Control of Micro-satellite Orientation Using Bounded-input, Fully-reversed MEMS Actuators

Sung Key Koh*  James P. Ostrowski*  G.K. Ananthasuresh*

February 18, 2002

Abstract

We present a novel technique for controlling the attitude of micro-satellites using MEMS-based microactuators. In addition to being restricted in the magnitudes of inputs, most microactuators need to be returned to their original inactive state at the end of the attitude control maneuver in order to reduce energy consumption. This type of bounded and fully-reversed actuation on a free rigid body leads to a special type of system in which nonholonomic effects have been incorporated by design. We discuss issues of controllability and develop motion planning algorithms for such a system considering the nonintegrable nature of rigid body rotations. We use approximate methods to develop two algorithms for controlling satellite orientation. The first method uses a series of pair-wise actuation steps and is guaranteed to converge to the desired orientation; the second uses fewer steps, and in certain cases uses less input power. We also present preliminary results of a practical implementation using electro-thermal microactuation.

1 Introduction

The problem of rigid body reorientation has been studied in a wide variety of contexts in the fields of robotics and control. A very interesting sub-class of these problems deals with free rigid bodies where there are no external forces acting on the body. This includes the study of neutrally buoyant

*Department of Mechanical Engineering and Applied Mechanics, General Robotics, Automation, Sensing and Perception Laboratory, University of Pennsylvania, 3401 Walnut Street, Philadelphia, PA 19104-6228, USA, {skkoh,jpo,gksuresh}@seas.upenn.edu
underwater or aerial vehicles, as well as spacecraft. In this paper, we study the attitude control problem for miniature satellites and introduce a novel actuation technique and associated control algorithms.

A large body of literature has been devoted to studying the reorientation of satellites, generally using either thrusters or momentum wheels as the control inputs, though others have investigated utilizing redundancies in attached robotic arms or even variations in the Earth’s magnetic fields [18, 26]. Generally, the work has fallen into one of two classes: either the system is fully actuated, and techniques are studied to robustly or efficiently control attitude [2, 3, 4, 9, 28]; or the system is underactuated, i.e., some degrees of freedom are not directly actuated, and controllers are developed to achieve total control of all three degrees of freedom [15, 22]. For example, Bullo and Murray [4, 6] and Bharadwaj et al. [2] developed mechanisms that generalized traditional control techniques, e.g., PID control, to the non-Euclidean manifold, $SO(3)$, describing spatial rigid body rotations. In the area of underactuated attitude control, the primary focus has been on using two orthogonally mounted momentum wheels to generate rotations about all three body axes. By increasing or decreasing the speed of rotation of these wheels, reaction torques are created to make the spacecraft assume different orientations and or spin velocities. On the other hand, for systems starting with zero angular momentum, the wheels can be simply rotated to a different orientation and brought to rest. The conservation of angular momentum effectively implies a nonholonomic, or non-integrable, “constraint.” Using relatively standard tools from geometric nonlinear control theory, one can then show that the two inputs are enough to provide controllability (see, e.g., work by Walsh et al. and Leonard and Krishnaprasad [11, 19, 27]). That is, they allow one to fully control the attitude of the system, and give a mechanism for developing the sequence of controls to achieve this.

Our interest in the use of nonholonomic effects by design arises in the context of miniature spacecraft, or what are variously referred to as either micro-, nano-, or pico-satellites [7, 25]. When spacecraft are miniaturized, naturally the momentum wheels that have been predominantly studied in the literature should also be miniaturized. However, designs, microfabrication techniques, and packaging of rotary micro motors and bearings are not as well developed as linear, reciprocating ones. Furthermore, microactuators that can deliver large displacements and torques are usually based on

---

1Some authors have argued that velocity constraints introduced by conservation laws should be distinguished from those arising due to external constraints, but we will not worry about the semantics here, and simply refer to these conservation laws as implying nonholonomic constraints.
elastic deformation that is a consequence of one of many principles of transduction at the micro scale. Thus, these microactuators elastically deform upon activation and then return to their original state when they are deactivated. Since power is consumed as long as the actuators are active, it is not economical to use them for extended periods of time to effect changes in orientation. But, we do know that the non-integrable nature of the angular momentum conservation laws, coupled with the non-commutativity of finite rigid-body rotations, can result in net changes in orientation even though all individual actuators are brought to their initial inactive state. It is, therefore, an interesting prospect to achieve permanent change in the orientation of the free rigid-body using actuators capable of only fully-reversed, bounded motions.

The problem that we consider in this paper is the inverse kinematics-type control problem: “What sequence of rotations would give a desired net change in orientation of the body while the net rotation about any axis is zero?” We address many interesting issues related to this problem and present two approximate methods for solving it in the case of fully-reversed, bounded input motions. In Section 2, we describe the general problem and some novel issues that arise when using these types of inputs. In Section 3, we develop two motion planning schemes that can be used to solve the satellite orientation problem and discuss tradeoffs associated with each scheme. This is followed in Section 4 by descriptions of initial experimental results with a planar microactuator that confirm the plausibility of utilizing such actuation mechanisms. Finally, we conclude in Sections 5 and 6 with a discussion of further challenges and future work.

2 Problem statement

Many micro-actuation schemes exist [10] and many of them can be utilized for attitude control of miniature spacecraft provided that the associated practical problems of sufficient torque, speed of response, microfabrication, and packaging are suitably addressed. For example, Reiter et al. [24] recently proposed a MEMS-based rotary mechanism using electrostatic actuation.

2.1 Bounded, fully reversed actuation

A schematic for a potential mechanism for generating rotational motions is shown in Figure 1, where the radially projecting elastic beams have concentrated masses at the tips and can be actuated in unison to rotate the masses about the center in clockwise or counter-clockwise directions.
This is called a **pseudo-wheel** [1] because while the actuator system does not rotate upon actuation, it does cause the body on which it is mounted to rotate in the opposite direction. A specific embodiment of this using electro-thermal microactuation and called an **Electro-Thermal Compliant (ETC) pseudo-wheel** is presented in [13], and is briefly described in Section 4.

