

Nonlinear inverse models for the control of satellites with flexible structures

Matthias J. Reiner Johann Bals

German Aerospace Center (DLR), Institute of System Dynamics and Control
82234 Wessling, Germany, {Matthias.Reiner, Johann.Bals}@DLR.de

Abstract

Nonlinear inverse dynamic models can be utilized in various parts of advanced model-based control system design: reference trajectory optimization, feed-forward control and feedback linearization [35]. In this paper, a new synthesis approach for nonlinear inverse dynamic models of satellites with flexible structures is presented. For satellite configurations with unstable zero dynamics, a stable inverse model approximation is proposed which has been successfully applied to robots with flexible bodies.

This inverse modeling approach is part of the newly developed DLR Space Systems Library for object-oriented modeling and simulation of satellites and launchers in a detailed space environment. For satellites with flexible structures, the library provides models for normal simulation mode and the necessary tools to directly generate approximate inverse models.

In this paper, trajectory optimization is shown to be an important use case for inverse dynamic models. By inversion based reformulation of the trajectory optimization problem, the optimal reference motion of the control system can be determined in a reliable and efficient way.

Keywords: satellite modeling; nonlinear inverse model; trajectory optimization; flexible structure

1 Introduction

The Space Systems Library (SSL) was built to develop advanced control systems for satellites / spacecraft with flexible structures. In particular, one goal of this library is to generate nonlinear inverse models for the controller. The library contains state-of-the-art Low Earth Orbit (LEO) space environment models and components. It was implemented in the Modelica modeling language [23].

The theory of satellite dynamics is well understood and many important aspects of spacecraft modeling

are described in publications such as [24]. A wide range of satellite simulators exist [3, 2, 11, 5, 36] that are able to accurately simulate a satellite in orbit.

There also exist simulators based on the Modelica modeling language. The authors made use of the advantages of using the Modelica language for the implementation and the inversion of rigid satellites [29]. In addition, several related publications such as [30, 20, 19] describe the simulation of satellites in detail. Satellites with flexible appendages were considered in [33]. Many important aspects of spacecraft modeling are already covered in these publications and promising results were reported by these authors.

The Space Systems Library was developed to implement nonlinear inverse models of satellites with flexible structures, such as solar panels, that cannot be easily implemented within existing simulators. Having direct access to all component equations from the SSL allows the successful implementation of algorithms from recent results obtained in the field of robotics regarding the inversion of flexible multi-body systems and trajectory optimization based on inverse models [32, 31].

The new DLR internal library builds upon the Modelica Standard Library [23], and especially the Modelica MultiBody Library [25], as well as the DLR FlexibleBodies Library [12], DLR Visualization Library [1] and the DLR Optimization Library [27]. The SSL combines their capabilities to achieve a wide range of possible applications, ranging from visualization of space missions to high accuracy simulations, optimizations and development of control systems for satellites with flexible structures.

The objective of this paper, apart from introducing the SSL, is to demonstrate its capabilities by modeling a near Earth satellite based on the TET-1 prototype [8], which is part of the FireBird mission – a mission of the German Aerospace Center (DLR) for fire reconnaissance [4, 16]. The modeling of the satellite is described in section 3.

The satellite is modeled with flexible solar panels¹. From the nonlinear (direct) satellite model, an inverse model is derived (sec. 4). This inverse model is used to calculate optimal trajectories for a reorientation maneuver of the satellite using the reaction wheels under constraints (sec. 5) and can also be used as a feed-forward controller.

2 The Space Systems Library

The SSL enables object-oriented, acausal, and equation-based modeling of space systems dynamics and its corresponding environments. This in turn allows controller design and verification, as well as development of path planning and other algorithms.

The Space Systems world model The default world model of the Modelica MultiBody Library [25] is exchanged by a new world model which is compatible with the default world model. It offers additional options and methods for space environment simulations. The world model handles the global simulation time that is used to calculate planet positions and various transformations. The initial time can be given in calendar- or Julian date format. The latter is also implemented internally as time format. The basis coordinate system is chosen to be the Earth Centered Inertial (ECI) coordinate system, which is suitable for near earth satellite simulations. The world model offers a connector and transformation for the Earth Centered Earth Fixed (ECEF) coordinate system, which is useful for the simulation of objects on earth, like emitter stations. The transformation from ECI to ECEF coordinate system is computed as described in [24, 15]. This calculation also considers the difference in seconds between Universal and Universal Coordinated Time (leap seconds, $t_{UT1-UTC}$) that has to be given as an initial value and can be taken from tabular data [37].

Gravity acceleration computation The gravity acceleration $g_0 \in \mathbb{R}^3$ plays a very important role for the simulation of satellites. Hence, multiple gravity models of different complexity were implemented. The most precise model implemented is the EGM96 gravity model [18]. A computational efficient approximation of this model was implemented, which uses terms of up to the second degree of the zonal harmonic coefficients of the gravitational potential. In addition,

moon and sun gravities have been included, considering them as important perturbation factors. These are modeled as point gravity [24]. Although newer, more advanced gravity models exist, the accuracy of the chosen models is sufficient for our multi-body approach that focuses on short term simulations.

Gravity gradient torque The gravity gradient torque is modeled as a torque τ_a that acts on the frame a (position $r_{0,a} \in \mathbb{R}^3$ and orientation $R_a \in \mathbb{R}^{3 \times 3}$. The index a is used to describe a generic frame, which is instantiated for every object) to which it is connected. This frame should be connected to the center of mass of the body in consideration. The torque is caused by the mass distribution of the body in consideration, and depends on the inertia tensor $I \in \mathbb{R}^{3 \times 3}$ as follows.

$$\tau_a = \left(R_a \cdot g_0(r_{0,a}, t_J) \frac{3}{\|r_{0,a}\|} \right) \times \left(I \cdot R_a \frac{-r_{0,a}}{\|r_{0,a}\|} \right) \quad (1)$$

In eq. (1), the gravity acceleration vector $g_0 \in \mathbb{R}^3$ is a function of the position $r_{0,a}$ and the Julian date t_J (to compute planet positions). See [17] for more details.

Solar radiation pressure The effect of the solar radiation pressure p_\odot is modeled as a force element $f_{sp} \in \mathbb{R}^3$ that acts on the element to which it is connected. Shadows of the moon and sun are considered using a cylindrical shadow model. The models are implemented as proposed in [24].

Atmospheric drag The atmospheric drag is caused by friction with the atmosphere depending on the height of the satellite above the Earth. Like the radiation pressure, it is modeled as a force and torque element acting on the attached body which should be located at the center of pressure.

