

# Sloshing-aware attitude control of impulsively actuated spacecraft

## Tutorial Paper

Pantelis Sopasakis<sup>a</sup>, Daniele Bernardini<sup>a,c</sup>, Hans Strauch<sup>b</sup>, Samir Bennani<sup>d</sup> and Alberto Bemporad<sup>a,c</sup>

**Abstract**—Upper stages of launchers sometimes drift, with the main engine switched-off, for a longer period of time until re-ignition and subsequent payload release. During this period a large amount of propellant is still in the tank and the motion of the fluid (sloshing) has an impact on the attitude of the stage. For the flight phase the classical spring/damper or pendulum models cannot be applied. A more elaborate sloshing-aware model is describe in the paper involving a time-varying inertia tensor.

Using principles of hybrid systems theory we model the minimum impulse bit (MIB) effect, that is, the minimum torque that can be exerted by the thrusters. We design a hybrid model predictive control scheme for the attitude control of a launcher during its long coasting period, aiming at minimising the actuation count of the thrusters.

**Index Terms**—Attitude Control, Upper Stage, Sloshing, Minimum Impulse Thrusters, Hybrid Model Predictive Control, Aerospace.

## I. INTRODUCTION

Upper stages of launchers sometimes have a control mode known as “long coasting phase” which can last up to five hours. The spacecraft, already in orbit with the main engine switched off, drifts with its payloads toward the point on the orbit where the separation shall take place. The stage slowly rotates around its roll axis in order to avoid heating up (*barbecue mode*). The control torques are generated by thrusters. For some types of thrusters the accepted total number of actuation is limited and the long duration of the coasting period makes this problem rather challenging.

Although the spin rate is low (1 to 5deg/sec), the gyroscopic coupling cannot be neglected and the plant dynamics must be treated in a multiple-input/multiple-output (MIMO) fashion [1]. Sometimes the main engine will be re-ignited just prior to the payload release in order to change the orbit parameters. Therefore a considerable amount of propellant is left in the tank (up to 1 to 2 tons). Compared to the dry-mass (in the order of 4 to 6 tons) the torques generated by the propellant motion cannot be neglected (*sloshing phenomenon*).

The classical way to model the fluid motion are pendulum or spring/damper models (see e.g., [2], [3]). Such models are fairly representative if a sufficient acceleration (either from the main engine or in a gravitational field) is present. In a near zero-g environment no constant acceleration is present,

which could generate the restoring force, responsible for the oscillating behaviour of the fluid and the motion of the fluid is only dominated by the surface tension. Unlike these two cases the barbecue mode during the long coasting flight generates a special acceleration environment.

In the case of a cylindrical, centrally placed tank the spin rate generates, via the centrifugal force, a rotational symmetrical acceleration field. In principle, an oscillating behaviour could be expected again, however, a spinning body in free-fall condition will exhibit a motion combining the spinning around the body axis plus a slower rotation of the axis itself (nutation and precision, see [4] and Figure 1). The fluid collects as a bulge and, following the rotating acceleration vector, slowly rotates along the tank wall as it is also reported by Veldman and Vogels [5]. This motion creates large, time-varying off-diagonal elements in the inertia tensor. Computational fluid dynamics (CFD) analyses have been proposed and are best suited to model the sloshing effect, but the difficulty to perform such simulations in real time renders them unsuitable for the purposes of control [6], [7]. In this paper we describe a control-oriented model in analytic form whose parameters are determined offline based on CFD computations.

A noteworthy burgeoning interest in applications of model predictive control (MPC) in aerospace and, in particular, in attitude control can be observed. The use of MPC for attitude control has been proposed by Manikonda *et al.* [8], Vieira *et al.* [9] and other authors. Hegrenæs *et al.* propose an explicit MPC control scheme for attitude control [10], [11]. Other attitude control approaches have been proposed in the literature. Xiao *et al.* [12] study the problem of fault-tolerant attitude control considering the saturation of the actuators. Bang *et al.* [13] propose a sliding-mode controller based on the feedback linearisation of the system’s dynamics and Tournes *et al.* [14] propose a sliding mode controller taking into account the discontinuous nature of the thrusters. Simpler control solutions such as PD and LQR have also been proposed without, however, being able to take consistently into account the constraints that apply on the system [15], [16]. Nonlinear model predictive control approaches for constrained attitude control have been proposed by Kalabic *et al.* [17].

The thrusters of the spacecraft that are used to control its attitude are subject to a *minimum impulse bit*, that is, once activated they will apply a minimum torque to the spacecraft. This effect leads to a hybrid description of the dynamics of the spacecraft and, eventually, to a *hybrid MPC* problem which is formulated as a mixed-integer quadratic problem. Mixed integer programming has been used by Richards

<sup>a</sup> IMT Institute for Advanced Studies Lucca, Piazza San Ponziano 6, 55100 Lucca, Italy. Emails: {pantelis.sopasakis, daniele.bernardini, alberto.bemporad}@imtlucca.it.

<sup>b</sup> Airbus DS, Airbus-Allee 1, 28199 Bremen, Germany. Email: hans.strauch@astrium.eads.net.

<sup>c</sup> ODYS Srl, via della Chiesa XXXII trav. I n. 231, 55100 Lucca, Italy.