![Satellite sketch with three momentum wheels (front face deflected)](image)

**Figure 1: Satellite sketch with three momentum wheels (front face deflected)**

To simplify the analysis, we assume the miniature spacecraft to be a spherically symmetric rigid body with inertia matrix given by $J = \text{Diag}\{J_1, J_1, J_1\}$. We assume the pseudo-wheels mounted on each orthogonal body-axis can be modeled as simple, planar rigid bodies attached at the center of mass, with inertia $J_2$ about the given axis. Thus, for a given angular rotation, $\gamma$, of the pseudo-wheel, a straightforward integration of the dynamic equations implies that the satellite will experience a counter-rotation (in the opposite direction) of

$$\theta = -\frac{J_2}{J_1 + J_2} \gamma.$$  

(1)

In general, the rotation that can be achieved by the pseudo-wheel is bounded. If we let $\gamma_{\text{max}} > 0$ denote the maximum deflection of the pseudo-wheel, then $|\gamma| \leq \gamma_{\text{max}}$ implies that each individual actuator can impart to the rigid body a maximum rotation of $|\theta| \leq \theta_{\text{max}} = \frac{J_2}{J_1 + J_2} \gamma_{\text{max}}$. A small experiment demonstrating the feasibility of such an actuator is described in Section 4, with more details found in [13]. Based on these experiments, we have chosen $\theta_{\text{max}} = 1^\circ = 0.0175$ rad.
We stress that the problem we introduce here is unique from previously studied satellite reorientation problems in two major respects:

1. In problems using thrusters and momentum wheels, there are generally no limits on the amount of rotation that can be achieved about an actuated axis by a single actuator. In contrast to this, the actuators we consider provide only a bounded net rotation, and pay a penalty in terms of energy consumption if they are not returned to their rest state. In addition, when working with bounded actuation one generally finds that control authority is lost whenever the actuator is at or near its limits. In other words, if the actuator is in its on state, it cannot generate any additional deflection in a subsequent step. Therefore, we insist that the actuators be returned to their rest state at the end of the orientation maneuver.

2. In most previous problems where nonholonomic effects have been considered, two degrees of freedom are fully actuated and the primary focus is on providing attitude control for the third. In our problem, all degrees of freedom are actuated by bounded and fully reversed inputs, but we must effectively use the nonholonomic effect, or anholonomy, to reorient the system about all three axes.

The challenges that arise due to these two restrictions will be discussed below. In terms of how this impacts our ability to utilize previous research, we note that much successful work on steering for nonholonomic systems such as these implicitly assumes unbounded actuator inputs. For example, work by Leonard and Krishnaprasad [11] and Murray and Sastry [17] utilize coupled periodic inputs to control degrees of freedom that are not directly actuated. It is always assumed, however, that the degrees of freedom on which the control acts can be effectively ignored—these degrees of freedom are directly controllable. In the current situation, the bounded inputs imply that all degrees of freedom must be indirectly controlled through coupled inputs. Of course, this leads us to consider in Section 3.1 sequential inputs in which each axis of the satellite is controlled separately. While this clearly can be used to solve the problem, it is rather inelegant. Rather, one would hope to control the motion of all degrees of freedom simultaneously, as is done in Section 3.2.

In addition, the fact that the inputs are required to return to their original state (we term this “fully-reversed”) provides an additional obstruction to using previous results. In most cases, the coupling of inputs also requires a proper phasing between cyclic pairs of inputs. In the case of kinematic sys-
tems (in contrast to dynamic systems [5, 20]), it is most often required that the inputs be driven out of phase (e.g., coupling a sine with a cosine term). In order to maximize the amplitude of the inputs, this implies that one of the inputs should start (and finish) at the maximum amplitude. This directly contradicts the requirements in our situation of returning the actuators to their “off” state.

In the work we present below, we will thus constrain the types of inputs that can be utilized, in order to help limit the scope of the problem and the presentation of solutions. In addition to the above mentioned constraints, we will simplify the analysis by choosing inputs sequentially and in a “binary” manner. In other words, at step $i$ an individual actuator can be turned on to generate a rotation in either direction, $-\gamma_{\text{max}} \leq \gamma_i \leq \gamma_{\text{max}}$, and then turned off at some subsequent step to reach its rest configuration, $\gamma = 0$. At the end of any given actuation sequence, the actuators are all returned to their off position. The use of sequential, on-off input sequences helps simplify the analysis even further beyond the coupled sinusoids used in past research on steering. In addition, this makes better sense as a model input space for microactuators, which would run open-loop at high frequencies (in the kHz range), making time-parameterized input paths such as sinusoids less feasible.

Lastly, we note that an optimal solution to this problem should consider energy consumption and the total time taken to complete a re-orientation maneuver as paramount concerns in choosing a motion planning sequence. A natural cost function based on energy consumption for an electro-thermal actuator is

$$C_E = \int_0^{t_f} \sum_{i=1}^3 \|V_i I_i\| dt, \quad (2)$$

where $V_i$ and $I_i$ represent the voltage and current applied to the $i^{th}$ pseudowheel. For our purposes, we investigate energetic costs using a slightly modified metric, in which a time integral of the deflection angle of the actuators is used in place of the power consumption:

$$C_t = \int_0^{t_f} \sum_{i=1}^3 |\gamma_i(t)| \ dt, \quad (3)$$

where $\gamma_i(t)$ denotes the deflection at time $t$ of the actuator about the $i^{th}$ axis. Although we do not explicitly attempt to optimize these cost functions in our control algorithms, we use them as heuristics to guide our choice of controllers. For example, this motivates our requirement stated above that
the actuators are returned to a zero configuration after being deflected. In our simulations, we plot the cost function to give a sense of the respective costs of actuation. Lastly, we note that we do not consider the transient dynamics of the actuators themselves in this paper. Instead, we assume that the actuators can be turned on or off and reach their desired states at a rate of 1 kHz (see Section 4 for more details).

2.2 The motion planning problem

The problem we address is a motion planning problem that consists of determining the sequence of bounded and fully reversed rotations about individual axes that will reorient the satellite from an initial orientation, \( R_0 \in SO(3) \), to a final orientation, \( R_f \), where \( SO(3) \) is the special orthogonal group and represents rigid body orientation. Without loss of generality, we assume that the initial orientation is the identity, \( R_0 = I \), so that the goal is to generate a net attitude shift by \( R_f \).

![Figure 2: Sequence of body-axis rotations leading to curve in SO(3) to desired orientation.](image)

To make the initial discussion clearer by way of an example, consider a body-fixed \( xyz \) coordinate system, as shown in Figure 2. To start a rotation sequence, we deflect each of the actuators mounted on the body axes by \( \gamma_x \); \( \gamma_y \), and \( \gamma_z \) (in that order), to generate a rotation relative to the respective body-fixed axes of \( \theta_x \), \( \theta_y \), and \( \theta_z \). The net rotation of the rigid body will clearly depend on the order in which these rotations are reversed. If for example, the actuators are turned off in the same order, the final orientation
of the rigid body will be given by
\[
R = R_{xyzx}^{-} = R_z(-\theta_z)R_y(-\theta_y)R_x(-\theta_x)R_z(\theta_z)R_y(\theta_y)R_x(\theta_x),
\]
where \(R_i(\theta_i)\) denotes a rotation by \(\theta_i\) about axis \(i\). We assume that the deformation of each individual actuator is complete before the next actuation is initiated. We will use the notation in Eq. 4 to specify the sequence of actuator deflections. For example, \(R_{xyzx}^{-}\) signifies that the actuators are turned on in order \(x, y, z\) and then returned to their rest ("off") state in a slightly different order, \(-x, -z, -y\). Alternatively, we will denote the series of rotations by their magnitude, listing them in the order in which they are applied, so that the sequence corresponding to this example would be \(\{\theta_x, \theta_y, \theta_z, -\theta_x, -\theta_y, -\theta_z\}\). Notice that Eq. 4 thus gives the forward kinematics for this problem. Repeated application of this type of sequence leads to a trajectory in \(SO(3)\), as is sketched in Figure 2, where motion at the end of each step is denoted by \(R_i\).