The density $\rho(\lambda, \phi, h, t_J)$ of the atmosphere is computed using the NRLMSISE-00 atmospheric density model [28]. The density ρ depends on the longitude λ , latitude ϕ and height above earth h , that can be computed from $r_{0,a}$, as well as the actual Julian date t_J . The drag force $f_a \in \mathbb{R}^3$ and torque $\tau_a \in \mathbb{R}^3$ can be computed using eq. (2).

$$v_{rel} = \dot{r}_{0,a} - \omega_\oplus \times r_{0,a} \quad (2a)$$

$$f_a = -R_a \frac{1}{2} c_d A_{ad} \rho \|v_{rel}\| v_{rel} \quad (2b)$$

assuming $\|v_{rel}\| \neq 0$ (otherwise $\tau_a = 0$):

$$\tau_a = R_a \frac{1}{2} c_d A_{ad} \rho \|v_{rel}\|^2 \left(\frac{v_{rel}}{\|v_{rel}\|} \times (R_a s_{cp}) \right) \quad (2c)$$

¹The term "flexible satellite" will be used hereafter as a shorthand notation.

In eq. (2), $\omega_{\oplus} \in \mathbb{R}^3$ is the earth angular velocity, c_d is the drag coefficient, A_{ad} is the affected area (which can depend on R_a) and $s_{cp} \in \mathbb{R}^3$ is the vector from the center of pressure to the center of mass, all resolved in the attached frame a .

Geomagnetic field The geomagnetic field can be computed at a desired frame (position $r_{0,a}$) by a geomagnetic field component, using the US/UK World Magnetic Model [22]. The model provides a magnetic field vector $B_m(\lambda, \phi, h, t_J) \in \mathbb{R}^3$ that depends on the longitude λ , latitude ϕ and height above earth h , which can be computed from $r_{0,a}$ and the Julian date t_J .

Variable mass systems The effects of variable mass systems are very important for the modeling of launchers but their consideration can also be necessary for satellites that use gas thrusters, if high precision for the simulation is required. If mass flow $\dot{m} \in \mathbb{R}$ is small, variable mass is usually neglected. Effects of thrusters and jets together with their tanks can be modeled as variable mass systems.

For the SSL, variable mass systems were implemented based on [9, 10] using the concept of a variable mass cylinder with different models of fuel burning. The cylinder represents the fuel or gas tank that is directly attached to the nozzle (of the thruster or jet).

3 The satellite model with flexible structures

The satellite model consists of flexible structures that are modeled as modal bodies as described in [12]. They are based on the definition of Standard Input Data (SID) as defined in [38] as well as rigid bodies and powertrain elements. The equations of motion of each flexible part i are given in eq. (3). The $\tilde{(\cdot)}$ operator is used to generate a skew-symmetric matrix of a vector.

$$\begin{pmatrix} m_i I_3 & & \text{sym.} \\ m_i \tilde{d}_{CM,i}(q_i) & \Theta_i(q_i) & \\ C_{t,i}(q_i) & C_{r,i}(q_i) & M_{e,i} \end{pmatrix} \begin{pmatrix} a_{R,i} \\ \dot{\omega}_{R,i} \\ \dot{q}_i \end{pmatrix} + \begin{pmatrix} 2\tilde{\omega}_{R,i} C_{t,i}^T(q_i) \dot{q}_i + \tilde{\omega}_{R,i} \tilde{\omega}_{R,i} d_{CM,i}(q_i) \\ G_{r,i}(\dot{q}_i) \tilde{\omega}_{R,i} + \tilde{\omega}_{R,i} \Theta_i(q_i) \omega_{R,i} \\ G_{e,i}(\dot{q}_i) \tilde{\omega}_{R,i} + O_{e,i}(q_i) \Omega(\omega_{R,i}) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ K_{e,i} q_i + D_{e,i} \dot{q}_i \end{pmatrix} = \begin{pmatrix} h_{t,i} \\ h_{r,i} \\ h_{e,i} \end{pmatrix} \quad (3)$$

The meaning of the symbols in eq. (3) are listed in table 1. The file of a flexible body (SID) can be ob-

Table 1: SID abbreviations	
$a_R, \omega_R \in \mathbb{R}^3$	acceleration of the ref. frame
$q \in \mathbb{R}^{n_e}$	modal amplitudes
$m \in \mathbb{R}$	body mass
$I_3 \in \mathbb{R}^{3 \times 3}$	identity matrix
$d_{CM}(q) \in \mathbb{R}^3$	position of center of mass
$\Theta(q) \in \mathbb{R}^{3 \times 3}$	inertia tensor
$C_t(q) \in \mathbb{R}^{n_e \times 3}$	inertia coupling mat. (trans.)
$C_r(q) \in \mathbb{R}^{n_e \times 3}$	inertia coupling mat. (rot.)
$k_{\omega}(\omega_R, q, \dot{q}) \in \mathbb{R}^{3+n_e}$	gyro. and centrifugal forces
$\Omega(\omega_R) \in \mathbb{R}^6$	components of ω_R
$G_e(\dot{q}) \in \mathbb{R}^{n_e \times 3}$	gyroscopic matrix (modal)
$G_r(\dot{q}) \in \mathbb{R}^{n_e \times 3}$	gyroscopic matrix (rot.)
$O_e(q) \in \mathbb{R}^{n_e \times 6}$	centrifugal matrix (modal)
$h(q) \in \mathbb{R}^{3+n_e}$	external forces
$M_e \in \mathbb{R}^{n_e \times n_e}$	modal mass matrix
$K_e \in \mathbb{R}^{n_e \times n_e}$	modal stiffness matrix
$D_e \in \mathbb{R}^{n_e \times n_e}$	modal damping matrix

tained from an FEM-analysis (e.g. using ANSYS® or ABAQUS® in combination with SIMPACK®) directly from CAD and material data of the component. In the modal reduction, n_e modes are selected and the required data to calculate eq. (3) is stored in the SID file. The flexible bodies can be combined with other flexible and rigid bodies to model the structural dynamics of the satellite.

The reaction wheels are driven by a motor together with a powertrain including friction. Furthermore, most powertrains used in today's satellites are very stiff. Future satellite designs may incorporate more lightweight constructions with elastic effects in the powertrains, in combination with more powerful and agile motors, as in robotics today. For this reason nonlinear elasticity between the motor and the reaction wheel disk can be taken into account. The elasticity can result from the material of the drive shaft or due to the construction of the coupling. By combining one dimensional models of the powertrain with three dimensional inertia and mass elements, computational efficient models can be designed by using mounting and rotor elements as described in [34]. To model a powertrain in a way which can also be used to generate inverse models is described in [32, 31]. It consists of approximate friction and nonlinear elasticity models that have strictly monotonic characteristics.