<sup>d</sup> European Space Agency (ESA), Keplerlaan 1, Noordwijk, The Netherlands, Email: samir.bennani@esa.int.

*et al.* [18] and Mellinger *et al.* [19] for offline trajectory planning. The proposed approach takes trajectory planning online and applies the control actions in a receding horizon manner.

Recent developments in optimisation theory, enable the design of fast embedded MPC controllers with guaranteed convergence in fixed-point arithmetic [20]. These results have made their appearance in the field of attitude control [21]. At the same time, Frick *et al.* [22] proposed certain heuristics to considerably speed-up the solution of hybrid MPC optimisation problems and yield near-optimal solutions. Evidently, optimisation and control theory offer the tools for the use of MPC in real aerospace applications.

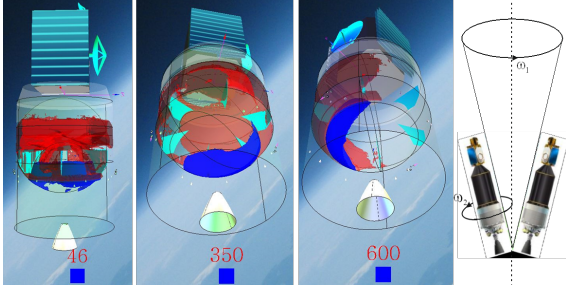


Fig. 1: CFD simulation of upper stage with two cylindrical tanks. Two types of propellant in red and blue. From left to right: (i) end of spin-up phase, started from initially flat propellant distribution, (ii) fluid is collected as a bulge, (iii) the bulge has slowly rotated with the tank, (iv) illustration of spinning and nutation; precession not shown.

The paper is organised as follows: First, in Section III-A we present a novel kinematic model for the spacecraft attitude dynamics involving sloshing and we explain how an extended Kalman filter can be used to reconstruct the system's state in Section III-A. In Section III-C we linearise the system dynamics and in Section III-D we model the impulsive thrusters in a hybrid systems framework. The proposed hybrid model predictive control scheme is presented in Section IV succeeded by simulation results to demonstrate the performance of the closed-loop system.

## II. NOTATION

Let  $\mathbb{N}$ ,  $\mathbb{R}$ ,  $\mathbb{R}^n$ ,  $\mathbb{R}^{m \times n}$  be the sets of natural numbers, real numbers, real  $n$ -vectors and real  $m$ -by- $n$  matrices. Let  $P$  be a logical proposition. We denote by  $[P]$  its *truth value*, i.e.,  $[P] = 1$  if  $P$  is true and  $[P] = 0$  otherwise. We use the notation  $x(t)$  with  $t \geq 0$  for continuous-time signals and  $x_k$  with  $k \in \mathbb{N}$  for discrete-time ones. For any nonnegative integers  $k_1, k_2$  with  $k_1 \leq k_2$ , the finite set  $\{k_1, \dots, k_2\}$  is denoted by  $\mathbb{N}_{[k_1, k_2]}$ .

## III. NONLINEAR KINEMATIC MODEL

In this paper we study the attitude dynamics using a body-fixed (BF) frame which is a right-handed, orthonormal reference frame fixed to the spacecraft so that the  $x$ -axis is aligned to its principal axis and the rotation about it is denoted by  $\Phi$  and is called the *roll angle*. The rotational

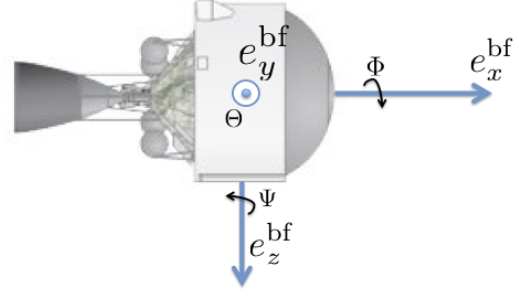


Fig. 2: Body fixed (BF) frame with the  $x$ -axis, defined by  $e_x^{\text{bf}}$ , aligned with the principal axis of the spacecraft.

displacement about the  $y$ -axis defines the *pitch angle*  $\Theta$  and the rotation about the  $z$ -axis is the *yaw angle*  $\Psi$ . The BF body frame is illustrated in Figure 2.

In this section we provide a detailed discussion on the derivation of a dynamical model that captures the upper stage attitude dynamics in light of the additional torques caused by the sloshing of fuel and the impulsive thrusters which are used to control the attitude of the spacecraft.

### A. Sloshing-aware Modelling of Upper Stage Dynamics

In the following the dynamics of motion is derived via the Lagrange formalism (see [23]). The system is modeled as a rigid body (stage plus payloads) and a ring within which a point mass  $m_f$  can rotate (see Figure 3). The parameters  $p$ ,  $r$  and  $\alpha$  are determined via CFD simulations and represent the position of the fluid within the tank for a specific spin rate.

