.

The details of the procedure for boundary curve tracking are explained. Fig. 6(a), (b), (c), and (d) describes each step of the following boundary curve tracking method.

- (1) As shown in Fig. 3, several boundary curves start from one point and return to the same point. Such points are considered critical points, and special treatment is required to handle these points. The critical points can be searched by setting mesh points with small intervals in the sub-domain and checking if the normal vector of the condition surface $K(\delta_{s1}, \delta_{s3}) = 0$ at the mesh points is nearly parallel to $(0, 0, 1)^T$. If the normal vector is parallel to $(0, 0, 1)^T$, then the corresponding mesh point can be treated as a critical point, as shown in Fig. 6(a).

It should be noted that even if the search for critical points is not successful, because almost all boundary curves start from a border of the sub-domain with a $\pi/2$ rad \times $\pi/2$ rad interval, the search for points on the boundary curve will successfully end by step (4).

- (2) After detecting the critical points, boundary curve tracking is started. The curve tracking can be performed using the geometrical intersection tracking method. As explained in the previous subsection, the condition of the boundary curves is given by $1/\kappa = 0$. From this condition, the boundary curves are intersections between plane $z = 0$ and $z = K(u, v) = 1/\kappa$. The next boundary point is determined along the direction of the cross product of the normal vectors of plane $z = 0$ and $z = K(u, v) = 1/\kappa$ at the current point, as shown in Fig. 6(b).
- (3) The second step is repeated until reaching another (or the same) critical point or a border of the sub-domain, as shown in Fig. 6(c).
- (4) After all of the boundaries starting from the critical points are searched, a search for a boundary starting from the border of the sub-domain is started, and then the boundary point search process is repeated until searches from all of the borders are finished, as shown in Fig. 6(d). If a boundary

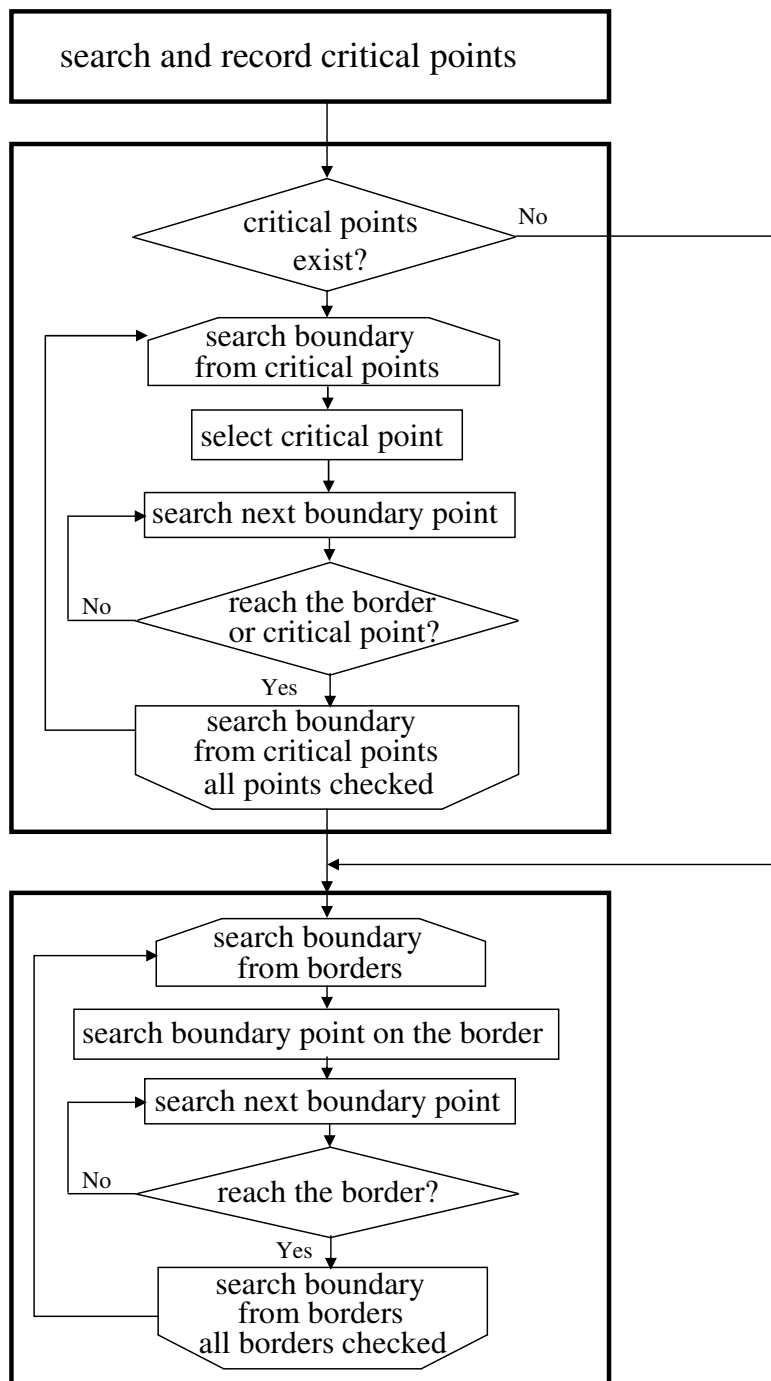


Figure 5: Flowchart for the boundary tracking method.

starting from the border is considered the end of a boundary starting from a critical point, then the search process from such a point is skipped because it is unnecessary to trace a boundary that is already obtained.