If, on the other hand, the desired final rotation is given by \(R_f\), then we could ask whether solutions exist to the inverse kinematics problem. This problem involves computing the values of \(\theta_x, \theta_y, \text{ and } \theta_z\) that will lead to a final orientation of \(R_f\). Clearly, the sequence shown in Eq. 4 is only one of many possible ways of ordering the rotations. Even if we allow only one rotation and its reversal for each axis in a binary manner, i.e., a total of six rotations in the sequence, there will be 24 distinct possibilities. If we were to permit several rotations about each axis with the condition that the net rotation about each is zero by the end of the sequence, there would be an unlimited number of possibilities. In this paper, we restrict our attention to the kind of sequence shown in Eq. 4 and its 23 other permutations. These are shown using the aforementioned shorthand notation in the left column of Table 1. We will return to the issues of the right hand column of this table and why the sequences are divided into three groups later in the paper.

### 2.3 Existence of solutions

Unfortunately, the issue of existence of solutions and the minimal number of individual rotations about different axes necessary to guarantee a solution is still an open issue. Let us examine this further. Consider the sequence of rotations, \(R_{xyzx}^{-}\), given in Eq. 4. A good first question to ask is

\[\text{Additionally, there are six more unique sequences that do not generate any rotation about one of the three axes. For example, } \{\theta_x, \theta_y, \theta_z, -\theta_x, -\theta_y\} = \{\theta_x, \theta_y, -\theta_x, -\theta_y\} \text{ is one of them. The inverse kinematics problems for these cases can be solved in closed-form. See [13] for details.}\]
Table 1: List of possible sequences with fully reversed inputs and their corresponding leading order, resultant net rotation in unskewed ($S^3$) form

<table>
<thead>
<tr>
<th>Rotation sequence</th>
<th>Screw motion to second order</th>
</tr>
</thead>
<tbody>
<tr>
<td>${\theta_x, \theta_z, \theta_y, -\theta_z, -\theta_y, -\theta_x}$</td>
<td>${\theta_y \theta_z, 0, 0}$</td>
</tr>
<tr>
<td>${\theta_x, \theta_y, -\theta_z, -\theta_y, -\theta_x}$</td>
<td>${-\theta_y \theta_z, 0, 0}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_z, \theta_x, -\theta_z, -\theta_x, -\theta_y}$</td>
<td>${0, -\theta_x \theta_z, 0}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_x, -\theta_z, -\theta_z, -\theta_x, -\theta_y}$</td>
<td>${0, \theta_x \theta_z, 0}$</td>
</tr>
<tr>
<td>${\theta_x, \theta_y, -\theta_z, -\theta_x, -\theta_y, -\theta_z}$</td>
<td>${0, 0, \theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_z, \theta_x, \theta_y, -\theta_x, -\theta_y, -\theta_z}$</td>
<td>${0, 0, -\theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_z, \theta_x, -\theta_z, -\theta_y, -\theta_x}$</td>
<td>${0, -\theta_x \theta_z, \theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_z, \theta_y, \theta_x, -\theta_z, -\theta_x, -\theta_y}$</td>
<td>${-\theta_y \theta_z, 0, \theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_x, \theta_z, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${-\theta_y \theta_z, -\theta_x \theta_y, 0}$</td>
</tr>
<tr>
<td>${\theta_x, \theta_y, \theta_z, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${-\theta_y \theta_z, 0, -\theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_z, \theta_x, \theta_y, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${\theta_y \theta_z, -\theta_x \theta_y, 0}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_x, \theta_z, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${\theta_y \theta_z, 0, \theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_x, \theta_z, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${-\theta_y \theta_z, 0, -\theta_x \theta_y}$</td>
</tr>
<tr>
<td>${\theta_z, \theta_x, \theta_y, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${\theta_y \theta_z, -\theta_x \theta_y, 0}$</td>
</tr>
<tr>
<td>${\theta_y, \theta_x, \theta_z, -\theta_x, -\theta_z, -\theta_y}$</td>
<td>${-\theta_y \theta_z, \theta_x \theta_y, 0}$</td>
</tr>
</tbody>
</table>
whether the mapping described by Eq. 4 is at least a local diffeomorphism\(^3\) of \(S^3 \rightarrow SO(3)\) about the point \((\theta_x, \theta_y, \theta_z) = (0, 0, 0)\). Here, \(S^3 = S \times S \times S\) is the product space of three rotations, and we note that locally \(S^3\) is diffeomorphic to \(\mathbb{R}^3\). Unfortunately, it is clear that there does not exist even a local diffeomorphism for the mapping described in Eq. 4 or any of the other 23 permutations of input sequences that involve a single on-off pair for each of the three axes. In fact, the mapping is neither locally injective nor locally surjective, which we state in two separate propositions:

**Proposition 2.1** The map \(R^{\text{xyz}x^{-y^{-z^{-}}}}\) described by Eq. 4, as well as each of the maps given in Table 1, is not locally injective.

**Proof:** It is easy to see that the mapping for any of these combinations is not injective (one-to-one), by noticing that there are several combinations of inputs that will lead to no net rotation. For example, any sequence for which two inputs are zero (e.g., \(\theta_x = \theta_y = 0\)) leads to no resultant orientation change, i.e., \(R^{\text{xyz}x^{-y^{-z^{-}}} = R^{zz^{-}}} = I\), the identity element. \(\blacksquare\)

More important for the current situation, however, is whether there exists a solution to the inverse kinematic problem, i.e., in determining \((\theta_x, \theta_y, \theta_z)\) necessary to generate a desired \(R_f\). This, coupled with the requirements that we use bounded inputs, leads to an issue of local surjectivity. That is, whether or not local neighborhoods around the origin in \(S^3\) map onto a neighborhood of the identity element in \(SO(3)\). We see in the following proposition that this is not the case. The proof is deferred until the appropriate notation has been introduced in Section 3.2 below.

**Proposition 2.2** The map \(R^{\text{xyz}x^{-y^{-z^{-}}}}\) described by Eq. 4, as well as each of the maps given in Table 1, does not map local neighborhoods of \(0 \in S^3\) surjectively onto local neighborhoods of \(I \in SO(3)\).

Before leaving these issues, we note that while local regions in \(S^3\) do not map surjectively to \(SO(3)\), the question of whether there exist solutions to the inverse kinematics problem when bounded inputs are not required is still an open one. Our initial numerical investigations of this suggest that the mapping might be globally surjective from \(S^3\), but not locally. In other words, for any local neighborhood of the identity in \(SO(3)\), there seems to exist a set of \((\theta_x, \theta_y, \theta_z) \in S^3\) that generates it. We state this as a conjecture:

\(^3\)It is a well-known result that there is no global diffeomorphism from \(S^3 \rightarrow SO(3)\).
Conjecture 2.3 The map $R^{xydez}y^{-z-}$ described by Eq. 4 (and perhaps each of the maps given in Table 1) is surjective as a map from $S^3$ to local neighborhoods of $I \in SO(3)$.