The generalised coordinates  $q$  of the system are the angular rates  $\omega$  and the propellant position  $\alpha$ . The energy  $T$  of the system is can be expressed as:

$$T = \frac{1}{2} \sum_{i,j=1}^3 I_{ij} \omega_i \omega_j + \frac{1}{2} m_f r^2 \dot{\alpha}^2. \quad (1)$$

Solving

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}_i} \right) - \frac{\partial T}{\partial q_i} = 0, \quad (2)$$

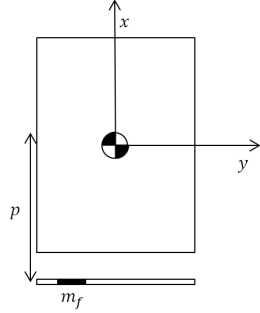
provides the equation of motion, thus, the motion of the mass  $m_f$  is given by

$$\frac{d^2 \alpha}{dt^2} = \kappa \frac{d\alpha}{dt} + \frac{\theta(\alpha, \omega)}{m_f r^2}, \quad (3)$$

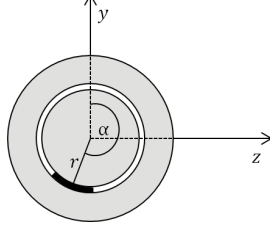
where  $\kappa$  is a constant representing the wall friction and  $\theta$  is given as

$$\begin{aligned} \theta(\alpha, \omega) = & r^2 (\omega_y^2 - \omega_z^2) \sin \alpha \cos \alpha - p r \omega_x \omega_y \sin \alpha \\ & + p r \omega_x \omega_z \cos \alpha - r^2 \omega_y \omega_z \cos(2\alpha). \end{aligned} \quad (4)$$

Equation (4) is the coupling of the main body motion onto the moving mass  $m_f$ , i.e., the excitation caused by the combined motion of precession and nutation. Equation (3) is the sum of the accelerations acting on  $m_f$  and  $\kappa$  must be



(a) Side view of multi-body spacecraft system



(b) Top view of multi-body spacecraft system

Fig. 3: Multibody model for the bulge phenomenon. The values of  $p$  and  $r$  define the circular rotation which the bulge can move (see Fig. 1) and  $\alpha$  describes the current fluid position.

adapted such that relative motion of  $m_f$  resembles the bulge motion computed via CFD. Equation (3) can be written as

$$\frac{d\alpha}{dt} = \beta, \quad (5a)$$

$$\frac{d\beta}{dt} = \kappa\beta + \frac{\theta(\alpha, \omega)}{m_f r^2}. \quad (5b)$$

The inertia tensor  $J$  of the upper stage is a function of the sloshing state  $\alpha$  as follows

$$J(\alpha) = J_0 + J_{m_f}(\alpha), \quad (6)$$

where  $J_0$  is the inertia tensor of the spacecraft without the effect of sloshing (which is a diagonal matrix) and  $J_{m_f}(\alpha)$  is the contribution of the moving mass to the overall inertia given by the symmetric matrix

$$J_{m_f}(\alpha) = m_f \begin{bmatrix} r^2 & pr \cos \alpha & pr \sin \alpha \\ * & p^2 + r^2 \sin^2 \alpha & -\sin \alpha \cos \alpha \\ * & * & p^2 + r^2 \cos^2 \alpha \end{bmatrix}. \quad (7)$$

The torque  $\tau$  given by

$$\tau = \tau_{ext} + \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \cdot l, \quad (8)$$

where  $l = J\omega$  is the angular momentum and  $\tau_{ext}$  is the torque applied by the thrusters. Differentiating  $l$  and by virtue of (8) we have

$$\frac{dl}{dt} = \frac{dJ}{dt}\omega + J\frac{d\omega}{dt} = \frac{dJ}{d\alpha}\frac{d\alpha}{dt}\omega + J\frac{d\omega}{dt}, \quad (9)$$

and given that  $\tau = dl/dt$ , we have that

$$\tau = \frac{dJ}{d\alpha}\frac{d\alpha}{dt}\omega + J\frac{d\omega}{dt}. \quad (10)$$

For convenience let us define

$$\Omega(\omega) = \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix}. \quad (11)$$

The attitude dynamics, by virtue of (8) and (10), is now described by

$$\frac{d\omega}{dt} = J^{-1}\Omega J\omega + J^{-1}\tau_{ext} - J^{-1}\frac{dJ}{d\alpha}\beta\omega. \quad (12)$$

The right-hand side of (12) is a complex function of  $\omega$  mainly because of the involvement of the inverse  $J^{-1}$ ; its derivation in explicit form was carried out using the Symbolic Toolbox of MATLAB 2013a. The pitch and yaw errors, denoted by  $\epsilon_y(t)$  and  $\epsilon_z(t)$  respectively, follow the dynamics:

$$\frac{d\epsilon_y(t)}{dt} = \omega_y(t) + \epsilon_z(t)\omega_x(t), \quad (13a)$$

$$\frac{d\epsilon_z(t)}{dt} = \omega_z(t) - \epsilon_y(t)\omega_x(t). \quad (13b)$$

The attitude dynamics is described by the state vector  $z(t) = (\epsilon_y, \epsilon_z, \omega_x, \omega_y, \omega_z, \alpha, \beta)'$  with input  $u(t) = (\tau_{ext,x}, \tau_{ext,y}, \tau_{ext,z})'$ . The overall dynamics given in equations (5), (12) and (13) can then be written concisely in the form

$$\frac{dz(t)}{dt} = F(z(t), u(t)), \quad (14a)$$

$$y(t) = Cz(t), \quad (14b)$$

where in particular  $F: \mathbb{R}^7 \times \mathbb{R}^3 \rightarrow \mathbb{R}^7$  has the input-affine form  $F(z, u) = f(z) + g(z)u$  and  $C \in \mathbb{R}^{5 \times 7}$  is the matrix  $C = [I_5 \ 0]$ , i.e., the sloshing states  $\alpha$  and  $\beta$  cannot be measured directly in real time.