Using the components of the library, a detailed mechanical satellite model can be built. The benchmark model used here is a satellite model which is based

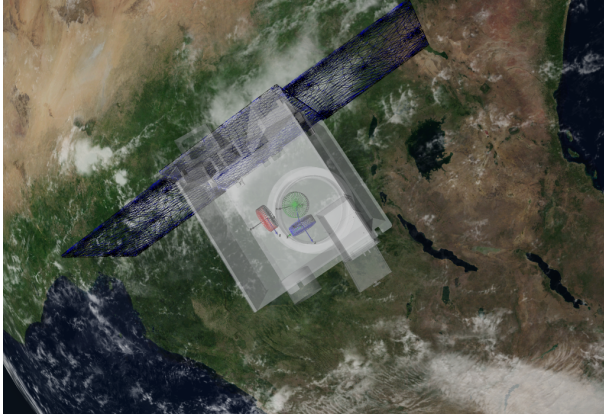


Figure 1: Animation of the satellite with the SSL.

on the TET-1 prototype [8], with slightly exaggerated elasticities, in order to demonstrate the capabilities of the library and future lightweight satellite designs. Three solar panel elements have been modeled as modal bodies. These have been generated from CAD data, a subsequent FEM-analysis, followed by a modal reduction and finally obtaining an SID model. The main body of the satellite is modeled as a rigid body because it is considerably less flexible than the solar panels. Fig. 1 shows an animation of the satellite generated with the SSL. Three reaction wheels mounted at the main satellite body are modeled as rigid bodies representing flywheels. They are connected to one-dimensional (rotational) flexible powertrains with friction and first order motor dynamics. The gyroscopic torques of the one-dimensional powertrain elements are considered in the model by using the mounted rotor elements as described in [34]. The gravity g_0 which is acting on the rigid and flexible bodies is calculated using the gravity model, which is defined by the (global) satellite world model. Additional force and torque elements are connected to the satellite to account for the solar radiation pressure and atmospheric drag as well as the gravity gradient torque, described in sec. 2.

4 Inversion of a structural elastic satellite model

On the top level, a satellite dynamics model has the structure as shown in the top half of fig. 2.

The motor currents I_m are used to drive the reaction wheels of the satellite, from which in turn the rotation of the satellite is determined by the solution of a differential-algebraic equation system. In particular, the angular velocity ω_{ACS} of the central satellite frame

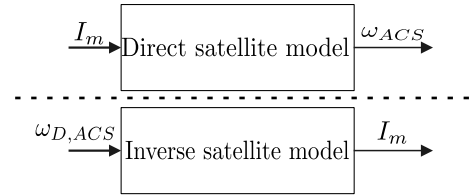


Figure 2: Top level view of direct and inverse satellite model.

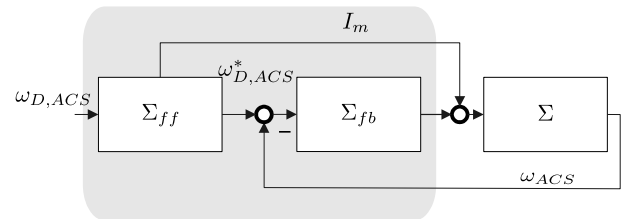


Figure 3: Two degree of freedom control of the satellite motors. With feed-forward controller Σ_{ff} , feedback controller Σ_{fb} and satellite Σ .

is computed, it will be referred to as the ACS-frame (Attitude Control System). This computation is a standard task of satellite simulators.

For advanced control systems, the inverse of this model is needed, as shown in the lower half of fig. 2. The previously given motor currents I_m shall now be computed from the desired angular velocity $\omega_{D,ACS}$. The nonlinear inverse model is based on the satellite model described in section 3. The basis for the inversion of a nonlinear (Modelica) model are index-reduction techniques using the algorithm of Pantelides. The index reduction method [26, 21] allows to choose which equations have to be used to generate a DAE of (at most) index one. All Modelica simulators support this or similar algorithms and automatically perform the inversion. However, both the generation of the inverse model, as well as the numerical computation may fail, if the underlying model does not fulfill certain requirements. Especially, the system must be smoothly continuously differentiable up to the necessary order of differentiation that is determined by the algorithm of Pantelides.

Such a nonlinear inverse model can be, for example, used as feed-forward controller Σ_{ff} for the inner control loop, see fig. 3. An additional outer feedback control loop based on the satellites star tracker would be needed to achieve stationary accuracy of the satellite. For a rigid satellite with rigid powertrains without motor dynamics $\omega_{D,ACS}$ and the equations of motion have to be continuous differentiable at least once so that the equations for the inverse model can be solved via the

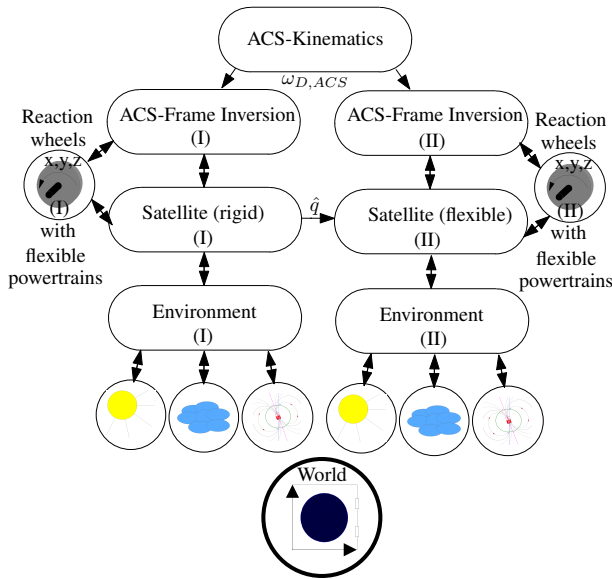


Figure 4: Setup of the approximate inverse satellite model with flexible solar panels and powertrains. The global world model provides the gravity model and simulation time. The environment causes forces and torques on the satellite as described in section 2. Because the environment effects can depend on the states of the satellite, they have to be included for both parallel models of the satellite (I & II).

algorithm of Pantelides. For a satellite with flexible structures and flexible powertrains and first order motor dynamics, as described in section 3, $\omega_{D,ACS}$ as well as the equations of motion have to be differentiable at least three times. In practice, this should be increased by at least one order to avoid non-differentiable motor currents. If these higher order derivatives of $\omega_{D,ACS}$ are not available, a low-pass filter can be used to approximately compute $\omega_{D,ACS}^*$ and its derivatives (see [35, 32, 31]). This is a reason why exact sticking friction cannot be used as part of the inverse model and an approximation has to be used. For components that use non differentiable (e. g. tabular) data like the NRLMSISE-00 atmospheric density model [28] the derivatives have to be either approximated or neglected. This can be achieved in Modelica by using user-defined derivative-functions instead of automatic symbolic differentiation.

