Retrospective Cost Adaptive Spacecraft Attitude Control using Control Moment Gyros

Marc Camblor, Gerardo Cruz, Sergio Esteban, Fred A. Leve, and Dennis S. Bernstein

Abstract—We present a numerical investigation of the performance of retrospective cost adaptive control (RCAC) for spacecraft attitude control using control-moment-gyroscopes (CMG). The setup consists of three orthogonally mounted CMG’s that are velocity commanded without the use of a steering law. RCAC is applied in a decentralized architecture; each CMG is commanded by an independent RCAC control law. This architecture simplifies the required modeling information and treats the axis-coupling effects as unmodeled disturbances. A rotation-matrix parameterization of attitude is used to implement a dynamic compensator with state feedback. The adaptive controller is able to complete various slew and spin maneuvers using limited information about the mass-distribution of the spacecraft.

I. INTRODUCTION

The traditional approach to spacecraft attitude control using CMG’s is to implement a controller that specifies a desired torque. This torque is then realized by a steering law that commands the angular velocities of each CMG gimbal [1]. A difficulty of this approach is the fact that the CMG’s may reach a singular configuration in which the torques that can be produced are confined to a plane. The singularity prevents the use of the matrix inverse required by many steering laws [2]. Therefore, various methods for constructing singularity-avoiding steering laws have been developed, [1], [3]–[5]. Singularities can also be avoided by implementing CMG’s with variable-speed wheels at the cost of additional mechanical complexity [6].

In the present paper we consider spacecraft attitude control with CMG’s using an adaptive control law that directly commands the gimbal velocities. Previous adaptive methods for single-gimbal CMG’s include singularity avoiding steering laws as part of the control synthesis [7]. Other adaptive methods have focused on variable speed [8], or double gimbaled CMG’s [9]. Instead of using steering laws and pseudo-inverses to deal with singularities we apply retrospective cost adaptive control. This adaptive controller requires minimal modeling of the spacecraft inertia for thruster [10], reaction-wheel [11], and magnetic-torque [12] actuation.

The primary objective of this paper is to assess the performance and robustness of RCAC in the presence of time-varying singularities in the input matrix for velocity-commanded CMG’s. The setup consists of three orthogonally mounted, single-gimbal, fixed-speed CMG’s in a decentralized architecture; each of the CMG’s has an independent controller. This architecture simplifies the required modeling information, leaves axis-coupling effects to be compensated by RCAC, and avoids the need for matrix inversion.

Two classes of maneuvers are considered, namely, motion-to-rest maneuvers, where the goal is to bring the spacecraft attitude to rest with a specified inertial orientation, and motion-to-spin maneuvers, where the goal is to bring the spacecraft to a spin with a given rate about a specified body-fixed axis with a specified inertial attitude.

II. SPACECRAFT MODEL WITH CMGS

We develop the dynamics for a rigid spacecraft actuated by three single-gimbal, fixed-speed CMGs. We begin by defining several frames: an inertial frame F, a body frame B, fixed to the spacecraft bus, a frame Fc, attached to each gimbal, and a frame Fi, fixed to each wheel. We specify the kinematics using Poisson’s equation,

\[ \dot{R}_{B/I} = R_{B/I} \times \dot{\omega}_{B/I}, \]

where \( \dot{R}_{B/I} \) denotes a derivative with respect to F. The proper orthogonal matrix \( R_{B/I} \) is a rotation matrix that transforms the components of a vector resolved in frame B into the components of the same vector resolved in frame I, \( \dot{\omega}_{B/I} \) is the angular velocity of the body frame with respect to the inertial frame, and the superscript \( \times \) denotes the skew-symmetric cross product matrix.

Consider a spacecraft composed of a bus b and control moment gyros. We can describe the spacecraft’s dynamics using Euler’s equation,

\[ \dot{H}_{sc/b/c} = \omega_{B/I} \times H_{sc/b/c} + \tau_d, \]

where \( \dot{J}_{b/c} \) is the inertia tensor of the bus relative to the spacecraft’s center of mass c, \( \tau_d \) is the sum of external disturbance torques, and \( n_Q \) is the number of CMG’s. The angular momentum of gyro wheel i relative to c with respect to F is given by

\[ \dot{H}_{w_i/c/1} = \dot{J}_{w_i/c} \omega_{B/I} + \dot{J}_{w_i/c} \omega_{W_i/B}, \]

where the inertia of gyro wheel i relative to the spacecraft’s center of mass is \( \dot{J}_{w_i/c} \) and \( \dot{J}_{w_i/c} \) is the inertia tensor of gyro wheel i relative to its respective center of mass c. The angular velocity of gyro wheel i relative to F is given by

\[ \dot{\omega}_{W_i/B} = \dot{\omega}_{W_i/G_i} + \dot{\omega}_{G_i/B}, \]