The above system is discretised with sampling period  $h > 0$  to give:

$$z_{k+1} = z_k + hF(z_k, u_k), \quad (15a)$$

$$y_k = Cz_k, \quad (15b)$$

which will be used as the nominal plant model in the formulation of the state estimation and model predictive control problems in what follows. For convenience we define  $f_h(z, u) = z + hF(z, u)$ .

## B. State Estimation

We employ an extended Kalman filter (EKF) to estimate the state of the discrete-time system (15). The extended Kalman filter is the nonlinear version of the Kalman filter which makes use of the nominal nonlinear system dynamics to predict the evolution of the state while it uses updated local linearisations of the nonlinear system at the current estimated state to estimate the covariance of the state vector [24]. Many authors have also used the EKF approach to reconstruct

the spacecraft attitude parameters [25], [26]. Here, the state estimates  $\hat{z}_k$  are updated according to the nonlinear equation

$$\hat{z}_{k+1} = (I - K_k C f_h(\hat{z}_k, u_k)) + K_k y_k, \quad (16)$$

where  $K_k \in \mathbb{R}^{7 \times 7}$  is determined by

$$K_k = G_k C' (C G_k C' + R)^{-1}, \quad (17a)$$

$$P_{k+1} = (I - K_k C) P_k, \quad (17b)$$

with

$$F_k = \left. \frac{\partial f_h}{\partial z} \right|_{(\hat{z}_k, u_k)}, \quad (17c)$$

$$G_k = F_k P_k F_k' + Q. \quad (17d)$$

The matrix  $Q$  in the above equations is the covariance matrix of a term  $w_k$  acting as a zero-mean additive noise on the system dynamics, that is,  $z_{k+1} = z_k + hF(z_k, u_k) + w_k$ , and  $P_k$  is a covariance estimate for the current state estimate  $\hat{z}_k$ . The matrix  $R$  is the covariance matrix of a zero-mean additive measurement noise  $n_k$ , that is,  $y_k = C z_k + n_k$ . The estimates of the EKF as in (16) are expected to converge to the extent the initial estimate  $\hat{z}_0$  is sufficiently close to the actual initial state.

### C. Linearisation of the model

At time  $k_j$  we linearise the discrete time model (15) around the current estimated state  $\hat{z}_{k_j}$  and input  $u_{k_j}$  to arrive at the following affine dynamical model

$$z_{k+1} = A_{k_j} z_k + B_{k_j} u_k + f_{k_j}, \quad (18)$$

for  $k > k_j$ , where  $A_{k_j}$ ,  $B_{k_j}$  and  $f_{k_j}$  are functions of  $\hat{z}_{k_j}$  and  $u_{k_j}$  as follows

$$A_{k_j} = A(\hat{z}_{k_j}, u_{k_j}) = I + h \left. \frac{\partial F}{\partial z} \right|_{(\hat{z}_{k_j}, u_{k_j})} \quad (19)$$

$$B_{k_j} = B(\hat{z}_{k_j}, u_{k_j}) = h \left. \frac{\partial F}{\partial u} \right|_{(\hat{z}_{k_j}, u_{k_j})} \quad (20)$$

$$f_{k_j} = f(\hat{z}_{k_j}, u_{k_j}) = \hat{z}_{k_j} + F(\hat{z}_{k_j}, u_{k_j}) \quad (21)$$

The linearisation is updated every  $N_l$  sampling periods, that is  $k_j = jN_l$  and the resulting, time-varying, estimated system matrices are given to the MPC controller.

### D. Minimum Impulse Thrusters

In this section we model the hybrid behaviour of the actuators using the *mixed logic and dynamics* framework [27]. The torques exerted by the thrusters are subject to a *minimum impulse bit* (MIB), meaning that, once the thrusters are switched on they cannot be turned off immediately and there is a fixed minimum period of time for which they remain open. The minimum time interval for which the thrusters must stay on implies a minimum exerted torque on the corresponding axis. This can be modelled by non-convex constraints of the form  $u_k \in \mathcal{U}$ , with

$$\mathcal{U} = [-u_{\max}, u_{\min}] \cup \{0\} \cup [u_{\min}, u_{\max}], \quad (22)$$

where  $u_{\min} \in \mathbb{R}^3$  denotes the minimum impulse bit and  $u_{\max} \in \mathbb{R}^3$  denotes the maximum torque that can be

provided in a sampling interval on each axis. In order to translate this constraint to a computationally tractable form, we consider the convex constraints