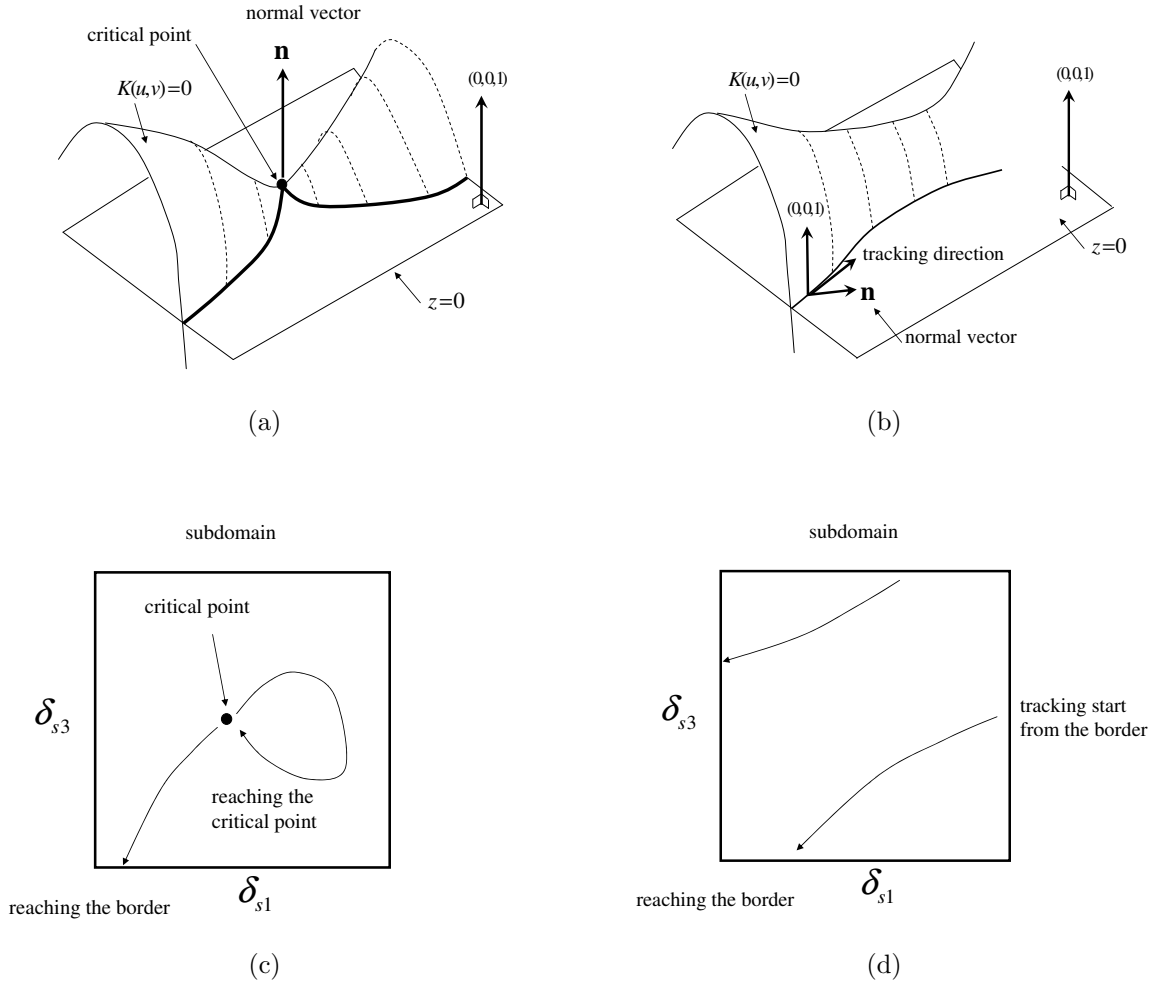


Figure 6: Each step of the boundary point search process.

Fig. 7 shows the obtained boundaries, and they are indicated in different colors for each sub-domain with $\pi/2$ rad intervals. By comparing Fig. 7 with Fig. 3, it can be said that the search for boundaries succeeded.

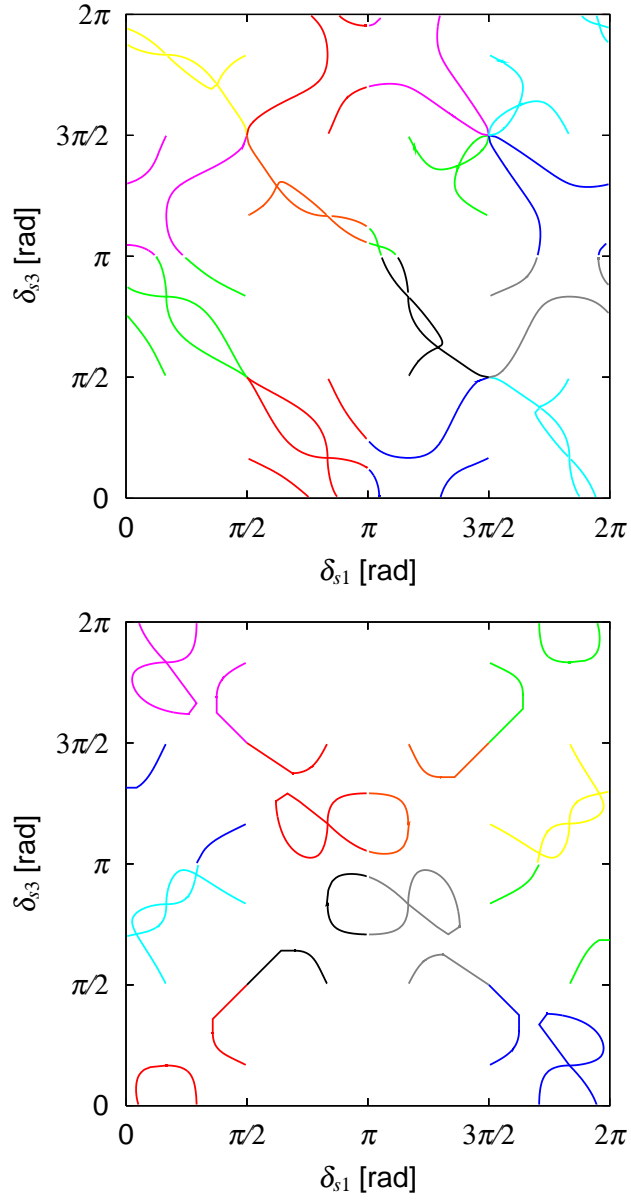


Figure 7: Boundaries indicated in different colors for each sub-domain.