Indeed, we have found through numerical solution techniques that the magnitudes of the required inputs needed to generate small, local rotational motions using the sequences described in Table 1 can actually be quite large. An example of this is described in Section 5 below, where we have found through numerical experiments that it requires rotations of $\theta_x = 0.3$, $\theta_y = 0.31$, and $\theta_z = -0.04$ rad in Eq. 4 to generate a simple rotation about the $z$-axis of only $\alpha_z = 0.1$ rad.

In the next section, we present two distinct methods to solve the motion planning problem using approximate methods. A pair-wise method is discussed in Section 3.1, and a single, minimal sequence method is presented in Section 3.2. Both algorithms obey the boundedness and fully-reversed nature of the inputs space.

3 Motion planning approaches to the problem

In practice, we are interested in generating large rotations that are not achievable by any simple combination of small inputs. For this reason, it is important to first decompose the overall motion into a series of steps. To do this, we take an incremental approach that seeks to follow the optimal, or geodesic, curve between the initial and desired configurations.

The construction of this path is as follows. Denote by $\hat{\omega} \in \mathfrak{so}(3)$ the body velocity defined by $\hat{\omega} = R^T \dot{R}$, where $\langle \cdot, \cdot \rangle$ denotes the skew-symmetry operation on $\mathbb{R}^3$. Let a Riemannian metric for $SO(3)$ be defined by $\langle R, \dot{R} \rangle_R = \frac{1}{2} (\hat{\omega})^T \hat{\omega} = \omega^T \omega$. Given any curve connecting two points in $SO(3)$, say $R(t), t \in [0, t_f]$, define the length of that curve to be

$$l = \int_0^{t_f} \langle \dot{R}(t), \ddot{R}(t) \rangle_{R(t)}^\frac{1}{2} dt = \int_0^{t_f} (\omega(t)^T \omega(t))^\frac{1}{2} dt. \tag{5}$$

Then, the curve connecting $R_0 = I$ to $R_f$ that minimizes the length of the path is the one that follows everywhere the screw motion defined by

$$\dot{\xi} = \log(R_f) = \frac{\phi}{2 \sin \phi} (R - R^T), \tag{6}$$

where $\phi$ represents the magnitude of the rotation and satisfies $1 + 2 \cos \phi =
The variable \( \hat{\xi} \) represents a skew symmetric matrix, is formally an element of the Lie algebra, \( \mathfrak{so}(3) \), of \( SO(3) \), and represents the velocity of the rigid body in a body-fixed frame (see [2]). The logarithm mapping, \( \log : SO(3) \rightarrow \mathfrak{so}(3) \), returns the body velocity that generates a given rotation (via Eq. 6). Similarly, we can define an exponential mapping, \( \exp : \mathfrak{so}(3) \rightarrow SO(3) \), that maps body velocities to rotations. For \( \xi \in \mathfrak{so}(3) \), the element \( \exp \xi \in SO(3) \) corresponds to the solution at \( t = 1 \) of \( \dot{R}(t) = R(t) \hat{\xi} \). For small desired rotations (i.e., when \( \|\xi\| \) is small), it may be possible to generate the motions using a simple sequence of inputs. More generally, however, we can divide the motion into a sequence of \( N \) repeated inputs by using instead \( \xi^{des} = \frac{\xi}{N} \). Since in practice we will use approximations to generate the desired motions, it will be advantageous to re-evaluate the desired rotation at each step along the way, in order to provide better tracking. Alternatively, we adaptively decrease the size of the increment until the original path is followed everywhere within a specified tolerance.

Since each body-fixed rotation can be written as the exponential of a Lie algebra motion (alternatively, think of this as exponentiating the skew-symmetric matrix associated with a twist), we can write, for example, the rotation sequence given in Eq. 4 as

\[
R^{xyz \ x \ y \ z} = \exp(-\xi_x \theta_z) \exp(-\xi_y \theta_y) \exp(-\xi_z \theta_x) \exp(\xi_z \theta_z) \exp(\xi_y \theta_y) \exp(\xi_z \theta_x),
\]

where \( \xi_x, \xi_y, \xi_z \) represent the Lie algebra elements (skew-symmetric matrices) associated with rotations about the body-fixed \( x, y, \) and \( z \) axes, respectively. The desired rotation is simply \( R_f = \exp(\xi_{des}) \), so if we can write \( R^{xyz \ x \ y \ z} \) from Eq. 7 as a single exponential, \( \exp(\eta(\theta_x, \theta_y, \theta_z)) \), then we can hope to solve for the necessary input sequence, i.e., for \( (\theta_x, \theta_y, \theta_z) \), by solving \( \eta(\theta_x, \theta_y, \theta_z) = \xi_{des} \).

To generate this single exponential, we invoke the Campbell-Baker-Hausdorff (CBH) formula, and note that for the small deflections we are using to generate rotations, we can truncate the resultant polynomial series to provide a good approximation of the actual rotation. Recall the basic CBH result. Let \( R_1 = \exp(\eta_1) \) and \( R_2 = \exp(\eta_2) \) be two elements of \( SO(3) \) in a neighborhood \( U \) of the identity element, \( I \), and generated respectively by \( \eta_1, \eta_2 \in \mathfrak{so}(3) \). Then, there exists \( \eta_3 \in \mathfrak{so}(3) \) such that

\[
\exp(\eta_3) = \exp(\eta_1) \exp(\eta_2).
\]

\(^4\)We will not worry at this point about uniqueness of the log function, but instead assume that this value exists.
Furthermore, the element $\eta_3$ can be explicitly expressed in the Dynkin form as
\[ \eta_3 = \eta_1 + \eta_2 + \frac{1}{2} [\eta_1, \eta_2] + \frac{1}{12} [\eta_1, [\eta_1, \eta_2]] + \frac{1}{12} [\eta_2, [\eta_2, \eta_1]] + \cdots \quad (9) \]

Repeated application of the CBH formula allows us to write Eq. 7 for $R^{xyzx - y - z -}$ as a single exponential in which the exponential is described by a (infinite) polynomial series in $(\theta_x, \theta_y, \theta_z)$. Assuming small rotations, we can truncate this series to arrive at approximate solutions to the desired input sequences. The resultant Lie algebra element, $\xi_{xyzx - y - z -}$ can be written to third order (and in an unskewed form) as
\[ \xi_{xyzx - y - z -} = \begin{pmatrix} -\theta_y \theta_z + \frac{5}{12} \theta_y (\theta_y^2 + \theta_z^2) \\
\theta_x \theta_z - \theta_y \left( \frac{5}{12} \theta_z^2 + \frac{1}{3} \theta_x^2 \right) \\
-\theta_x \theta_y + \frac{5}{12} \theta_z (\theta_x^2 + \theta_y^2) \end{pmatrix} . \quad (10) \]

In the following two subsections, we present two methods that use these concepts to generate the rotation sequence to yield the desired attitude shift. In Section 3.1, we use three sets of pair-wise rotations, where each pair is used to control a single axis. This approach is similar to previous approaches in the literature—for example, Leonard and Krishnaprasad [11] used a similar setup to control a single axis of a satellite. Here we control all three axes using the same basic principle and with discrete (on/off) actuation. It is proven to converge for all initial conditions, but may require more rotations (and thus incur a higher cost) than is necessary. In contrast, the algorithm discussed in Section 3.2 demonstrates a control sequence that can be chosen to generate desired rotations using the minimal (six-step) sequence. We give some discussion to when this technique is applicable.