$$-u_{\max} \leq u_k \leq u_{\max}, \quad (23)$$

introduce the binary vectors  $\delta_k \in \{0, 1\}^3$  and  $\theta_k \in \{0, 1\}^3$  which are written as  $\delta_k = [\delta_k^1, \delta_k^2, \delta_k^3]'$  and  $\theta_k = [\theta_k^1, \theta_k^2, \theta_k^3]'$ , and we establish the correspondence

$$\delta_k^i = [u_i(k) \leq -u_{\min, i}], \quad (24a)$$

$$\theta_k^i = [u_i(k) \geq u_{\min, i}], \quad (24b)$$

for  $i = 1, 2, 3$ . Notice that whenever  $\delta^i = 1$  or  $\theta^i = 1$ , the control action  $u$  is outside the bounds defined by  $\{u \mid -u_{\min} \leq u \leq u_{\min}\}$ , so in light of (23) it can be applied to the system. We define the following propositional logic constraints on the auxiliary continuous variables  $\eta_k \in \mathbb{R}^3$ :

$$(\delta_k^i \vee \theta_k^i) \rightarrow (\eta_{i,k} = u_{i,k}), \quad (25a)$$

$$\neg(\delta_k^i \vee \theta_k^i) \rightarrow (\eta_{i,k} = 0), \quad (25b)$$

for all  $i = 1, 2, 3$  and where  $\vee$  denotes the logical disjunction (OR) operator and  $\neg$  is the negation operator. In addition, we introduce the auxiliary variables  $v_k \in \mathbb{R}^3$  to trace whether at every time instant a thruster activation takes place:

$$v_{i,k} = [\delta_k^i \vee \theta_k^i] \quad (26)$$

for all  $i = 1, 2, 3$ .

By (23), (24) and (25) we have  $\eta_k \in \mathcal{U}$ . Hence, the system dynamics subject to the thrusters constraints can be described by the linear discrete-time model

$$z_{k+1} = A z_k + B \eta_k + f \quad (27a)$$

$$\gamma_{k+1} = \gamma_k + [1 \ 1 \ 1] v_k, \quad (27b)$$

where the additional state variable  $\gamma_k \in \mathbb{R}$ , namely the *activation count*, stands for the number of thrusters activations and, if necessary, can be bounded by the number of maximum activations allowed  $\gamma_{\max}$  according to:

$$\gamma_k \leq \gamma_{\max}. \quad (28)$$

This constraint is likely to become active only if the prediction horizon is long enough to foresee the exhaustion of available actuations or  $\gamma_k$  is close to  $\gamma_{\max}$ . In (27a),  $A$ ,  $B$  and  $f$  are provided by the linearisation step explained in Subsection III-C.

## IV. HYBRID MODEL PREDICTIVE CONTROL

MPC is an optimisation-based control methodology where at each time instant a performance index is optimised using a discrete-time model of the controlled process taking into account the constraints on the state and input variables of the system. This optimisation yields a sequence of control actions whose first element is applied to the system as input while other elements are discarded [28], [29]. As already mentioned, the proposed control scheme aims at (i) steering the pitch and yaw errors and the spin rate to desired set-points while (ii) accounting for the aforementioned constraints and (iii) exhibiting a sparse actuation profile.

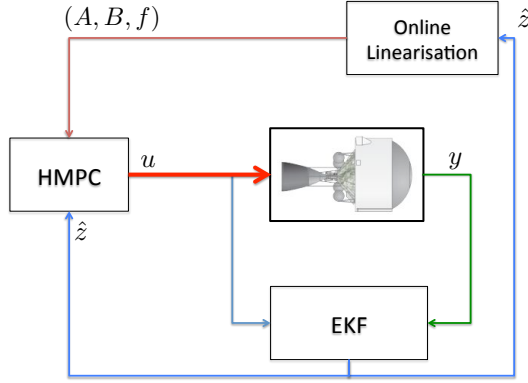


Fig. 4: The proposed control scheme with the hybrid MPC controller, the EKF and the online linearisation.

The model used by MPC here is an affine model of the system obtained by linearisation and the proposed hybrid MPC scheme, illustrated in Figure 4, is formulated as follows:

$$\mathbb{P}(x_0, \gamma_0, A, B, f) : \min_{\pi_N} V_N(\pi_N, \gamma_0) \quad (29a)$$

$$\text{s.t. } x(0) = x_0, \gamma(0) = \gamma_0, \quad (29b)$$

$$\text{Constraints (23) – (28), for } k \in \mathbb{N}_{[0, N_u]}, \quad (29c)$$

$$z_{k+1} = Az_k + B\eta_k + f, \text{ for } k \in \mathbb{N}_{[N_u, N-1]}, \quad (29d)$$

$$-u_{max} \leq \eta_k \leq u_{max}, \text{ for } k \in \mathbb{N}_{[N_u, N-1]} \quad (29e)$$