4. Fitting of Boundary Curves between Elliptic and Hyperbolic Singularities

4.1. Approximate functions for the boundaries

In this subsection, approximate functions for the boundaries are explained. The Curve Fitting Toolbox in MATLAB [®] is used to obtain the fitting function of the boundaries by applying the Levenberg-Marquardt algorithm [26] as the iteration scheme. It is seen in Fig. 7(a) and (b) that there exist boundary curves that can be expressed by a single value function of the gimbal angle δ_1 or δ_3 , but several curves cannot be expressed this way. Two of the latter are closed curves, and the others are line symmetric. By taking this into account, a plan to handle such curves is considered here. The line-symmetric boundary curves are expressed as a function of a variable on a symmetric line, and the closed curves are expressed in polar coordinates about a point inside the closed curve. To easily specify the curve, the boundary curves are numbered, as shown in Fig. 8.

The boundary curves, which can be expressed by a single value function of u , are expressed in a polynomial function form as

$$v = \sum_{i=0}^j a_i \left(\frac{u - \bar{x}}{\sigma} \right)^i \quad (36)$$

where (u, v) are the coordinate variables, a_i are coefficients, and \bar{x} and σ are the mean and standard deviation of the independent variable u , respectively. The line-symmetric boundary curves are expressed in the same form as above, but the coordinate variables (u, v) are replaced with 45 deg rotated coordinate variables (u', v') so that the function becomes a single value function. Note that the order of the term j is determined under the constraint $j \leq 9$ so that the curve fitting is as precise as possible.

On the other hand, the closed curve (curve-9) is expressed in polar coordinates (r, θ) with the origin at (u_0, v_0) as

$$r(\theta) = \sum_{i=0}^9 a_i \left(\frac{\theta - \bar{x}}{\sigma} \right)^i \quad (37a)$$

$$\theta = \text{atan2}(v - v_0, u - u_0) \quad (37b)$$

and the closed curve (curve-10) is expressed in polar coordinates (r, θ) with the origin at (u_0, v_0) as

$$r(\theta) = \sum_{i=1}^4 a_{3i-3} \exp \left(- \left(\frac{\hat{\theta} - a_{3i-2}}{a_{3i-1}} \right)^2 \right) \quad (38a)$$

$$\hat{\theta} = \frac{\theta - \bar{x}}{\sigma} \quad (38b)$$

$$\theta = \text{atan2}(v - v_0, u - u_0) \quad (38c)$$

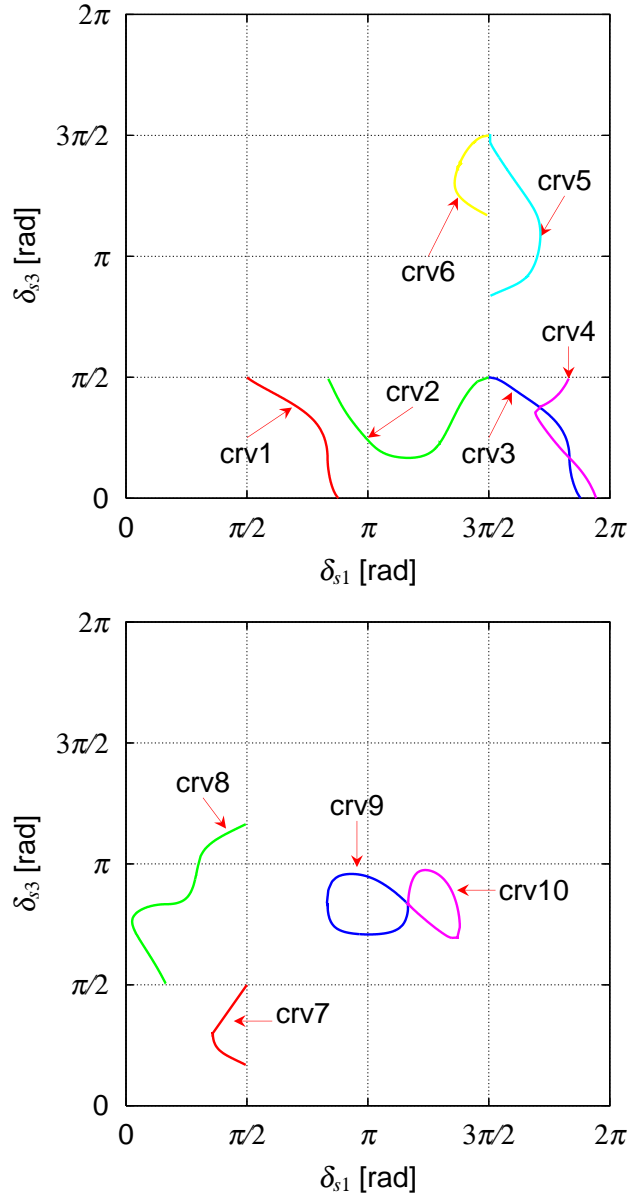


Figure 8: Numbering of the boundary curves.

The reason why Eq. (38a) is used to express curve-10 instead of Eq. (37a) is that Eq. (37a) cannot precisely express curve-10, but Eq. (38a) can. The parameters of the curves resulting from the Curve Fitting Toolbox are listed in Table 1.

4.2. Data size comparison

In this subsection, the data size required for allocation of the above fitted curves is compared to those in [20] and [21].