It is also necessary that the inverse model is stable. For a linear model this means that the model to be inverted has no (transmission-) zeros in the right half of the complex plane. For the nonlinear model this means that the zero dynamics has to be stable. For highly nonlinear models it is usually very difficult or even impossible to calculate the zero dynam-

ics analytically (e. g. using methods described in [13] that involve solving partial differential equations). Often, the only practical way is to calculate linearizations of the nonlinear model to verify the stability of the (transmission-) zeros.

One typical reason for unstable (transmission-) zeros for systems with flexible structures is the non-collocation of inputs (actuators) and outputs (sensors) of the system. For structural elastic robots this can be the case if motor torques are chosen as output and the robot tip position is chosen as input of an inverse model [32]. The same problem is also possible for flexible satellites, e. g. if the main satellite body attitude should be controlled by actuators that are positioned on a flexible mounting. In general, in order to achieve a stable inverse model of a flexible satellite, approximations must be used, as explained in sec. 4.1, because the exact inverse model is unstable. A first step for the construction of the satellite's inverse model is the definition of a base ACS-frame. The desired angular velocity $\omega_{D,ACS}$ of the satellite will be given with respect to this frame. A good choice for the ACS-frame is either the center of mass of the satellite, or the tip-position of an important on-board instrument of the satellite, e. g. a mounted camera or sensor. This frame is imposed as a root frame according to the Modelica MultiBody Library definition [25]. Hence, the orientation $R_a \in \mathbb{R}^{3 \times 3}$ as well as the angular velocity ω_a of the frame are computed consistently. In addition, the kinematic tree of the satellite is rooted in this frame, although the frame itself can be moving. This is necessary to handle overdetermined DAEs with symbolic transformation techniques in a way that is used in the Modelica MultiBody Library (for further details see the appendix in [25]). The translation of the ACS-frame is not restricted and it moves according to the external forces and the gravity, which act on the bodies connected to it. This results in a hybrid forward/inverse model corresponding to the translation and orientation. The model inversion is done by using the desired angular velocity $\omega_{D,ACS}$ as input for the local angular velocity of the ACS-frame ω_a , starting from a defined angular velocity $\omega_{a,0}$ and orientation. The original input for the forward model I_m , which is the current of the three reaction wheels, is chosen as the new output for the inverse model. The computation of the required angular velocity $\omega_{D,ACS}$, for a specific maneuver, will be described in section 5.

The ACS-frame is connected to the satellite body with its actuators. The desired angular velocity $\omega_{D,ACS}$ of the ACS-frame is achieved by adding a set of equa-

tions to force the angular velocities to the desired values (using Modelica inverse block constraints). This results in three additional equations for the calculation of the torques of the attached torque wheels $\tau_w \in \mathbb{R}^3$. The resulting equations for the ACS-frame model are given in eq. (4).

$$R_a = f_{rot}(\phi_{ACS}) \quad (4a)$$

$$\omega_a = f_{\omega}(\phi_{ACS}, \dot{\phi}_{ACS}) = \omega_{D,ACS} \quad (4b)$$

$$\tau_a = -\tau_w \quad (4c)$$

In eq. (4), f_{rot} and f_{ω} are functions to calculate the transformation matrix R_a and local angular velocity ω_a . Although the values of the Cardan angles ϕ_{ACS} can jump, the internal representation of the orientation is calculated using rotation matrices that remain continuous and show no singularities (see [25] for implementation details). The ACS-frame is forced to move along the desired orientation and the required torques are generated by the attached reaction wheels via τ_w from which the motor currents I_m can be computed by inversion of the elastic powertrain and first order motor dynamics [32, 31]. This is only possible if the reaction wheels are mounted in such a way that they can generate the required torque vector τ_w . Otherwise the equation system will not be solvable (singular). If more than three reaction wheels are used, a torque allocation algorithm has to be implemented in addition.

4.1 Approximation of the deformation of flexible structures of an inverse satellite model

For satellites with flexible structures, depending on the location of the ACS-frame, the exact inverse system can be unstable, and therefore not useful for a control system. This can be checked by calculating the transmission-zeros of the linearized system. If they have a positive real part, the inverse model will be unstable. They can be caused by the combination of flexible structures and the chosen ACS-frame location. In addition, the exact inversion of the equations of motion of flexible structures with weak damping can lead to numerical instability and stiff systems. A solution for this problem is to obtain an approximation of the elastic deformations for the inverse satellite model. Our method is based on a quasi-static approach using two parallel models. A similar method was already successfully implemented for the inversion of flexible robot arms in [32, 31]. Starting points are flexible bodies modeled after eq. (3) from an SID file. In the first

model, quasi-static approximations for the elastic deformation $\hat{q} \in \mathbb{R}^{n_e}$ are calculated. In the second model, these deformations are used as input for the flexible parts and the resulting forces are re-calculated. Figure 4 shows an overview for the setup of the inverse satellite model with flexible solar panels and powertrains. For the first of the two parallel models, the equations of motion for a flexible body are given in eq. (5). The index i for each component is omitted here.