where  $N_u \leq N$  defines the *hybrid control horizon*, i.e., the number of time steps for which the minimum impulse bit is taken into account as in (29c). Let us denote the set of optimisation variables of the MPC problem by  $\pi_N = \{\{u_k, v_k, \delta_k\}_{k \in \mathbb{N}_{[0, N_u]}}, \{\eta_k\}_{k \in \mathbb{N}_{[0, N-1]}}\}$ , where  $N \geq 1$  is the *prediction horizon* and let  $x_0$  be the current state,  $\gamma_0$  be the total number of activations up to the current time instant  $k$  and let  $A, B$  and  $f$  be estimated linearisation matrices of the system derived as in Section III-C. The MPC control action is computed in a *receding horizon* fashion: At every sampling time instant, optimisation problem (29) is solved to yield the optimal solution  $\pi_N^*$  and the first control action  $\eta_0^*$  is applied to the system. The MPC controller commands admissible torques to the thrusters which will be activated for a certain time between  $t_{min}$  and  $T_s$ , where  $t_{min}$  is the minimum time for which the thrusters can remain open (and corresponds to a  $u_{min}$  torque) and  $T_s$  is the sampling time. The proposed methodology accounts for the hybrid nature of the thrusters and offers a clear advantage over other approaches to attitude control that issue merely on/off commands [30]. In fact, with the proposed approach the sampling time  $T_s$  can be much larger than the minimum impulse time  $t_{min}$ , as a result, the MPC controller can have a greater foresight of the system evolution at a much lower computational cost.

A short hybrid control horizon compared to the prediction horizon is typically employed to reduce the complexity of

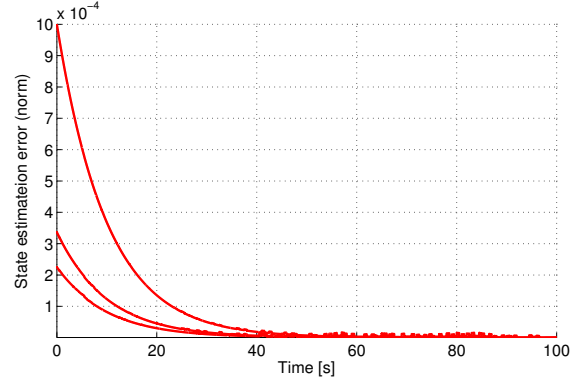


Fig. 5: Norm of the observer error for various initial state estimates  $\hat{z}_0$ .

the resulting optimisation problem. In the proposed formulation (29), the input is assumed to satisfy all constraints given by equations (23)–(28) for all  $k \in \mathbb{N}_{[0, N_u]}$ . For time instants after  $N_u$ , we relax the hybrid constraints and we assume that the input can take any value subject to the constraints (29e) and the system dynamics (29d).

For the chosen sampling time of  $0.5s$ , a large prediction horizon is required for the MPC controller to have enough foresight given that the rotational dynamics of the upper stage are relatively slow. At the same time, the computational complexity can be mitigated by choosing a short hybrid control horizon.

The cost function  $V_N$  in (29) is defined as

$$V_N(\pi_N, \gamma_0) = V_f(z_N, \gamma_N) + \sum_{k=0}^{N-1} \ell(z_k, \eta_k) \quad (30)$$

where  $\ell : \mathbb{R}^7 \times \mathbb{R}^3 \rightarrow \mathbb{R}_+$  is the stage cost

$$\ell(z, u) = \|Qz\|_p + \|Ru\|_p, \quad (31)$$

and  $V_f$  is the terminal cost defined as

$$V_f = \|Q_N z_N\|_p + \rho(\gamma_N - \gamma_0) \quad (32)$$

Matrices  $Q, Q_N, R$  and  $\rho$  are used to strike a desired trade-off between pointing accuracy and usage of thrusters. Any  $p$ -norm ( $1 \leq p \leq \infty$ ) can be used for the stage and terminal cost functions. Here we used  $p = \infty$  so that the resulting problem is a mixed-integer linear problem (MILP).

## V. SIMULATION RESULTS

Firstly we assess the performance of the EKF observer. We define the observer error as  $e_k = z_k - \hat{z}_k$ . The norm of  $e_k$  for different initial state estimates is presented in Figure 5. The observer, tested in closed-loop with the proposed hybrid MPC controller, was found to converge in a small neighbourhood about the actual state of the system.

In order to assess the performance of the closed-loop system using different tuning parameters we introduce certain



TABLE I: Evaluation of the closed-loop performance by simulations over an period  $T_{\text{sim}} = 300\text{s}$ .

$\rho$	$J_{\text{act}}^y$	$J_{\text{act}}^z$	$J_{\text{act}}$	$J_{\text{pa}}^{40}$	$J_{\text{tse}}$
0.005	37	131	168	$1.14 \cdot 10^{-4}$	0.0510
0.05	28	56	84	0.0018	0.0834
0.1	32	48	80	0.0349	0.2988
0.2	34	45	79	0.1279	1.6862

performance indicators. First, the *pointing-accuracy indicator*

$$J_{\text{pa}}^K = \sum_{k=T_{\text{sim}}-K}^{T_{\text{sim}}} \epsilon_{y,k}^2 + \epsilon_{z,k}^2, \quad (33)$$

where  $T_{\text{sim}}$  is the simulation time and  $K$  is the number of time instants before the end of the simulation to be considered. We also introduce the *total squared error indicator*

$$J_{\text{tse}} = \sum_{k=0}^{T_{\text{sim}}} \epsilon_{y,k}^2 + \epsilon_{z,k}^2, \quad (34)$$

the number of thruster actuations along the  $y$  and  $z$  axes, denoted by  $J_{\text{act}}^y$  and  $J_{\text{act}}^z$  and the total actuation count on the  $y$  and  $z$  axes,  $J_{\text{act}}$ . In all cases  $\omega_x$  converges fast to its set-point with few actuations along the  $x$  axis (see for example Figures 7 and 8).