1University of Michigan, Ann Arbor, MI
2Universidad de Sevilla, Sevilla, Spain
3U.S. Air Force Research Laboratory, Albuquerque, NM
where $\vec{\omega}_{W_i/G_i}$ is the angular velocity of gyro wheel $i$ relative to gimbal $i$, and $\vec{\omega}_{G_i/B}$ is the angular velocity of gimbal $i$ with respect to $F_B$.

We assume that the gimbal wheels are spherical with moment of inertia $\beta$, 

$\vec{J}_{W_i/c_i} = \beta \vec{I}$, 

where $\vec{I}$ is the physical identity matrix. This assumption results in a time-invariant spacecraft inertia and sidesteps a common assumption in the CMG literature in which the gyro wheels are assumed to be much smaller than the bus. Furthermore, we assume that each wheel spins about its axis at a constant angular rate that is identical for all CMGs.

The above assumptions combined with (2), (3), and (4), yield the dynamic equation for a spacecraft with spherical, constant-speed CMG's,

$\vec{J}_{sc/c} \vec{\omega}_{B/1} = \left[ \vec{J}_{sc/c} \vec{\omega}_{B/1} + \beta \sum_{i=1}^{n_G} \vec{\omega}_{W_i/G_i} \right] \times \vec{\omega}_{B/1} + \beta \sum_{i=1}^{n_G} \vec{\omega}_{W_i/G_i} + \vec{\omega}_{G_i/B} + \tau_d$. (5)

where $\vec{J}_{sc/c}$ is the combined inertia of the bus and the CMG's relative to c. A more detailed derivation is given in [13].

To formulate the control problem we resolve the kinematics and dynamics in the body frame. To resolve (5) in $F_B$ we first resolve $\vec{\omega}_{W_i/G_i}$ and $\vec{\omega}_{G_i/B}$ in their respective gyro frames, that is,

$\vec{\omega}_{W_i/G_i} |_{G_i} = e_1 q_i, \quad \vec{\omega}_{G_i/B} |_{G_i} = e_2 u_i$

where $e_i$ is the $i$th column of the identity matrix, $u_i$ is the scalar commanded angular rate of gimbal $i$, and $q > 0$ is the angular rate of each wheel. Thus, in the body frame,

$\vec{\omega}_{W_i/G_i} |_{B} = \mathcal{O}_i e_1 q_i, \quad \vec{\omega}_{G_i/B} |_{B} = \mathcal{O}_i e_2 u_i$

where $\mathcal{O}_i$ transforms vectors resolved in $F_{G_i}$ into $F_B$. We assume that $n_G = 3$ and that the gimbals are mounted orthogonally along the body axes such that

$\vec{\omega}_{G_i/B} |_{B} = e_i u_i, \quad \vec{\omega}_{G_i/B} |_{B} = e_i \omega_i$.

Note that, the control $u_i$ also affects the alignment of the gyro, that is,

$\dot{\mathcal{O}}_i = -u_i e_i^T \mathcal{O}_i$. (6)

Finally, we define the notation

$\hat{R} \triangleq \vec{R}_{B/1} |_{B}, \quad J_{sc} \triangleq \vec{J}_{sc/c} |_{B}, \quad \vec{\omega} \triangleq \vec{\omega}_{B/1} |_{B}, \quad \vec{\omega} \triangleq \vec{\omega}_{B/1} |_{B}, \quad \tau_d \triangleq \vec{\tau}_d |_{B}.$

Therefore, the kinematics in (1) are given by,

$\hat{R} = R \dot{\omega}^x,$

and the dynamics in (5) are

$J_{sc} \dot{\omega} = \left[ J_{sc} \omega + \beta \sum_{i=1}^{n_G} \Omega_i e_1 q \right] \times \omega + B_{sc} u - \beta \dot{u} + \tau_d$, (8)

where

$u = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T$,

and the $i$th column of $B_{sc}$ is given by

$b_i \triangleq \beta (\Omega_i e_1 q - \omega) \times e_i$. (9)