### 3.1 Pair-wise rotations

As was noted in Section 2.3 above, a sequence of fully reversed rotations about two body-fixed axes can be used to generate a rotation about the third axis that is accurate to second order. This can be clearly seen by setting any of the individual rotations to zero in Eq. 10, and is just the traditional nonholonomic effect studied elsewhere, though primarily this effect has been generated using momentum wheels. For example, given the rotation sequence, $R^{xyz - y - z -}$:
\[ R^{xyz - y - z -} = R_y (-\theta_y) R_x (-\theta_x) R_y (\theta_y) R_x (\theta_x), \quad (11) \]

setting $\theta_z = 0$ in Eq. 10 gives us that the resulting motion will be approximately (to second order) $R_z (-\theta_x \theta_y)$. Likewise, setting either $\theta_x$ or $\theta_y$ to zero can be used to generate rotations about the $x$- or $y$-axes, respectively.
In the case that the desired rotation, $R_f$, is about one of the principal body-fixed axes, this approximation can be used to select a sequence of inputs that approximately yields the rotation. If we decompose any general desired rotation into a sequence of three successive rotations about the body-fixed axes, then these approximations can be used at each successive step. In essence, this is equivalent to decomposing the desired rotation into an Euler angle sequence (e.g., $Z$-$Y$-$X$ Euler angles), and then executing motions along each successive rotation.

In our implementation, we have used a slightly different motion planning scheme that enables better feedback control. To generate a desired screw motion, $\xi = \log(R_f)$, we first divide it into a series of smaller steps as described above, by scaling $\xi$ appropriately so that the largest component is less than $\theta_{max}^2$, the maximum rotation that can be achieved using a pair-wise actuation of the pseudo-wheels. Then, we perform a sequence of rotations, starting with $\xi_z^{des}$ (the $z$-axis component of $\xi^{des} = \xi/N$) where we generate the desired motion about the $z$-axis by combining an input pair of $\theta_x = -\theta_y = (\xi_z^{des})^{\frac{1}{2}}$. While this gives approximately the desired rotation, it is not exact—instead of a change in orientation by $\Delta R = \exp(\xi^{des})$, we get $\Delta \hat{R}$. At this stage, we can either continue with a set of rotations about the next axis, or we can determine a new $\xi^{des}$ for the next step. This can be done either by taking a measurement of the new desired rotation, or by computing the effect of the actual rotation on the desired. In other words, at this step, which we denote by $i + \frac{1}{3}$ since each step requires three sub-sequences, the new desired rotation is $R_{i+\frac{1}{3}} = \Delta \hat{R}^T R_i$. From this, a new $\xi^{des}$ can be computed, and we continue by executing a sequence of rotations to generate an approximate attitude shift about the $y$-axis. This results in a motion that can again either be measured or modeled, to provide a new set point, $R_{i+\frac{3}{2}}$. The final component of the $(i + 1)^{th}$ step is then to use a $y$-$z$ actuation pair (i.e., $R_{yz}^{\text{y}z}$) to generate a rotation about the $x$-axis. This procedure is then repeated until the attitude converges to the desired one.

**Proposition 3.1** The control algorithm described above, which takes an approximate step proportional to $\log(R_f^x R_i)$ at each step, converges to the desired value, $R_f$. 

14
Proof: Let

\[
\mathcal{L}(R) = \| R - R_f \| = \text{trace}[(R - R_f)(R - R_f)^T] = \text{tr}([R_f^T(R - I)]^T(R_f^T R - I)) = 2(3 - \text{tr}(R_f^T R)).
\]  

(12)

This function is positive everywhere, except for at \( R = R_f \), where it is zero. Note in particular, that if we write \( R = \exp(\hat{\xi}) \), Eq. 12 can be written as \( \mathcal{L}(R) = 4(1 - \cos(\| \xi \|)) \). We will use this function as a discrete-time Lyapunov candidate function.

To compute the change in \( \mathcal{L} \) with each step of the algorithm, suppose we start at the point \( R_i \), with \( R_f^T R_i = \exp(\hat{\xi}_i) \), for some \( \xi_i \in (-\pi, \pi] \). Then, \( \mathcal{L}(R_i) = 4(1 - \| \xi_i \|) \). The control law is chosen to generate a rotation in the direction specified by \( \xi_i \), which we will call \( \xi_i^{\text{des}} = -\epsilon \xi_i \), for small \( \epsilon > 0 \). However, as noted above the control law generates only an approximate rotation, so that after the control sequence is applied, the new rotational state of the satellite will be such that \( R_f^T R_{i+1} = \exp(\hat{\xi}_i - \epsilon \xi_i + \gamma_i) \).

For a reasonably small choice of \( \epsilon \), the CBH formula implies that the perturbation should be small compared with the leading order term used in the algorithm, i.e., \( \| \epsilon \xi_i \| = \epsilon \| \xi_i \| > \| \gamma_i \| \). The Lyapunov function at this step satisfies the following relationship:

\[
\mathcal{L}(R_{i+1}) = 4(1 - \cos(\| \xi_i - \epsilon \xi_i + \gamma \|)) = 4(1 - \cos((1 - \epsilon) \| \xi_i \| + \| \gamma \|)) \\
\leq 4(1 - \cos((1 - \epsilon) \| \xi_i \| + \| \gamma_i \|)) < 4(1 - \cos((1 - \epsilon) \| \xi_i \| + \epsilon \| \xi_i \|)) = 4(1 - \cos(\| \xi_i \|)) = \mathcal{L}(R_i).
\]

In other words, \( \Delta \mathcal{L} = \mathcal{L}(R_{i+1}) - \mathcal{L}(R_i) < 0 \), i.e., it is negative definite. Thus, the rotation converges to the desired rotation, \( R_f \), that makes \( \mathcal{L}(R) \) zero [8].

Remarks:

1. It is interesting to note that the proof requires only that the desired rotation be updated at the completion of every 12-step sequence. It can be seen from the proof (and simulations presented below) that if this is not done, convergence is not assured. The algorithm simulated here, however, update the position at the end of each individual axis rotation (i.e., every four steps).
2. A very similar proof can be generated in the continuous setting, using a feedback law of the form $\omega = -\epsilon \zeta$, for $\omega$ the instantaneous body velocity of the satellite.