$$\begin{pmatrix} mI_3 & sym. \\ m\tilde{d}_{CM}|_{q=0} & \Theta|_{q=0} \end{pmatrix} \begin{pmatrix} a_R^{(r)} \\ \dot{a}_R^{(r)} \end{pmatrix} \quad (5a)$$

$$+ \begin{pmatrix} \tilde{\omega}_R^{(r)} \tilde{\omega}_R^{(r)} d_{CM}|_{q=0} \\ \tilde{\omega}_R^{(r)} + \tilde{\omega}_R^{(r)} \Theta|_{q=0} \omega_R^{(r)} \end{pmatrix} = \begin{pmatrix} h_t^{(r)} \\ h_r^{(r)} \end{pmatrix} \quad (5b)$$

$$\begin{aligned} h_e^{(r)} &= C_t|_{q=\hat{q}} g_{ref}^{(r)} \\ &+ \sum_j^{n_{node}} \left(\Phi_{m,j}^T|_{q=\hat{q}} f_{ref,j}^{(r)} + \Psi_{m,j}^T|_{q=\hat{q}} \tau_{ref,j}^{(r)} \right) \\ &= C_t|_{q=\hat{q}} a_R^{(r)} + C_r|_{q=\hat{q}} \dot{a}_R^{(r)} + G_e|_{q=\hat{q}} \omega_R^{(r)} \\ &\quad + O_e|_{q=\hat{q}} \Omega(\omega_R^{(r)}) + K_e \hat{q} + D_e \dot{\hat{q}} \end{aligned}$$

The notation $(\cdot)^{(r)}$ denotes variables calculated using the assumption that forces which cause the deformations can be approximated by neglecting elastic deformations (quasi-static). This also leads to a simplification of terms that would normally depend on the elastic deformation q . Therefore, the flexible structural parts are calculated as rigid bodies and approximations for the deformations \hat{q} are calculated using eq. (5) in the first parallel model. To get a stable approximation, terms for the second derivative with respect to time involving $\ddot{\hat{q}}$ are neglected in eq. (5), so that a nonlinear first order differential equation (that is linear in its highest derivative $\dot{\hat{q}}$) is obtained to approximate q in the inverse model.

The approximation \hat{q} is used as input for the second (parallel) inverse satellite model. In the second model, the approximations for \hat{q} are used to recalculate all forces and positions. This is necessary because the resulting deformations change the forces and torques that act on the connected bodies. The equations of motion of the flexible parts in the second model are given

in eq. (6).

$$\begin{pmatrix} mI_3 & m\tilde{d}_{CM}^T|_{q=\hat{q}} & C_t^T|_{q=\hat{q}} \\ m\tilde{d}_{CM}|_{q=\hat{q}} & \Theta|_{q=\hat{q}} & C_r^T|_{q=\hat{q}} \end{pmatrix} \begin{pmatrix} a_R^{(II)} \\ \dot{\omega}_R^{(II)} \\ \ddot{\hat{q}} \end{pmatrix} + \begin{pmatrix} 2\tilde{\omega}_R^{(II)} C_t^T|_{q=\hat{q}}\dot{\hat{q}} + \tilde{\omega}_R^{(II)} \tilde{\omega}_R^{(II)} d_{CM}|_{q=\hat{q}} \\ G_m|_{\dot{q}=\dot{\hat{q}}} \tilde{\omega}_R^{(II)} + \tilde{\omega}_R^{(II)} \Theta|_{q=\hat{q}} \omega_R^{(II)} \end{pmatrix} = \begin{pmatrix} h_t^{(II)} \\ h_r^{(II)} \end{pmatrix} \quad (6)$$

Because the approximation of the modal amplitudes \hat{q} are used directly as input, equations for the modal forces h_e are not necessary in eq. (6). In this equation, the second derivative $\ddot{\hat{q}}$ is again considered. In general, if the flexible parts are connected to each other, it is also possible that the reference accelerations and velocities as well as the external forces change, so they are denoted with the notation $(.)^{(II)}$.

This approximation is valid for small deformations and smooth reorientation maneuvers without extreme external forces and can always be calculated where otherwise no solution could be found (because the exact inverse model would be unstable, depending on the chosen ACS-frame location). If the approximation is sufficient for a specific application can be verified by simulating the approximate inverse model in combination with the original forward model and looking at the resulting error in the calculated torques and deformations. If the resulting error is within the mission tolerance, the approximation can be used. A feedback controller can minimize the remaining error. A typical result for the approximation shows fig. 5, where the first bending mode of an outer solar panel is compared to the resulting first mode of the solar panel without approximation, when using the computed motor currents I_m as feed-forward command (without any feedback controller). The approximation \hat{q}_1 follows q_1 closely, but a residual vibration remains at the end of the movement which results from the (stable) first order approximation for the modal amplitude in eq. (5). This vibration can be further damped by a feedback-controller.

The resulting state x_{inv} , input u_{inv} and the output y_{inv} for the inverse model model setup of fig. 4 is given in eq. (7). The state of the inverse model consists of the states of several subsystems. The states of the ACS-frame x_{ACS} , the three powertrains x_{PT} and the states of the approximation of the modal amplitudes \hat{q}_s for the three solar panel elements. They are modeled with $n_e = 3$ modes for each panel. For the powertrains the flexibility of the connection to the reaction wheels is modeled as spring damper systems (states

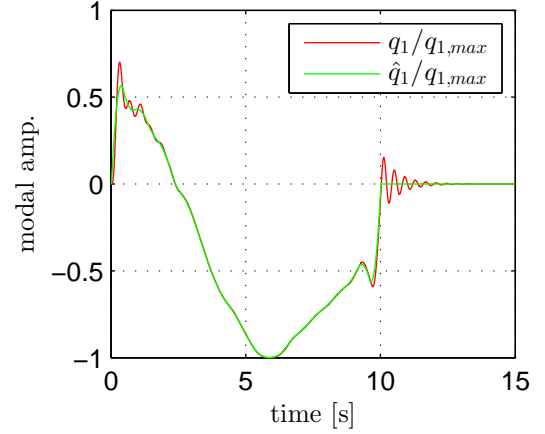


Figure 5: Comparison of the first mode (normalized) between approximation \hat{q}_1 and resulting mode q_1 .

$\phi_{rel,PT}, \dot{\phi}_{rel,PT}$). Because the parallel model setup is used, two sets of states (I & II) are needed for x_{ACS} and x_{PT} .

$$x_{inv} = \left(x_{ACS}^{(I)}, x_{ACS}^{(II)}, x_{PT}^{(I)}, x_{PT}^{(II)}, \hat{q}_s \right)^T \in \mathbb{R}^{51} \quad (7a)$$

$$u_{inv} = \omega_{D,ACS} \in \mathbb{R}^3, y_{inv} = I_m \in \mathbb{R}^3 \quad (7b)$$

with:

$$x_{ACS} = (\phi_{ACS}, r_{0,ACS}, v_{0,ACS}) \quad (7c)$$

$$x_{PT} = (\phi_m, \dot{\phi}_m, \phi_{rel,PT}, \dot{\phi}_{rel,PT}) \quad (7d)$$

If the input for u_{inv} is not differentiable, a filter has to be used and the states of the filter are then also part of x_{inv} . Simulation results showed that the inverse satellite model can be used to accurately calculate the required motor currents and estimate the deformations of the flexible parts using the new approximation method for typical trajectories.