In [20], the physical space covering the CMG momentum envelope is divided into 33^3 segments. Each segment has a data record of the angular momentum vector and the normal vector of the singular surface nearest to the corresponding angular momentum. The data size required for recording each angular momentum vector and normal vector is 12 words (6×2 words), that is, 384 bits, where the data size of each parameter is assumed to be two words, that is, “double precision” in 32-bit C language. Therefore, the total data size required for recording the singular surfaces is 13,799,808 bits.

In [21], the physical space covering the CMG momentum envelope is divided into $34^2 \times 51$ grids. Each grid contains position data and an index to represent whether or not the corresponding grid is a singularity. Because the data size required for recording one grid is six words for the position data and one word for the singularity index, the total data size required is approximately 12,947,200 bits.

Regarding the method proposed in this paper, for simplicity, it is assumed that one curve needs 16 parameters. Because 10 kinds of curves are recorded in this study, a total of eight curves of the same shape are needed to represent all of the boundaries of the pyramid-type four CMG system, and the total required data size is approximately 81,920 bits.

From the above comparison, it is confirmed that the data size for the proposed method is much smaller than those of [20] and [21]. Thus, the proposed method is quite efficient at recording the singular surface properties.

5. Application of Fitted Curves

In this section, two applications of the fitted curves will be presented. They are “testing of singular type detection” and “determination of concrete gimbal angle perturbation for elliptic singularity avoidance.”

5.1. Testing of singularity type detection

A singularity type detectable matrix \mathcal{M} (Eq. (28)) is typically used for singularity type detection. As mentioned earlier, if \mathcal{M} is sign-definite, then the corresponding singularity is impassable, whereas if \mathcal{M} is indefinite, the corresponding singularity is passable.

Instead of using \mathcal{M} as explained above, in this paper, singularity type detection is performed by checking whether a set of singular gimbal angles $(\delta_{s1}, \delta_{s3})$ is located inside or outside of the impassable regions. In other words, the singularity type (hyperbolic/elliptic) can be determined using the fitted curves without execution of SVD. The functions of the fitted curves, which correspond to parabolic singularities, are rewritten in the form of an implicit function as

$$P(\delta_{s1}, \delta_{s3}) = 0 \quad (39)$$

Because several curves exist in one sub-domain, it is necessary to calculate the values of the functions corresponding to all the curves in the concerned sub-domain. For the purpose of singularity type detection, the sign of the function is appropriately set so that the multiplication of the function values is positive for the elliptic (impassable) singularities, whereas it is negative for the hyperbolic (passable) singularities in the concerned sub-domain.

Fig. 9 depicts a schematic illustration of the sign of the functions, in which the regions with a plus sign correspond to elliptic (impassable) singularities, whereas the regions with a minus sign correspond to hyperbolic (passable) singularities. In other words, the sign of the multiplication of the functions corresponds to the sign of \mathcal{M} .

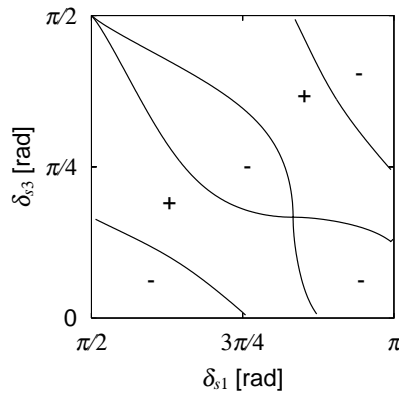


Figure 9: Signs of regions for singularity type detection.

5.2. Determination of concrete gimbal angle perturbation for elliptic singularity avoidance

In the local gradient methods, the direction of the gimbal angle perturbation is calculated so as to decrease the condition number of the Jacobian matrix \mathbf{C} , that is, so that CMG controllability is maintained to generate the desired/commanded torques. However, the local gradient methods do not determine concrete gimbal angle perturbation for elliptic singularity avoidance. If the concrete gimbal angle perturbation can be determined in a real-time calculation, then a model predictive elliptic singularity avoidance steering law will be achieved.

In order to determine the concrete gimbal angle perturbation, the local gradient methods could still be used. However, the computational cost would be high because the local gradient methods are based on SVD, which uses an iterative calculation. In this section, it will be explained how to overcome this problem by utilizing the fitted curve functions. Hereinafter, it is assumed that the gimbal angles are in an elliptic singular state.

There are two candidate methods. One of them is where the concrete gimbal angle perturbation is determined as an intersection point between a line along the local gradient direction and the fitted curves. The other method is where the concrete gimbal angle perturbation is determined by minimizing the gimbal angle perturbation subject to the fitted curves. The former method is hereinafter referred to as the “local gradient-based method,” and the latter is referred to as the “minimization-based method” in this paper.

5.2.1. Local gradient-based method

The local gradient-based method consists of the following two steps. Firstly, the local gradient direction is determined using SVD as the direction of the gimbal angle perturbation. Secondly, the intersection point (δ_1, δ_3) between the line with the local gradient direction and the fitted curves is calculated in the concerned sub-domain. The calculation of the intersection point can be performed using the Newton-Raphson method or the bisection method. There are two cases: intersection with the open-loop curves, and intersection with the close-loop curves. Each case is explained below.

Let $(\Delta\delta_1, \Delta\delta_3)$ denote the local gradient direction for the gimbal angle perturbation. A line starting from the current gimbal angles $(\delta_{c1}, \delta_{c3})$ and having a local gradient direction can be expressed using the parameter p as

$$\begin{bmatrix} u_a \\ v_a \end{bmatrix} = \begin{bmatrix} \delta_1 \\ \delta_3 \end{bmatrix} = \begin{bmatrix} \delta_{c1} \\ \delta_{c3} \end{bmatrix} + p \begin{bmatrix} \Delta\delta_1 \\ \Delta\delta_3 \end{bmatrix} \quad (40)$$

When the concerned sub-domain does not contain the close-loop curves, the intersection point is obtained by substituting Eq. (40) into Eq. (36) and solving it with respect to p using the Newton-Raphson method, the bisection method, or the Jenkins-Traub method. When the Jenkins-Traub method is used, more than one solution (including complex number solutions) could be obtained. However, because only one real solution that is the nearest to the current set of gimbal angles $(\delta_1, \delta_2, \delta_3, \delta_4)$ is necessary, such a solution is chosen.