A numerical simulation of this procedure is shown in Figure 3. The satellite is initially at the origin, and is driven to a relatively arbitrary location (here, $R_f = R_z(\pi/3)R_y(\pi/3)R_x(\pi/3)$). Notice that because the step size is very small, the plot of the error trajectory essentially looks like a straight line in the log space. The second plot in Figure 3 shows the error that is present during the course of the maneuver. The error norm we use is the square of the Frobenius matrix norm,

$$|R_{\text{error}}|^2 = \text{trace}(R_{\text{error}}^T R_{\text{error}}),$$  \hspace{1cm} (13)

where $R_{\text{error},i} = R_i - R_f$ is the rotation error at the $i^{th}$ step. As is clear from the figure, the error quickly goes to zero, as the trajectory is completed by this actuation sequence.

![Figure 3: Error during motion, as screw axis (3D) and as error norm](image)

In Figure 4, we show the cost of the motion as a function of time, as given by the time-based measure, Eq. 3. This also follows a linear relationship, since essentially the same rotation is carried out during each three-step, pair-wise sequence.

The above strategy works quite well, when feedback is present. We stress the need for feedback during the motion; since we are using only an approximate, model-based procedure, it is clear that errors will arise during the actuation sequence. In Figure 5 we present a simulation that motivates this need for feedback in this system. The solid line shows the strategy described above, while the dashed line shows the result when the system
runs open-loop—that is, when a single measurement is taken at the start of the motion, and the appropriate number of steps are taken using that initial estimate. As can be seen from the figure, the open-loop procedure actually matches quite closely the one using feedback for much of the motion, but then is unable to converge to the desired value, due to the approximate nature of the solution.

Lastly, we note that a slightly altered scheme can be used, with only moderate improvements in convergence, by varying the order in which the pair-wise actuations are chosen (e.g., \{R^{xyz - y -}, R^{yz - z -}, R^{zxx - x -}\} vs. \{R^{zxx - x -}, R^{yz - z -}, R^{xyz - y -}\}), based on \(\xi^{des}\) at each step. This is akin to trying to choose the best Euler angle parameterization at each step. One way to implement this is to choose as the axis to rotate about the one that corresponds to the smallest component of \(\xi^{des}\). For example, suppose that...
\( \xi_y^{\text{des}} \) is the smallest value. Then, we use a combination input sequence of \( \theta_x \) and \( \theta_z \) to generate a rotational motion about the \( y \)-axis. Next, the second largest value is chosen, and so forth. This has the advantage that the smaller initial rotations contribute less error to the final rotations, since we have used a truncated series to approximate our input solutions.

### 3.2 Using a single sequence of rotations

The above procedure is guaranteed to converge to the desired solution using the same sequence of rotations (but with different magnitudes) at each stage. However, it requires at each incremental step a sequence of 12 rotations, four for each of the three pair-wise, fully reversed rotations. An alternative approach is to achieve the incremental step in a single sequence of six rotations generated by turning on and off the actuators for each axis. To solve this problem, we again rely on the exponentiation operation on the Lie group to equate sequences of body-fixed rotations with the desired rotation, \( \mathbf{R}_f \), or its log, \( \xi^{\text{des}} = \log(\mathbf{R}_f) \).

Equating \( \xi_{xyz\bar{x}-y\bar{z}} \) in Eq. 10 with \( \xi^{\text{des}} \), one could potentially solve numerically for the necessary input sequence. However, if we truncate the series up to second order terms, then we can be more explicit about the types of solutions that can be found. Let \( \xi^{\text{des}} = (\phi_1, \phi_2, \phi_3) \). Then, to second order, Eq. 10 reads:

\[
\begin{align*}
-\theta_y \theta_z &= \phi_1 \\
\theta_x \theta_z &= \phi_2 \\
-\theta_x \theta_y &= \phi_3,
\end{align*}
\]

which has solutions described by

\[
\begin{align*}
\theta_x^2 &= \frac{\phi_2 \phi_3}{\phi_1} \\
\theta_y^2 &= \frac{\phi_1 \phi_3}{\phi_2} \\
\theta_z^2 &= \frac{\phi_1 \phi_2}{\phi_3},
\end{align*}
\]

whenever \( \phi_1, \phi_2, \phi_3 \neq 0 \).

These equations reveal two very interesting insights into the limitations placed on using this type of sequence, namely, i) real solutions only exist when the product of the components of \( \xi^{\text{des}} \), that is \( \phi_1 \phi_2 \phi_3 \), is positive, in which case two solutions, opposites of each other, exist; and ii) solutions
only exist when all of the components of $\xi^{des}$ are not equal to zero (although this could be stated also as $\phi_1\phi_2\phi_3 > 0$, it highlights a separate issue). Next we address each of these issues individually.

First, if we assume that all of the entries for $\xi^{des}$ are nonzero, notice that solutions only exist for screw motions that have an even (including zero) signature—that is, for which $\xi^{des}$ has only zero or two negative entries ($\{+,+,+\},\{-,+,+\},\{+,+,\}$, and $\{+,+,\}$). In $\mathbb{R}^3$ (locally), this implies that solutions can be determined for four of the eight octants. In these cases, there exist two roots for each of the values of $\theta_x, \theta_y, \theta_z$, yielding a total of six potential solutions. However, checking these solutions against Eq. 14, we see that only two of the combinations are true solutions, and that these two solutions are equal and opposite, namely, $(\theta_x, \theta_y, \theta_z)$ and $(-\theta_x, -\theta_y, -\theta_z)$.

Before continuing with this discussion, we pause briefly to prove Proposition 2.2, since it is a direct result of the ideas just presented:

**Proof:** (of Proposition 2.2) The proposition follows directly from the observations made just above—locally around $(0,0,0) \in S^3$, only certain regions of $SO(3)$ around the identity, $I$, can be mapped to using $R^{xyz}_{x-y-z}$. Rotations described by screw motions not contained in the four octants mentioned above cannot be generated, and so local regions around $0 \in S^3$ do not map to open sets (local neighborhoods) around $I \in SO(3)$.

The implication of Proposition 2.2 is that the sequence given by $R^{xyz}_{x-y-z}$ is not sufficient to control the satellite, since solutions cannot be found for the remaining four octants. In Table 1, we list this rotation (the last entry), as well as the other 23 possible sequences, and their corresponding leading order (unskewed) single exponential representation as derived by using the CBH formula. Comparing the entries in the bottom grouping, we notice that a sequence given by $R^{zyxz}_{y-x-z}$ gives an approximate rotation by

$$R^{zyxz}_{y-x-z} \approx \exp(\text{skew}(\theta_y \theta_z, -\theta_x \theta_z, \theta_x \theta_y)).$$

This has solutions that are exactly complementary to those found in Eq. 14, namely:

$$\theta_x^2 = -\frac{\phi_2 \phi_3}{\phi_1}$$
$$\theta_y^2 = -\frac{\phi_1 \phi_3}{\phi_2}$$
$$\theta_z^2 = -\frac{\phi_1 \phi_2}{\phi_3}$$
whenever $\phi_1, \phi_2, \phi_3 \neq 0$. As could be expected, the solutions for this sequence exists only when the number of negative entries in $\xi^{des}$ is odd.