Note that $B_{sc}$ is implicitly time-varying due to its dependence on the states. Furthermore, in practice $q >> ||\omega||$. Therefore, the input matrix is dominated by the gimbal angles embedded in the orientation matrices $\mathcal{O}_i$:

$B_{sc} \approx q \beta \left[ (\Omega_1 e_1)^x e_1 \ (\Omega_2 e_1)^x e_2 \ (\Omega_3 e_1)^x e_3 \right]$. (10)

III. RCAC ALGORITHM

We review retrospective cost adaptive control, a more detailed development can be found in [14]. Consider a MIMO discrete-time system

$x(k) = Ax(k - 1) + Bu(k - 1), \quad (11)$

$z(k) = Cx(k) - r(k)$, (12)

where $x(k) \in \mathbb{R}^n, u(k) \in \mathbb{R}^l, z(k) \in \mathbb{R}^l$ is the performance variable, $r(k) \in \mathbb{R}^l$ is the command, the matrices $A, B, C$ have appropriate dimensions, and $k \geq 1$.

RCAC is a direct adaptive controller that minimizes the performance variable $z$. The controller uses the Markov parameters of the transfer function $u$ to $z$ as modeling parameters. The $i$th Markov parameter is given by

$H_i = E_i A^{i-1} B, i \geq 1$. (13)

We define the controller $u(k)$ by the strictly proper time-series,

$u(k) = \sum_{i=1}^{n_c} M_i u(k - i) + \sum_{i=1}^{n_c} N_i z(k - i) = \theta(k) \phi(k)$, (14)

where

$\theta(k) \triangleq [M_1(k) \cdots M_{n_c}(k) \ N_1(k) \cdots N_{n_c}(k)] \in \mathbb{R}^{l_u \times n_c(l_u + l_z)}$, (15)

for $M_i(k) \in \mathbb{R}^{l_u \times l_u}$, $N_i(k) \in \mathbb{R}^{l_z \times l_z}$, and

$\phi(k) \triangleq \begin{bmatrix} u(k - n_c) \\ \vdots \\ u(k - 1) \\ z(k - 1) \\ \vdots \\ z(k - n_c) \end{bmatrix} \in \mathbb{R}^{n_c(l_u + l_z)}$. (16)
The controller parameter $\theta(k)$ is updated using the Markov parameters to compute the retrospective control,
\[ \hat{u}(k-1) \triangleq \begin{bmatrix} H_1^T H_1 + \eta(k) I(k) \end{bmatrix}^{-1} H_1^T \left[ H_1 u(k-1) - z(k) \right]. \tag{17} \]

The retrospective control is then used in a recursive least squares update given by,
\[ \theta_T(k) \triangleq \theta_T(k-1) + \frac{P(k-1) \phi(k-2) \left[ \theta(k-1) + \hat{u}(k-1) \right]^T}{[1 + \phi^T(k-2) P(k-1) \phi(k-2)]}, \tag{18} \]
where the regressor matrix $P(k)$ is updated by
\[ P(k) \triangleq P(k-1) - \frac{P(k-1) \phi(k-2) \phi^T(k-2) P(k-1)}{[1 + \phi^T(k-2) P(k-1) \phi(k-2)]}. \tag{19} \]

IV. ATTITUDE CONTROL WITH RCAC USING CMG’S

The controller parameter $\theta(k)$ is updated using the Markov parameters to compute the retrospective control and defined even in singular CMG configurations,
\[ \hat{u}_i(k-1) = H_i^T \left[ H_i u_i(k-1) - z(k) \right] \]
\[ H_i^T H_i + \eta(k). \]

Thus, we define the performance variable $z$ for the attitude control problem as
\[ z \triangleq \begin{bmatrix} \hat{\omega}^T & S^T_{od} & S^T_d \end{bmatrix}^T \in \mathbb{R}^9, \tag{25} \]

Therefore, $z = 0$ corresponds to $\hat{R} = I$, $\hat{\omega} = 0$.