On the other hand, when the concerned sub-domain contains close-loop curves such as curve-9 and curve-10, the condition for the intersection point (u_a, v_a) is given by

$$\begin{bmatrix} u_a \\ v_a \end{bmatrix} = \begin{bmatrix} \delta_{c1} \\ \delta_{c3} \end{bmatrix} + p \begin{bmatrix} \Delta\delta_1 \\ \Delta\delta_3 \end{bmatrix} = \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} + \begin{bmatrix} r(\theta) \cos \theta \\ r(\theta) \sin \theta \end{bmatrix} \quad (41)$$

By eliminating the parameter p from Eq. (41), the condition can be rewritten as

$$(\Delta\delta_1 \sin \theta - \Delta\delta_3 \cos \theta) r(\theta) + \Delta\delta_1(v_0 - \delta_{c3}) - \Delta\delta_3(u_0 - \delta_{c1}) = 0 \quad (42)$$

By substituting Eq. (37b) or Eq. (38c) into Eq. (42) and solving with respect to θ , the intersection point $(u_a, v_a)(= (\delta_{s1}, \delta_{s3}))$ is obtained. Fig. 10(a) and (b) shows a schematic image of the intersection point calculation with open-loop curves and close-loop curves, respectively. It should be noted that if the concerned sub-domain corresponds to the 4H singularity, that is, no curves exit in the sub-domain, it is unnecessary to carry out the above calculation because there is no intersection point in the sub-domain.

If the traditional local gradient method is used to determine the concrete gimbal angle perturbation, multiple executions of SVD are needed. On the other hand, the above process requires only one execution of SVD that performs iterative calculations. From this point of view, the method presented in this paper is more effective at calculating a concrete gimbal angle perturbation for elliptic singularity avoidance compared to the SVD-based methods.

5.2.2. Minimization-based method

Contrary to the previous method in which the gimbal angle perturbation is parallel to the local gradient with respect to (δ_1, δ_3) , in the minimization-based method, the gimbal angle perturbation is determined so as to minimize the \mathcal{L}_2 norm of the gimbal angle perturbation. Let us denote the difference vector from the current set of gimbal angles to the solution set of gimbal angles as

$$\mathbf{\Delta\delta} := (\Delta\delta_1, \Delta\delta_2, \Delta\delta_3, \Delta\delta_4)^T = (\delta_{s1} - \delta_{c1}, \delta_{s2} - \delta_{c2}, \delta_{s3} - \delta_{c3}, \delta_{s4} - \delta_{c4})^T \quad (43)$$

The problem in finding the minimum gimbal angle perturbation is formulated as

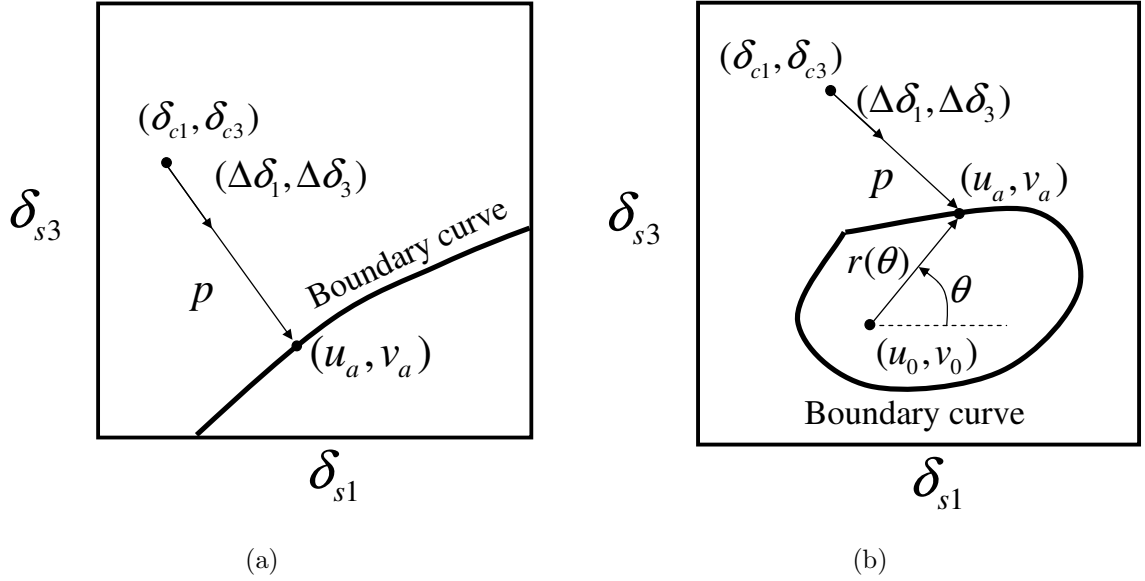


Figure 10: Schematic representation of intersection point calculation with (a) open-loop curves and (b) close-loop curves.

“find $\delta_A = (\delta_{s1}, \delta_{s2}, \delta_{s3}, \delta_{s4})^T$
such that minimizing $J := \|\Delta\delta\|^2 = \Delta\delta_1^2 + \Delta\delta_2^2 + \Delta\delta_3^2 + \Delta\delta_4^2$
subject to (Eq. (14) or Eq. (16)), (Eq. (15) or Eq. (17)), and Eq. (43).”