Thus, we find that any desired rotation for which $\log(\xi^{des})$ does not have any zeros can be generated using one of the two sequences, $R^{xyzx-y-z}$ or $R^{zyxz-y-x}$. This provides at least a partial theoretical answer to when solutions can be found, but it does not address the practical matter of the strict bounds that we have assumed on the magnitude of deflection of the inputs. We require that $\theta_i \leq \theta_{max}$ for $i \in \{x, y, z\}$, but this will clearly be violated in many cases as the ratios of $\phi_1, \phi_2, \phi_3$ vary. To solve this problem, we again note that we can divide the desired rotation into a large number of smaller steps, by taking $\xi^{des} = \frac{\log(R)}{N}$, where $N$ is the number of steps. Doing this also rescales the equations that need to be solved in order to determine $(\theta_x, \theta_y, \theta_z)$. The solutions (for $R^{xyzx-y-z}$) thus become:

$$
\theta_x^2 = \frac{\phi_2 \phi_3}{N \phi_1},
\theta_y^2 = \frac{\phi_1 \phi_3}{N \phi_2},
\theta_z^2 = \frac{\phi_1 \phi_2}{N \phi_3},
$$

and it is clear that by appropriately scaling $N$ large enough, a solution can always be found that satisfies the actuator bounds.

This rescaling effect gives a satisfactory solution to determining appropriately bounded solutions, as long as $N$ is not too large, in which case it would take an unreasonable amount of time to generate the desired reorientation. As it turns out, this issue is closely tied to the second problem we need to address—what to do when any of the entries, $\phi_1, \phi_2, \phi_3$, go to zero. To solve this problem, we again turn to Table 1. The middle grouping in this table shows that there are 12 sequences for which motion is generated about only two of the three screw axes. In fact, we notice that for each of the entries, the correct choice for $(\theta_x, \theta_y, \theta_z)$ can be used to generate one, two, or (trivially) three zeros for the net leading order rotation. Thus, we can use these, chosen appropriately, whenever there exist one or more zeros in $\xi^{des}$.

To reinforce, this, let us look at an example—consider the case where $\phi_1 = 0$, but $\phi_2, \phi_3 \neq 0$. Then, to satisfy $\phi_1 = 0$, we can pick the sequence given by $R^{xyzx-y-z}$, for which the generated motion to leading order is $\exp(\text{skew}(0, -\theta_2 \theta_z, \theta_x, \theta_y))$. If it turned out that the second entry, $\phi_2$, was also zero, then we could choose $\theta_z$ to be zero to satisfy this condition.
In practice, we also have found that these additional sequences provide a means for addressing the problem of the number of steps, $N$, growing very large, even though none of the entries in $\xi^{\text{des}}$ is zero. When this occurs, it indicates that one of the entries is approaching zero faster than the others. In this case, we can approximate this entry as zero and use the sequences in the middle grouping of Table 1 to move in the desired direction. Generally, this may imply switching between different types of actuation sequences, but the overall consequence is that the satellite can always be steered to the desired orientation.

![Figure 6: Error during motion, as screw axis (3D) and as error norm](image)

In Figure 6, we show a plot of the trajectory (in log space) and the corresponding error over time in order to generate the same desired net reorientation as was done in Section 3.1 using the pair-wise sequence, $R_f = R_z(\pi/3)R_y(\pi/3)R_x(\pi/3)$. In Figure 7, we show the cost associated with this motion.

Lastly, we have plotted the trajectory of the system for a large number of different initial conditions, to show that the procedure works well from any initial location. In the Figure 8, we show the trajectories to all possible combinations of the form: $R_f = R_z(k \cdot \pi/3)R_y(j \cdot \pi/3)R_x(i \cdot \pi/3)$, for $(i, j, k) \in \{-1, 0, +1\}$. Although we have not presented a formal proof of convergence (we feel that the arguments presented are strong enough), we have done extensive numerical simulations, using random desired orientations, to verify that the system always converges to the desired orientation. In the rightmost plot of Figure 8, we show the trajectories to the desired configuration for 100 randomly selected final configurations. To give a sense of convergence rates, the average time to completion of these 100 samples was 69s, using 1ms as the time to completely turn on or turn off a single...
Figure 7: Cost associated with given rotational motion

actuator, i.e., 1kHz actuation bandwidth. The worst case time was 141s.

Figure 8: Motions to desired configuration (0,0,0) for 26 different initial conditions; trajectories for 100 random initial conditions

4 Experimentation at the micro scale

To actuate the pseudo-wheel described in Section 2, for the purpose of proof-of-concept, we consider the small deflections but large forces generated by electrothermal compliant (ETC) mechanisms [16]. ETC devices work by using nonuniform Joule heating of various sections of the device to generate deflections. This principle is similar to the actuation found in bimetallic actuators but uses a structure made of only one material that has a suitable shape. An example of an ETC mechanism and its deflection are shown
in Figure 9. By changing the electrical connection from series to parallel, the actuator can be deflected in the opposite direction [16] (see Figure 9). The magnitude of the applied voltage determines the amount of deflection in either case. The maximum deflection is limited also by the maximum permissible temperature. The modeling, design, and microfabrication are presented in [14], [30] and [12] respectively. A disadvantage of thermal actuation is the need for heat removal after every cycle. This problem is accentuated in space applications where there is no convection. So, in addition to radiative cooling, conduction to the mounting part becomes the important mode of heat removal. The time constant for actuation can also be an issue. It has been shown that ETC actuators of this scale can be run at kHz frequencies [23]. Our transient simulations also indicate that the time for actuating these devices is on the order of milliseconds [13]. One important advantage of ETC actuators is that they can be designed for any voltage range and the aforementioned practical issues. Our purpose in this paper is only to show that ETC actuation can provide the required rotation about an axis. A schematic of an ETC pseudo-wheel [1] is shown in Figure 10.

![Figure 9: ETC mechanisms; ETC deflection using series and parallel connections](image)

We used a silicon-on-insulator wafer-based bulk-silicon micromachining process combined with an excimer laser micromachining. As shown in Fig. 10, several basic ETC actuators are arranged as a circular array. This helps increase the torque generated. To further enhance the torque, several layers of these can be stacked together as shown in Fig. 10. This is called an ETC pseudo-wheel as it does not rotate by itself but can cause rotation to the part on which it is mounted. The rotation of the part will be in a direction opposite to that of the deflections of the actuators. An
experimental prototype of cm-size with four actuators is shown in Fig. 9. This was suspended with thin gold wires as shown in Fig. 11. This ETC pseudo-wheel rotated by about 10 degrees for an applied voltage of 15 V. Details of the experimental work are discussed elsewhere [13].