A. Markov Parameter

We use the linearized model described in [11] to obtain an inertia-free Markov parameter for attitude control,
\[ H \triangleq CB \approx \begin{bmatrix} h B_{sc} \\ \frac{1}{2} h^2 M_a R_a M_a^T B_{sc} \\ \frac{1}{4} h^2 M_a M_a^T B_{sc} \end{bmatrix} \in \mathbb{R}^{9 \times 3}. \tag{26} \]

Where $h$ is the controller time-step and
\[ M_a \triangleq \begin{bmatrix} e_1^2 & e_2^2 & e_3^2 \end{bmatrix}, \quad R_a \triangleq A_{att} \otimes I_3. \]

Note that the Markov parameter corresponding to $S_d$ was chosen heuristically.

B. Singularities and Decentralized Controlled

![Block diagram for decentralized control of CMGs using RCAC.](image)

For the spacecraft described by (8), CMG singularities present themselves as a rank deficiency in the actuator matrix $B_{sc}$ in (10). Singular CMG configurations result in a rank deficient $H$ in (26) which may prevent the computation of (17). Therefore, we implement three multi-input single-output controllers; each RCAC subcontroller commands the shaft angular velocity of one of the gimbals. However, unlike the approach in [12], each controller utilizes the same performance variable as shown in Figure 1. The modeling information used by each controller is given by the $i$th column of (26), and thus the Markov parameter for the $i$th RCAC controller is given by
\[ H_i^T \triangleq H e_i \in \mathbb{R}^9. \tag{27} \]

Thus, the retrospective controls in (17) are now scalar and defined even in singular CMG configurations,
\[ \hat{u}_i(k-1) = \frac{H_i^T \left[ H_i^T u_i(k-1) - z(k) \right]}{H_i^T h_i^T H_i^T + \eta(k)}. \]
V. NUMERICAL EXAMPLES

We consider two scenarios, namely, motion-to-rest (M2R) maneuvers and motion-to-spin (M2S) maneuvers, where rest and spin refer to motion relative to an inertial frame. A M2R maneuver may begin from rest or an arbitrary angular velocity. Thus, M2R includes slews, detumbling, and stabilization, the goal is to have the spacecraft come to rest with a specified attitude. M2S maneuvers require that the spacecraft spin about a specified body axis that is pointed in a specified inertial direction. If the M2R and M2S maneuvers begin from zero angular velocity, then we use the terminology rest-to-rest (R2R) and rest-to-spin (R2S), respectively.

A. Gimbal Angles and Singularities

The orientation matrix $O_i$ depends on the angle $\phi_i$ rotated by gimbal $i$ about its shaft. We define these angles as follows, $\phi_1 = 0$ when $\dot{\omega}_{W_1/G_B} = \epsilon_2 q$, $\phi_2 = 0$ when $\dot{\omega}_{W_2/G_B} = \epsilon_3 q$, and $\phi_3 = 0$ when $\dot{\omega}_{W_3/G_B} = \epsilon_1 q$. The input matrix $B_{sc}$ is singular when the configuration of gimbal angles $\phi_i$ cannot provide instantaneous torque about every direction. When $B_{sc}$ is singular, also known as gimbal lock, the instantaneous torque that the gimbals can provide is confined to a plane. If the required instantaneous torque lies in this plane, we use the term non-obstructing singularity, otherwise it is an obstructing singularity.

B. Baseline Spacecraft Parameters

The bus inertia matrix $J_b$ is given by

$$J_{sc} = \text{diag}(3.2894, 7.8994, 10.7112) \text{ kg-m}^2.$$  \hfill (28)

The gyro-wheel moment of inertia is given by

$$\beta = 0.001 \text{ kg-m}^2.$$  \hfill (29)

For all of the simulations, the initial attitude is given by $R_0 = I$ and the initial desired attitude, $R_d(0)$, is a rotation about the body-fixed direction $n = [1 \ 0 \ 0]^T$ with angle $\theta_d(0)$. The gyro-wheels are assumed to be spinning at a constant angular velocity $q = 600$ rad/sec, and the initial shaft’s angular velocity is $u_0 = [0 \ 0 \ 0]^T$ rad/sec. Furthermore, each gimbal’s angular velocity and angular acceleration are saturated at $u_{\text{max}} = 1$ rad/sec and $\dot{u}_{\text{max}} = 1$ rad/sec$^2$, respectively, this keeps the control input small and bounded regardless of the adaptation gains.