In Table I we summarise the evaluation results of the closed-loop system for different values of  $\rho$  having fixed  $Q = \text{diag}(7, 7, 4, 1, 1, 1, 1)$ ,  $Q_N = 2Q$ ,  $R = 0$  and  $p = \infty$ . The linearisation is updated with  $N_l = 10$ , *i.e.*, every  $5\text{s}$ . The prediction horizon is fixed to  $N = 20$  and  $N_u = 8$ ; this particular choice of the prediction and hybrid control horizon was found to offer a good trade-off between optimality and computational complexity. It is interesting to see that if we decrease the hybrid control horizon to  $N_u = 2$ , the pointing accuracy worsens significantly leading to  $J_{\text{pa}}^{40} = 1.1669$  and  $J_{\text{tse}} = 52.98$  with  $J_{\text{act}} = 144$ . The respective simulations are presented in Figure 9. This important observation justifies the use of hybrid MPC for the control of spacecraft with impulsive thrusters. Moreover, hybrid control horizon values larger than 8 were not found to improve the closed-loop performance, thus, were avoided for the sake of retaining the complexity as low as possible. The average computation time for the derivation of the control action was  $0.12\text{s}$  on a  $2.2\text{GHz}$  Intel Core i7 machine. Simulations were carried out in MATLAB 2013a, using YALMIP [31] and the MILP solver of Gurobi [32].

We can notice that higher values of  $\rho$  lead to a sparser actuation profile reducing the number of  $y$  and  $z$  actuations at the cost of a lower pointing accuracy and overall performance (in terms of  $J_{\text{pa}}$  and  $J_{\text{tse}}$ ). The state trajectory for  $\rho = 0.05$  is illustrated on the  $\epsilon_y$ - $\epsilon_z$ -plane in Figure 6 where we observe that the state moves into a small neighbourhood of the set-point. The HMPC commands are shown in Figure 7 and the spin rate is presented in Figure 8.

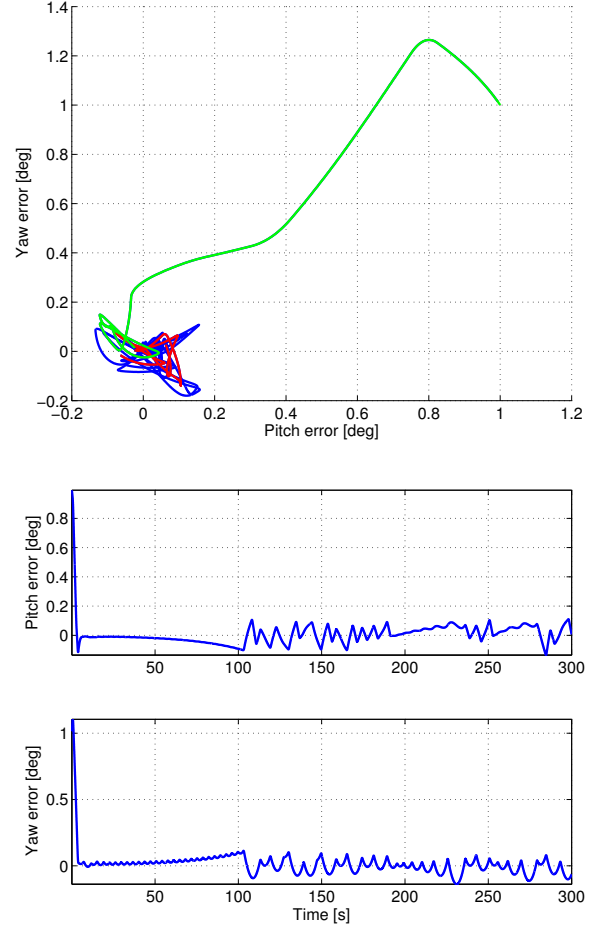


Fig. 6: Controlled trajectory on the  $yz$ -plane (up) and in the time domain (down) with EKF and HMPC in the loop for  $300\text{s}$  with  $\rho = 0.05$ . *Green*: State trajectory for the first  $12\text{s}$  and *Red*: last  $20\text{s}$  of the simulation.

## VI. ACKNOWLEDGMENT

In part, this work has been carried out in connection with ESA's Future Launcher Preparatory Program (FLPP) study "Upper Stage Attitude Control Design Framework" under the lead of Adriana Sirbi.

## REFERENCES

- [1] J. Wertz, ed., *Spacecraft attitude determination and control*. Dordrecht: Kluwer Academic Publishers, 1978.
- [2] A. E. P. Veldman, J. Gerrits, R. Luppens, J. A. Helder, and J. P. B. Vreeburg, "The numerical simulation of liquid sloshing on board spacecraft," *J. Comput. Physics*, vol. 224, no. 1, pp. 82–99, 2007.
- [3] R. Ibrahim, *Liquid sloshing dynamics*. Cambridge: Cambridge university press, 2005.
- [4] P. Hughes, *Spacecraft attitude dynamics*. Dover Publications Inc., 1986.
- [5] A. Veldman and M. Vogels, "Axisymmetric liquid sloshing under low-gravity conditions," *Acta Astronautica*, vol. 11, no. 10–11, pp. 641 – 649, 1984.
- [6] M. Lazzarin, M. Biolo, A. Bettella, M. Manente, R. D. Forno, and D. Pavarin, "EUCLID satellite: Sloshing model development through computational fluid dynamics," *Aerospace Science and Technology*, vol. 36, pp. 44 – 54, 2014.