5 Computing optimal trajectories using inverse satellite models

The goal of the attitude trajectory optimization is to find optimal motor currents $I_m \in \mathbb{R}^3$ reorienting the satellite from a starting orientation $Q_{D,0}$ (using a unit quaternion representation $Q_D \in \mathbb{R}^4$) at time t_0 to a desired end orientation $Q_{D,f}$ at time t_f . Using a forward model of the satellite with flexible structures, this op-

timization problem is defined by eq. (8).

$$\min \left\{ \xi_t t_f + \xi_I \sum_{i=1}^3 \left(\int_{t_0}^{t_f} |I_{m,i} \dot{\phi}_{m,i}| dt \right) \right\} \quad (8a)$$

with:

$$F(t, x, \dot{x}, z, u) = 0, x(t_0) = x_0, t \in [t_0, t_f] \quad (8b)$$

$$Q(x_0, z_0, t_0) = Q_{D,0} \quad (8c)$$

constrained by:

$$|\dot{\phi}_m| \leq \dot{\phi}_m^{max} \quad (8d)$$

$$Q(x, z, t) = Q_{D,f} \text{ for } t \geq t_f \quad (8e)$$

with control function:

$$u_c(t) = I_m(t) \in \mathbb{R}^3, |I_{m,i}| \leq I_m^{max} \quad (8f)$$

The chosen criteria allows the optimization of time- or energy-optimal attitude trajectories (or mixtures). The term $F(t, x, \dot{x}, z, u) = 0$ represents the nonlinear satellite model in implicit DAE form where t denotes the time, x the states, u the inputs (unknown motor currents $u = I_m$) and z the algebraic variables of the system. The scalar factors $\xi_t, \xi_I \in \mathbb{R}_+$ are used to weight the criteria for the optimization. The factor ξ_t weights the resulting end time t_f for which the end orientation $Q_{D,f}$ is reached, while ξ_I is a weight for the energy criteria and $\dot{\phi}_m \in \mathbb{R}^3$ are the motor angular velocities for the three motors.

The optimization problem in eq. (8) is difficult to solve directly because of the large solution space for $I_m(t)$ in combination with the strict constraint for the end orientation. By using an inverse satellite model with flexible structures from section 4 and a parameterization of the orientation, the optimization problem can be greatly simplified. The desired orientation is restricted to a path $Q_D(s(t))$ which is calculated by using a spherical linear quaternion interpolation (SLERP, see [6]) along a scalar path parameter $s(t) \in [0, 1]$. This results in an interpolation from the starting orientation $Q_{D,0} = Q_D(s=0)$ to the end orientation $Q_{D,f} = Q_D(s=1)$. The desired angular velocity $\omega_{D,ACS}$ for the ACS-frame which is used as input for the inverse satellite model can be calculated using eq. (9) where ω_0 is the initial angular velocity of the ACS-frame resolved in the ECI-frame and q_i are scalar elements of Q_D .

$$\omega_{D,ACS} = 2 \begin{pmatrix} q_4 & q_3 & -q_2 & -q_1 \\ -q_3 & q_4 & q_1 & -q_2 \\ q_2 & -q_1 & q_4 & -q_3 \end{pmatrix} \dot{Q}_D \quad (9)$$

$$+ 2((q_4 q_4 - 0.5)\omega_0 + ((q_1, q_2, q_3)^T \omega_0)(q_1, q_2, q_3)^T - q_4((q_1, q_2, q_3)^T \times \omega_0))$$

The resulting optimization problem is described in

eq. (10).

$$\min \left\{ \xi_t t_f + \xi_I \sum_{i=1}^3 \left(\int_{t_0}^{t_f} |I_{m,i} \dot{\phi}_{m,i}| dt \right) \right\} \quad (10a)$$

with:

$$F(t, x, \dot{x}, z, u_c) = 0, x(t_0) = x_0, t \in [t_0, t_f] \quad (10b)$$

constrained by:

$$|\dot{\phi}_m| \leq \dot{\phi}_m^{max} \quad (10c)$$

$$|I_{m,i}| \leq I_m^{max} \quad (10d)$$

$$\int_0^{t_f} u_c(t) dt = s(t_f) = 1 \quad (10e)$$

with control function:

$$u_c(t) = \dot{s}(t) \in \mathbb{R}_+ \quad (10f)$$

$$u_c(t_0) = u_c(t_f) = \dot{u}_c(t_0) = \dot{u}_c(t_f) = 0 \quad (10g)$$

To achieve a finite optimization problem, the infinite possibilities for the path parameter $s(t)$, given by the integration over the control function $u_c(t)$, have to be limited by using an appropriate parameterization. This is performed by using a B-spline [7] of order $n_s = 3$ to parameterize $u_c(t)$. Using inverse models in combination with a path parameter $s(t) \in [0, 1]$ inside the optimization offers great advantages over a direct optimization of the motor currents:

- The number of necessary tuners² is much smaller. Instead of having to parametrize all three motor currents $I_m(t)$, only one scalar function $u_c(t)$ has to be parametrized.
- The optimization does not have to stabilize the system like a controller. Using the desired path $Q_D(s)$ as input for the inverse model results in reaching $Q_{D,f}$ exactly.

The equality condition $s(t_f) = 1$ can still be difficult to achieve for optimization algorithms. There are two possible ways to overcome this. One possibility is to avoid the equality condition by using an additional criteria for the optimization in the form of eq. (11).

$$(s(t_f) - 1)^2 \rightarrow \min \quad (11)$$

However, this can lead to an unacceptable error in the end-orientation $Q_{D,f}$. A better way to avoid the equality condition is to directly reshape the B-spline control vector d_c each time the tuner vector is modified by the optimization algorithm. The modified vector is called

²The word tuners is used for the free parameters of an optimization that are changed by the optimizer.

$d_c^+ \in \mathbb{R}^{n_c}$ and is calculated using eq. (12).

$$d_c^+ = \frac{d_c}{\int_0^{t_f} \left(\sum_{i=1}^{n_c} d_{c,i} N_{i,n_s}(t) \right) dt} \quad (12a)$$

$$\Rightarrow s(t_f) = \int_0^{t_f} \left(\sum_{i=1}^{n_c} d_{c,i}^+ N_{i,n_s}(t) \right) dt = 1 \quad (12b)$$

This reshaping leads directly to $s(t_f) = 1$ so that the optimizer does not have to fulfill the equality condition by itself. For time optimal optimizations t_f is also a tuner and the discrete B-spline control points t_i are reshaped to lie in the interval $[0, t_f]$ if t_f is modified by the optimization algorithm.