Although it seems that the above problem contains four parameters to be optimized, because gimbal angles (δ_2, δ_4) are variables depending on gimbal angles (δ_1, δ_3) in a singular state, and gimbal angles δ_1 and δ_3 are dependent on each other (which is represented in the form of boundary curves between the elliptic and hyperbolic singularities), there is only one parameter to be optimized in the above optimization problem: δ_1 or δ_3 .

5.2.3. Numerical examples

To validate the above two methods, numerical case studies will be presented here. Eight points (δ_1, δ_3) are chosen as points to check. The points are listed in Table 2, in which gimbal angle perturbations resulting from the above two methods are also listed. Figs. 11(a) and (b) show the eight points chosen and the gimbal angle perturbation resulting from the local gradient-based method and the minimization-based method. It is shown in both figures that the gimbal angle perturbation for elliptic singularity avoidance is successfully obtained. It should be noted that some results of the local gradient-based method are quite similar to those of the minimization-based method, but some results are different. This

is because the gimbal angle perturbations resulting from the local gradient-based method are determined based on the local gradient, which is not always globally optimal from the viewpoint of minimum gimbal angle perturbation for the avoidance of elliptic singularities. In fact, it is known that the CMG angular momentum controlled by the local gradient-based method has a tendency to slide along elliptic singular surfaces, which exist in a narrow tube-like shape inside of the trumpet-like shape, and cannot avoid elliptic singularities. By recalling this fact and looking into Fig. 11(a) and (b), it is found that some of the gimbal angle perturbations determined by the local gradient-based method have a tendency along the main existing region of elliptic singularities. From the viewpoint of gimbal angle perturbations, it is confirmed again that the local gradient-based steering control laws are undesirable for elliptic singularity avoidance. On the other hand, the gimbal angle perturbations resulting from the minimization-based method do not slide along the elliptic singularity existing region and take a shortcut to the boundary curves between the elliptic and hyperbolic singularities. These results show that the minimization-based method is desirable for avoiding/escaping the elliptic singularities. From this point of view, it is of interest to develop a model predictive steering control law based on the minimization-based method using the fitted boundary curves presented in this paper for the elliptic singularity avoidance.

6. Conclusions

The boundaries between hyperbolic (passable) and elliptic (impassable) singularities are calculated using the geometrical intersection tracking method and are recorded in the form of a polynomial function using curve fitting techniques. The fitted curves of the boundaries can be used to detect the type of singularity (i.e., hyperbolic/elliptic) and to explicitly determine the gimbal angle perturbation to avoid elliptic singularities. The amount of recorded singularity data is discussed and compared to those of the previous studies, in which the data were recorded at mesh points, and the simplicity of the boundaries expressed in the form of polynomial functions is demonstrated by several examples for the determination of gimbal angle perturbation to avoid elliptic singularities. The future work is to develop a model predictive singularity avoidance steering control law for pyramid-type SGCMGs based on the results in this paper and to experimentally validate its effectiveness.

Acknowledgment

This study was partially supported by a Grant-in-Aid for Scientific Research (C) (No. 22560783) from the Japan Society for the Promotion of Science.