C. Motion-to-Rest Maneuvers

We command R2R and M2R maneuvers about principal axes using step and ramp commands. Figure 2 shows a M2R maneuver about a principal axis with a step command. The maneuver is initialized at an obstructing singularity as shown by the singular values of the matrix $B_{sc}$ in Figure 2(j) and more singularities are encountered along the maneuver. The spacecraft is able to avoid these singularities and achieve the desired attitude and angular velocity.

Next, Figure 3 shows a R2R maneuver about a principal axis with a ramp command. The maneuver is initialized at a non-obstructing singularity. The spacecraft initially deviates from the commanded ramp to avoid the singularity and returns to follow the command. Note that after 300 sec RCAC re-adapts as a response to the change in slope in the command.

Fig. 2. M2R for RCAC with the desired attitude $\theta_d(0) = 170$ deg and the initial angular velocity $\omega_0 = [-0.03 \ 0.02 \ -0.01]^T$ rad/sec. The controller weights are $R_2 = 2 \cdot 10^{-1} I$ and $R_3 = I$, the initial covariance is $P(0) = 10^2 I$, the initial controller parameter are $\theta(0) = 0$, and the controller order is $n_c = 4$. The initial gimbal angles are $\phi_1 = 0$ deg, $\phi_2 = 90$ deg, and $\phi_3 = 0$ deg which is an obstructing singularity.
The maneuver is initialized at rest and the spacecraft is able to follow the command.

\[
\omega_d(t) = 0.005 \begin{bmatrix} \sin(t) & \cos(\frac{\pi}{3}) & \sin(0.1t + \frac{\pi}{3}) \end{bmatrix}^T \text{ rad/sec.} \quad (31)
\]

The maneuver is initialized with arbitrary initial angles, and the
non-principal inertia is given by
\[
J_{sc} = \begin{bmatrix} 5.6096 & -1.0121 & 0.6143 \\ -1.0121 & 2.8718 & 0.3979 \\ 0.6143 & 0.3979 & 4.7187 \end{bmatrix} \text{ kg-m}^2. \quad (30)
\]

D. Motion-to-Spin Maneuvers

We command a step and a sinusoid desired angular velocity for motion-to-spin.

Figure 4 shows a R2S maneuver with a step command, where the final spin is about a non-principal axis. The maneuver is initialized with arbitrary initial angles, and the

Fig. 4. R2S for RCAC using the non-principal inertia matrix in (30) with the desired attitude \( \theta_d(0) = 30 \) deg, and the desired angular velocity 
\[
\omega_d = \begin{bmatrix} 0.02 & 0.03 & 0.04 \end{bmatrix}^T \text{ rad/sec.}
\]
The controller weights are \( R_k = I \) and \( R_u = 10^{-3}I \), the initial covariance is \( P(0) = 10^3 I \), the initial controller parameter is \( \theta(0) = 0 \), and the controller order is \( n_c = 4 \). The initial gimbal angles are \( \phi_1 = 0 \) deg, \( \phi_2 = 90 \) deg, and \( \phi_3 = 0 \) deg which is a non-obstructing singularity.
To study singularity avoidance, we command a rotation about a body-fixed direction and initialize the CMG’s shaft angles at a position for which a torque about the desired body-fixed direction cannot be initially commanded. When a step command is used, RCAC initially commands torque about some axis and then figures out how to achieve the desired attitude. When a ramp is commanded, RCAC deviates momentarily from the commanded rotation path to avoid the singularity and quickly comes back to follow the command. The error between the desired and commanded rotations is slight. In the case that no singularities appear, RCAC follows the command precisely. Furthermore, no saturated pseudo-inverse or similar methodologies to avoid singularities are used.

REFERENCES


VI. CONCLUSIONS

We apply RCAC to spacecraft attitude control with CMG’s actuation. The CMG’s are assumed to be mounted in a known and linearly independent gimbal-axis configuration with an arbitrary and unknown orientation relative to the spacecraft principal axes. A decentralized architecture with one RCAC for each channel is used. The RCAC algorithm is able to bring the spacecraft to a desired attitude and angular velocity for both M2R and M2S scenarios.

For R2R maneuvers, we use two different methods to achieve the desired attitude. First we specify the final desired attitude rotation matrix (that is, step command). Second we specify the rotational path as a function of time needed to achieve the desired attitude (that is, ramp command).