5.1 Trajectory optimization results

Trajectory optimizations on the basis of an inverse satellite model were performed for a reorientation maneuver of 28° around the axis $e_{rot} = (1, 1, 1)^T$.

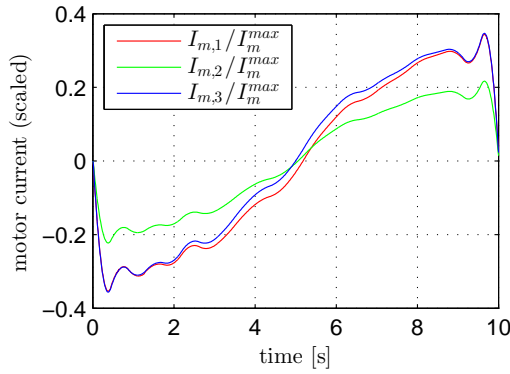


Figure 6: Motor currents for the energy optimal control function.

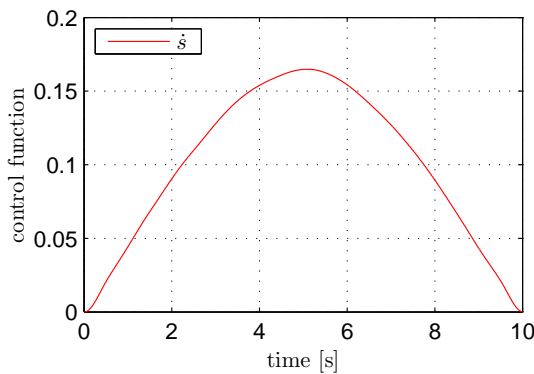


Figure 7: Energy optimal control function for fixed t_f .

All optimization are performed using a B-spline parameterization of the control function with $d_c \in \mathbb{R}^{30}$ as described in the last section. Starting values are found

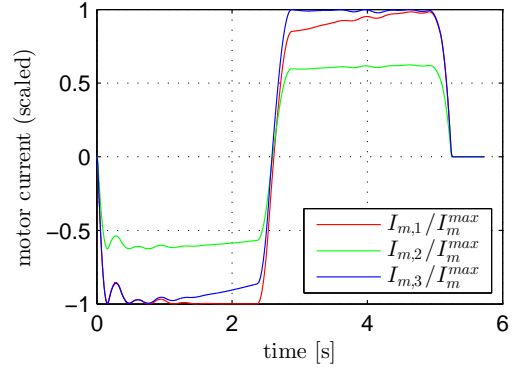


Figure 8: Motor currents for the time optimal control function.

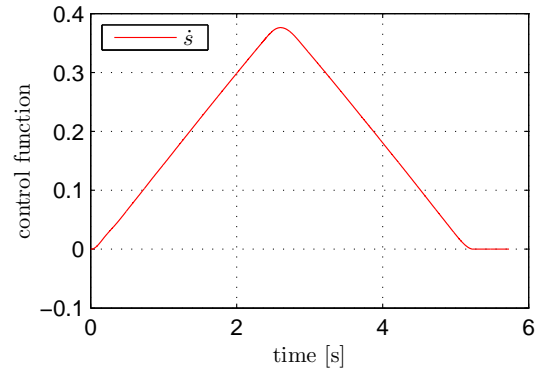


Figure 9: Time optimal control function.

using a global genetic algorithm with a population size of 100 and a rigid inverse satellite model. After good initial parameters have been found, the parameters are refined using a local pattern search algorithm [14, 27] with a complete approximate structural elastic inverse satellite model as described in sec. 3 (including flexible powertrains and flexible solar panels). For both optimization steps the reshaping of the tuners was used as described in the last section.

The first optimization was performed for a fixed end-time $t_f = 10s$. It was the goal to minimize the required energy of this maneuver, according to the criteria given in eq. (10) with $\xi_t = 0$ and $\xi_f = 1$. The plots in fig. 6 and fig. 8 show the resulting optimized motor currents and the corresponding control functions as shown in fig. 9 and fig. 7. The second optimization was performed for the parameters $\xi_t = 1$ and $\xi_f = 0$ in eq. (10) with free end time t_f (as tuner) and results in a time optimal solution for the given path. The resulting trajectories allow to reorient the satellite according to the desired SLERP path. Although there are still oscillations of the flexible components (e. g. solar panels), they are compensated by the reaction wheels

such that the ACS-frame follows the desired path. In addition, limitations on the motor currents and motor velocities are maintained. Small errors that result from the approximations used for the generation of the inverse model can be compensated by a feedback controller (in addition to modeling errors in the case of a real satellite).

6 Conclusion

In this paper, we have presented the Space Systems Library which provides components for satellite/spacecraft nonlinear modeling in Low Earth environment. We demonstrated the library capabilities with a new method for model-based attitude control of a satellite with flexible solar panels.

The method consists on transferring the inverse flexible model approach, successfully implemented in the field of industrial robotics, to the application of flexible satellites modeling and control.

The inverse satellite model allows the computation of the motor currents for a given trajectory of the satellite in a way that the elasticity of the powertrains and flexible solar panels is taken into account and compensated. Using the inverse model, optimal re-orientation maneuvers have been computed for energy and time optimality. The use of inverse satellite models in this optimization greatly improves the optimization process by simplifying the problem considerably.

As part of a two degree of freedom control system, the inverse satellite model can be used as a feed-forward controller to compensate elastic effects and other modeled nonlinear effects that act on the satellite.

7 Acknowledgment

The helpful remarks of M. Otter and A. Pfeiffer for this work are gratefully appreciated.