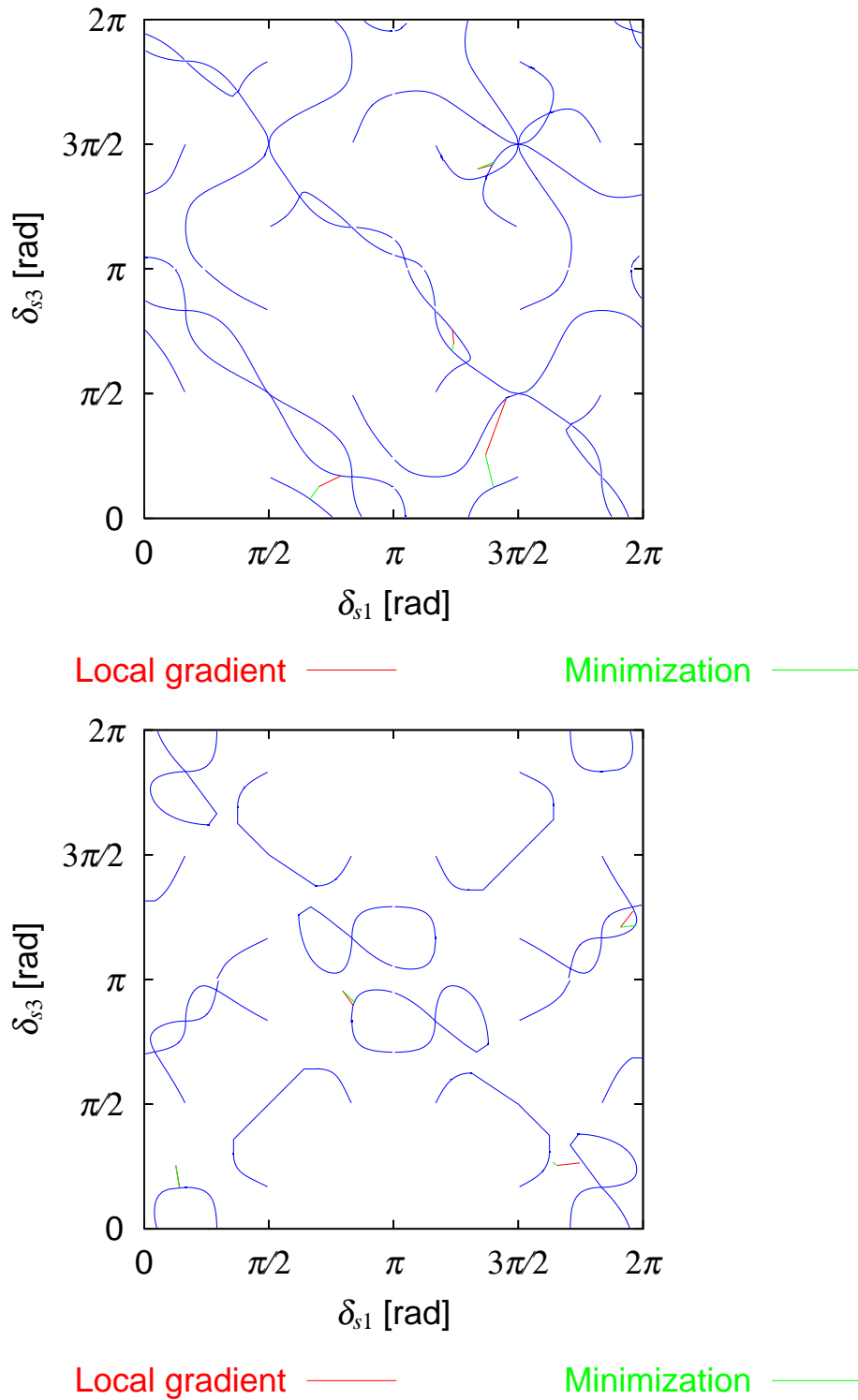


Figure 11: Gimbal angle perturbations resulting from the local gradient-based method (red) and the minimization-based method (green) corresponding to eight chosen points.

References

- [1] G. Margulies, J.N. Aubrun, Geometric theory of single-gimbal control moment gyro system, *Journal of Astronautical Sciences*, 26(1978) 159-191.
- [2] B. Wie, *Space vehicle dynamics and control*, 2nd edition, Reston, AIAA Education Series, American Institute of Aeronautics and Astronautics, Inc. 2008.
- [3] H. Kurokawa, A geometric study of single gimbal control moment gyros -singularity problems and steering law, Tech. Rep. Report, Mechanical Engineering Laboratory, Japan, No.175, Jan., 1998.
- [4] S.R. Vadali, H.S. Oh, S.R. Walker, Preferred gimbal angles for single gimbal control moment gyros, *Journal of Guidance, Control, and Dynamics*, 13(1990) 1090-1095.
<http://dx.doi.org/10.2514/3.20583>
- [5] J.A. Paradiso, Global steering of single gimbaled control moment gyroscopes using a directed search, *Journal of Guidance, Control and Dynamics*, 15(1992) 1236-1244.
<http://dx.doi.org/10.2514/3.20974>
- [6] A. Fleming, I.M. Ross, Singularity-free optimal steering of control moment gyros, *Advances in the Astronautical Sciences*, 123-III (2006) 2681-2700.
- [7] N.S. Bedrossian, J. Paradiso, E.V. Bergmann, D. Rowell, Steering law design for redundant single-gimbal control moment gyroscopes, *Journal of Guidance, Control, and Dynamics*, 13 (1990) 1083-1089.
<http://dx.doi.org/10.2514/3.20582>
- [8] H.S. Oh, S.R. Vadali, Feedback control and steering laws for spacecraft using single gimbal control moment gyros, *Journal of the Astronautical Sciences*, 39(1991) 183-203.
- [9] B. Wie, D. Bailey, C.J. Heiberg, Singularity robust steering logic for redundant single-gimbal control moment gyros, *Journal of Guidance, Control, and Dynamics*, 24(2001) 865-872.
<http://dx.doi.org/10.2514/2.4799>
- [10] B. Wie, Singularity escape/avoidance steering logic for control moment gyro systems, *Journal of Guidance, Control, and Dynamics*, 28(2005) 948-956.
<http://dx.doi.org/10.2514/1.10136>

- [11] K.A. Ford, C.D. Hall, Singular direction avoidance steering for control-moment gyros, *Journal of Guidance, Control, and Dynamics*, 23(2000) 648-656.
<http://dx.doi.org/10.2514/2.4610>
- [12] N.S. Bedrossian, J. Paradiso, E.V. Bergmann, D. Rowell, Redundant single gimbal control moment gyroscope singularity analysis, *Journal of Guidance, Control, and Dynamics*, 13(1990) 1096-1101.
<http://dx.doi.org/10.2514/3.20584>
- [13] H. Kurokawa, Constrained steering law of pyramid-type control moment gyros and ground tests, *Journal of Guidance, Control, and Dynamics*, 20(1997) 445-449.
<http://dx.doi.org/10.2514/2.4095>
- [14] H. Kurokawa, Survey of theory and steering laws of single-gimbal control moment gyros, *Journal of Guidance, Control and Dynamics*, 30(2007) 1331-1340.
<http://dx.doi.org/10.2514/1.27316>
- [15] A.F. Leve, G.N. Fitz-Coy, Hybrid steering logic for single-gimbal control moment gyroscopes, *Journal of Guidance, Control, and Dynamics*, 33(2010) 1202-1212.
<http://dx.doi.org/10.2514/1.46853>
- [16] H. Schaub, J.L. Junkins, Singularity avoidance using null motion and variable-speed control moment gyros, *Journal of Guidance Control, and Dynamics*, 23(2000) 11-16.
<http://dx.doi.org/10.2514/2.4514>
- [17] J. McMahon, H. Schaub, Simplified singularity avoidance using variable-speed control moment gyroscope null motion, *Journal of Guidance, Control, and Dynamics*, 32(2009) 1938-1943.
<http://dx.doi.org/10.2514/1.45433>
- [18] K. Takada, H. Kojima, Receding horizon control on steering of control moment gyro for fast attitude maneuver, *Transactions of the Japan Society for Aeronautical and Space Sciences*, 52(2009) 1-10.
<http://dx.doi.org/10.2322/tjsass.52.1>
- [19] H. Kojima, N. Matsuda, K. Takada, Adaptive skewing pyramid-type CMGs for fast attitude maneuver, *Transactions of the Japan Society for Aeronautical and Space Sciences, Space Technology Japan*, 7(2009) 19-24.
<http://dx.doi.org/10.2322/tstj.7.19>