Based on these experiments, we use a smaller value for the maximum rotation that can be effected in the rigid body, assuming roughly a 10:1 ratio between spacecraft-body and ETC inertias. For a value of $\theta_{max} = 1^\circ$, this

24
implies that a single pair-wise sequence of actuation steps, e.g., $R_{xy}$, leads to approximately $0.0175^\circ = 0.0003\text{rad}$ rotation about the desired axis of rotation. For actuation rates of, say, 1kHz, this would lead to a net rotation of roughly $0.05\text{rad/s} = 2.9^\circ/s$. Since the largest desired rotation would be $180^\circ$, this could be achieved in approximately 60s.

5 Discussion and future work

The unique situation of using an ETC pseudo-wheel to re-orient a micro-satellite has raised interesting issues in the control of a mechanical system with nonintegrable velocity constraints. In particular, the fact that these actuators must return to their undeformed state and have severe bounds on the magnitude of rotations about individual axes leads to new issues in nonholonomic control that have not previously been studied in the literature. Open issues still exist as to how to appropriately characterize fundamental properties such as controllability and reachability for such types of systems. We have provided some initial answers here, using approximate formulae to generate local motion plans, and least squares solution techniques to solve the problem when the bounds are less strict or not present.

There are several additional topics that merit further work. First, there is the issue of whether other types of actuators and configurations might yield better performance. The ETC actuators have significant advantages in terms of necessary supply voltages, and magnitudes of torques that can be exerted compared to many other microactuators; however, the need to dissipate heat is a severe limitation when working in space, and also is somewhat inefficient in that the energy is completely lost in turning off the motor. It would be worth exploring alternative actuation mechanisms such as piezoelectric, ferroelectric, etc. For example, Reiter et al. [24] recently proposed a moment gyroscope based on electrostatics. We also note that ETC actuation can be used to create a real wheel like the momentum wheel using microfabricated linkages, one-way clutches, and ratchets [21]. We did not consider them because they make the microfabrication more complicated and intermittent contacts present in them make the system less reliable.

Another issue is the use of the ETC pseudo-wheel. This is the most direct way of mimicking a momentum wheel, though clearly with limits on the magnitude of rotations that can be achieved. In addition, though, the satellite is only reoriented through combining the actuation about multiple axes, rather than directly controlling through a single actuation step. For this reason, it would be interesting to investigate whether a planar actuator
with higher degrees of freedom might be able to generate a nonholonomic
effect directly. For example, the planar, floating, four-bar linkages studied by
Yang and Krishnaprasad [29] provide an interesting starting point for such
a study. Using a compliant mechanism capable of moving a mass through
a circular trajectory would give an alternative way of generating controls to
provide reorientation. The key here is that while the net attitude change
in any single motion is small, these motions can be performed at very high
frequencies (kHz), yielding significant control authority over reasonable time
scales.

Lastly, we mention that it may be possible to numerically generate solu-
tions to the inverse kinematic problem of finding \((\theta_x, \theta_y, \theta_z)\) to satisfy
\(R^{x y z} - \theta = R_f\). By equating all the nine entries in the symbolically de-
duced \(R\) for any given sequence in Table 1 to a numerically specified \(R_f\), nine
algebraic-trigonometric equations are obtained. We can then seek to mini-
imize a residual based on the Frobenius norm given in Eq. 13 with respect
to three unknown angles of rotation in that sequence. One problem with this
method is that many local minima exist for the least squares minimization
problem and the correct solution is not necessarily obtained for any initial
guess given to the solution routine. A correct solution is the one for which
the residual is absolutely or very closely equal to zero. This warrants an
optimization algorithm that can obtain a global minimum. Deterministic
methods for obtaining global minima exist for special classes of problems.
One such class of problems is where both the objective and constraint func-
tions are generalized polynomials. Fortunately, the trigonometric terms in
the above nonlinear equations can easily be transformed to polynomials us-
ing the well-known \(\tan(\theta/2)\) substitution. This global minimization approach
will be the topic of future work.

It is also interesting to examine the magnitude of the rotations required
to achieve a small desired rotation. Consider, for example, the rotation
of Eq. 4, \(R^{x y z} - \theta = R_z(\theta)R_y(\theta)R_x(\theta)\). The se-
quence of rotations must pass through several intermediate positions to reach
the desired objective position. We can visualize the path of \(R^{x y z} - \theta\) by
measuring the distance between the desired position and the current posi-
tion. The distance is defined by the residual term of Eq. 13 above.

As an example, consider the goal of obtaining a desired rotation of \(R_z(\theta)\)
with \(\theta = 0.1\). If the allowable controls included a pure rotation about
the \(z\)-axis, the error seen during such a rotation about \(R_z\) would decrease
linearly. Using a sequence of on-off actuation, the same position is achieved
by \(R^{x y z} - \theta\) with \(\theta_x = 0.30, \theta_y = 0.31, \text{ and } \theta_z = -0.04\). Because the
actuation is done indirectly through a nonholonomic effect, the path taken in following the sequence of \( R^{xyz\bar{y}z\bar{z}} \) passes through intermediate positions that are far away from the shortest path, \( R_z \). Furthermore, the individual rotations that are needed, particularly \( \theta_x \) and \( \theta_y \), are much larger than the overall net rotation that is achieved. Thus, it is unlikely that any direct numerical solution can be used to solve the problem for bounded, fully-reversed inputs.

6 Conclusions

In this paper, we have presented a novel actuation mechanism for micro-satellites using microactuators. We have developed motion planning methods that can be used to perform satellite attitude control, while respecting the bounded and fully reversed nature of the inputs. Two incremental methods have been presented that use approximate (leading order) solutions to the problem. One method controls each axis of the rigid body independently, and so is less efficient, but is easier to understand and uses the same sequence of actuations at every step. The second method uses a minimum number of steps, but requires a more involved sequencing of actuation. We also have described a prototype experimental system using electro-thermal-compliant microactuators. The preliminary results suggest that this type of actuation can indeed be practical for micro-satellite attitude control.

Acknowledgments

The authors would like to thank Mr. Jun Li, who helped fabricate the ETC pseudo-wheel and supplied several figures of the experimental work. Pathway Technologies, Inc. provided the financial support in part for this work. The work was also supported by NSF grant IIS-9876301 and DARPA MURI grant DAAH04-96-1-0007. This support is gratefully acknowledged. The idea for the novel attitude control technique for miniature spacecraft originated in the initial discussions that the last author (GKA) had with Dr. Andy Suri of Pathway Technologies and Professor Ayyaswamy of University of Pennsylvania.

References

[1] G.K. Ananthasuresh, J. Li, P.S. Ayyaswamy, and A. Suri. A novel attitude control technique for miniature spacecraft using an electro-