References

- [1] T. Bellmann. Interactive simulations and advanced visualization with modelica. *Proceedings of the 7th Modelica Conference*, pages 541–550, 2009.
- [2] J. Biesiadecki, A. Jain, and M. James. Advanced simulation environment for autonomous spacecraft. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space, (Tokyo, Japan)*, 1997.
- [3] P. Bodin, M. Nylund, and M. Battelino. SATSIM - A real-time multi-satellite simulator for test and validation in formation flying projects. *Acta Astronautica*, 74:29–39, 2012.
- [4] K. Brieß, W. Bärwald, E. Gill, H. Kayal, O. Montenbruck, S. Montenegro, W. Halle, W. Skrbek, H. Studemund, T. Terzibaschian, and H. Venus. Technology demonstration by the BIRD-mission. *Acta Astronautica*, 56:57–63, 2005.
- [5] J. Carrico, D. Conway, D. Ginn, C. Folta, and K. Richon. Operational use of swingby-an interactive trajectory design and maneuver planning tool - for mission to the moon and beyond. In *Proceedings of the 1995 AAS/AIAA Astrodynamics Specialist (Halifax, Canada)*, 1995.
- [6] E. Dam, M. Koch, and M. Lillholm. Quaternions, interpolation and animation. Technical report, Department of Computer Science, University of Copenhagen, 1998.
- [7] C. de Boor. *A Practical Guide to Splines (Revised Edition)*. Springer-Verlag, 2001.
- [8] S. Eckert, S. Ritzmann, S. Roemer, and W. Bärwald. The TET-1 satellite bus a high reliability bus for earth observation, scientific and technology verification missions in leo. In *The 4S Symposium (Portugal)*, 2010.
- [9] F. Eke. Dynamics of variable mass systems. Technical report, Dep. of Mechanical and Aeronautical Engineering (Univ. of California), 1998.
- [10] F. Eke and T. Mao. On the dynamics of variable mass systems. *International Journal of Mechanical Engineering Education*, 30:123–137, 2000.
- [11] P. Ferguson, T. Yang, M. Tillerson, and J. How. New formation flying testbed for analyzing distributed estimation and control architectures. In *AIAA Guidance, Navigation, and Control Conference and Exhibit (Monterey)*, 2002.
- [12] A. Heckmann, M. Otter, S. Dietz, and J. Lopez. The DLR flexible bodies library to model large motions of beams and of flexible bodies exported from finite element programs. *The Modelica Association*, 2006.
- [13] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag London, 1995.
- [14] H. Joos, J. Bals, G. Looye, K. Schnepper, and A. Varga. A multi-objective optimisation based software environment for control systems design. *Proc. of 2002 IEEE International Conference on Control Applications and International Symposium on Computer Aided Control Systems Design, CCA/CACSD*, 2002.
- [15] G. Kaplan, J. Bangert, J. Bartlet, W. Puatua, and A. Monet. User guide to NOVAS 3.0. *USNO Circular 180*, 2009.
- [16] D. Klumppar, H. Spence, B. Larsen, J. Blake, L. Springer, A. Crew, E. Mosleh, and K. Mashburn. FIREBIRD: A dual satellite mission to examine the spatial and energy coherence scales of radiation belt electron microbursts. In *American Geophysical Union, Fall Meeting*, 2009.
- [17] W. Larson and J. Wertz, editors. *Space Mission Analysis and Design, 3rd edition*. Microcosm, 1999.
- [18] F. Lemoine, S. Kenyon, J. Factor, and R. Trimmer et al. The development of the joint nasa gsfc and nima geopotential model egm96. Technical report, NASA Goddard Space Flight Center, 1998.
- [19] M. Lovera. Object-oriented modelling of spacecraft attitude and orbit dynamics. In *54th International Astronautical Congress, Bremen, Germany*, 2003.
- [20] M. Lovera. Control-oriented modelling and simulation of spacecraft attitude and orbit dynamics. *Journal of Mathematical and Computer Modelling of Dynamical Systems*, 12:73–88, 2006.

- [21] S. Mattsson and G. Söderlind. Index reduction in differential-algebraic equations using dummy derivatives. *SIAM Journal of Scientific and Statistical Computing*, 14:677–692, 1993.
- [22] S. Maus, S. Macmillan, S. McLean, B. Hamilton, A. Thomson, M. Nair, and C. Rollins. The US/UK world magnetic model for 2010-2015. Technical report, NOAA, NESDIS/NGDC, 2010.
- [23] Modelica Association, editor. *Modelica - A Unified Object-Oriented Language for Physical Systems Modeling Language Specification Version 3.2*. 2010.
- [24] O. Montenbruck and E. Gill. *Satellite Orbits - Models, Methods, and Applications*. Springer Verlag, Heidelberg, 2000.
- [25] M. Otter, H. Elmqvist, and S. Mattsson. The new modelica multibody library. *Proceedings of the 3rd International Modelica Conference*, 2003.
- [26] C. Pantelides. The consistent initialization of differential-algebraic systems. *SIAM Journal of Scientific and Statistical Computing*, 9:213–231, 1988.
- [27] A. Pfeiffer. Optimization library for interactive multi-criteria optimization tasks. In *9th International Modelica Conference*, 2012.
- [28] J. Picone, A. Hedin, and D. Dro. NRL-MSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues. *Journal of Geophysical Research*, 2001.
- [29] T. Pulecchi, F. Casella, and M. Lovera. A modelica library for space flight dynamics. In *Proceedings of the 5th International Modelica Conference*, 2006.
- [30] T. Pulecchi, F. Casella, and M. Lovera. Object-oriented modelling for spacecraft dynamics: Tools and applications. *Simulation Modelling Practice and Theory*, 18(1):63 – 86, 2010.
- [31] M. Reiner. *Modellierung und Steuerung eines strukturelastischen Roboters*. PhD thesis, Technische Universität München, 2011.
- [32] M. Reiner, M. Otter, and H. Ulbrich. Modeling and feed-forward control of structural elastic robots. *AIP Conf. Proceedings of the Int. Con. of Numerical and Analysis and Applied Mathematics*, pages 378–381, 2010.
- [33] F. Schiavo and M. Lovera. Modelling, simulation and control of spacecraft with flexible appendages. In *5th International Symposium on Mathematical Modelling, Wien, Austria*, 2006.
- [34] C. Schweiger and M. Otter. Modeling 3D mechanical effects of 1D powertrains. *Proceedings of the third International Modelica Conference*, 2003.
- [35] M. Thümmel, G. Looye, M. Kurze, M. Otter, and J. Bals. Nonlinear inverse models for control. *Proceedings of the 4th International Modelica Conference, Hamburg, March 7-8*, 2005.
- [36] A. Turner. An open-source, extensible spacecraft simulation and modeling environment framework. Master’s thesis, Virginia Polytechnic Institute, 2003.
- [37] US Nautical Almanac Office. *The Astronomical Almanac for the Year 2011*. United Kingdom Hydrographic Office, 2010.
- [38] O. Wallrapp. Standardization of flexible body modeling in multibody system codes, part I: Definition of Standart Input Data. *Mechanical Structures & Machines*, 22(3):283–304, 1994.