- [20] K. Takada, H. Kojima, N. Matsuda, Control moment gyro singularity-avoidance steering control based on singular-surface cost function, *Journal of Guidance, Control, and Dynamics*, 33(2010) 1442-1450.
<http://dx.doi.org/10.2514/1.48381>
- [21] S. Sato, M. Takahashi, Singularity escape of satellite with control moment gyros using path planning of angular momentum, *Transaction of Japan Mechanical Engineering (C)*, 79(2013) 1526-1539 (in Japanese).
<http://dx.doi.org/10.1299/kikaic.79.1526>
- [22] E.P. Hart, J.N. Nilson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths, *IEEE Transactions on Systems Science and Cybernetics Transactions on Systems Science and Cybernetics*, Vol. 4, No. 2 (1968) 100-107.
- [23] B. Wie, Singularity analysis and visualization for single-gimbal control moment gyro systems, *Journal of Guidance, Control, and Dynamics*, 27(2004) 271-282.
<http://dx.doi.org/10.2514/1.9167>
- [24] H. Kojima, Singularity analysis and steering control law for adaptive-skew pyramid-type CMG, *Acta Astronautica*, 85(2013), 120-137.
<http://dx.doi.org/10.1016/j.actaastro.2012.12.019>
- [25] M. Higashi, T. Mori, M. Hosaka, High-quality intersection calculation of surfaces by geometrical tracing of intersections, *Japan Society of Precision Engineering*, 56(1990), 92-97 (in Japanese).
<http://dx.doi.org/10.2493/jjspe.56.92>
- [26] D. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *SIAM Journal on Applied Mathematics*, 11 (2), 431-441. (1963). <http://dx.doi.org/10.1137/0111030>.

Table 1: Parameters of fitted curves.

	crv-1	crv-2	crv-3	crv-4	crv-5
a_0	2.235	0.5244	5.743	5.714	5.344
a_1	0.6195	-5.184E-4	-0.1106	-0.4869	-0.2031
a_2	-0.2132	0.1740	-0.2227	-0.1073	-0.2646
a_3	-0.7287	7.781E-2	-0.1974	-0.1558	7.781E-3
a_4	0.4016	0.7683	6.195E-2	0.1721	8.100E-2
a_5	0.3995	0.2636	-3.526E-2	0.4691	2.184E-2
a_6	-0.3206	-0.5504	-4.303E-2	-2.967E-2	-2.386E-2
a_7	-7.436E-2	-0.2693	8.767E-2	-0.2206	
a_8	0.1119	0.1175	2.158E-2	-2.006E-3	
a_9	-2.425E-2	6.409E-2	-3.081E-2	3.149E-2	
a_{10}					
a_{11}					
\bar{x}	-1.467	3.666	0.6736	0.5829	3.643
σ	0.7510	0.6506	0.5773	0.5738	0.6013
	crv-6	crv-7	crv-8	crv-9	crv-10
a_0	4.281	1.126	1.374	0.2885	6.556E-2
a_1	7.466E-2	3.747E-3	0.1176	4.861E-2	-0.2029
a_2	7.494E-2	7.915E-3	0.7090	0.3220	0.3003
a_3	-8.389E-2	-5.988E-4	-0.2791	-0.1155	0.1105
a_4	6.979E-2	2.306E-5	-0.2425	-0.4150	1.059
a_5	4.734E-2		-0.1240	0.1350	8.401E-2
a_6	-3.383E-2		-0.3001	0.2958	0.9908
a_7	-0.2417		0.3657	-1.343E-2	0.3617
a_8	6.599E-3		0.1061	-6.286E-2	0.9161
a_9	5.047E-3		-0.1060	-9.456E-3	0.1262
a_{10}					0.9494
a_{11}					0.2050
\bar{x}	4.186	0.9327	2.530	3.233	1.888
σ	0.3140	6.524E-2	0.7427	1.929	0.6132
u_0				2.900	4.300
v_0				2.645	2.180

Table 2: Singular gimbal angles and perturbed gimbal angles resulting from the local gradient-based method and the minimization-based method.

case index	gimbal angles ($\delta_1, \delta_2, \delta_3, \delta_4$)	local gradient-based method ($\delta_1, \delta_2, \delta_3, \delta_4$)	minimization-based method ($\delta_1, \delta_2, \delta_3, \delta_4$)
1	2.2000, 4.2033, 0.4000, 2.7700	2.4848, 4.5934, 0.5381, 2.5262	2.0887, 4.0208, 0.2416, 2.8282
2	4.3000, 0.0177, 0.8000, 5.8857	4.5589, 6.0406, 1.5002, 0.1807	4.3987, 0.2465, 0.3867, 5.7184
3	3.9000, 2.1352, 2.2000, 4.5030	3.8842, 1.6986, 2.3702, 4.6814	3.8760, 2.3235, 2.1104, 4.3701
4	4.2000, 5.9982, 4.4000, 0.0195	4.3864, 6.1171, 4.4518, 6.2520	4.4102, 6.1066, 4.4892, 0.0033
5	0.4000, 8.8139, 0.8000, 6.0688	0.4483, 8.8261, 0.5254, 5.7616	0.4425, 8.8203, 0.5249, 5.7610
6	5.2000, 7.5046, 0.8000, 2.2707	5.4866, 7.8923, 0.8307, 1.4411	5.1455, 7.4677, 0.8422, 2.2741
7	6.0000, 8.2142, 3.8000, 6.1758	6.1571, 8.4882, 4.0036, 0.0767	6.2016, 8.4065, 3.8227, 6.0782
8	2.5000, 7.1347, 3.0000, 3.4846	2.6395, 7.0029, 2.8089, 3.6921	2.6549, 7.0475, 2.8477, 3.7140