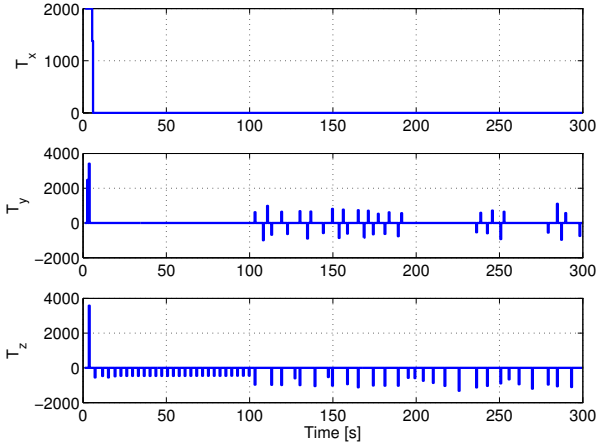


Fig. 7: Torques applied by the impulsive thrusters ( $\rho = 0.05$ ).

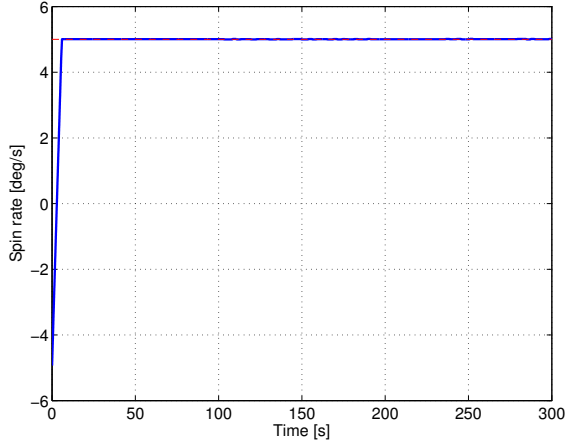


Fig. 8: Spacecraft in barbecue mode: the spin rate  $\omega_x$  tracks the desired set-point.

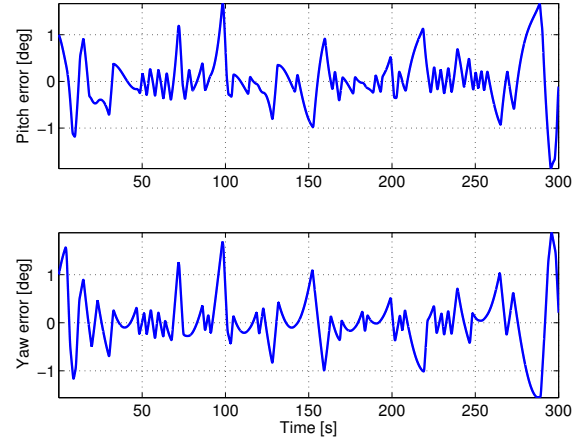


Fig. 9: Closed-loop trajectory with  $N = 20$  and  $N_u = 2$ .

- [7] J. Hervas and M. Reyhanoglu, "Thrust-vector control of a three-axis stabilized upper-stage rocket with fuel slosh dynamics," *Acta Astronautica*, vol. 98, no. 0, pp. 120 – 127, 2014.
- [8] V. Manikonda, P. Arambel, M. Gopinathan, R. Mehra, and F. Hadaegh, "A model predictive control-based approach for spacecraft formation keeping and attitude control," in *American Control Conference, 1999. Proceedings of the 1999*, vol. 6, pp. 4258–4262, 1999.
- [9] M. Vieira, R. Galvao, and K. Heinz Kienitz, "Attitude stabilization with actuators subject to switching-time constraints using explicit MPC," in *2011 IEEE Aerospace Conference*, pp. 1–8, March 2011.
- [10] Ø. Hegrenæs, J. Gravdahl, and P. Tondel, "Attitude control by means of explicit model predictive control, via multi-parametric quadratic programming," in *American Control Conference, 2005. Proceedings of the 2005*, pp. 901–906 vol. 2, June 2005.
- [11] Ø. Hegrenæs, J. Gravdahl, and P. Tondel, "Spacecraft attitude control using explicit model predictive control," *Automatica*, vol. 41, no. 12, pp. 2107 – 2114, 2005.
- [12] B. Xiao, Q. Hu, and P. Shi, "Attitude stabilization of spacecrafts under actuator saturation and partial loss of control effectiveness," *Control Systems Technology, IEEE Transactions on*, vol. 21, pp. 2251–2263, Nov 2013.
- [13] H. Bang, J.-S. Lee, and Y.-J. Eun, "Nonlinear attitude control for a rigid spacecraft by feedback linearization," *KSME International Journal*, vol. 18, no. 2, pp. 203–210, 2004.
- [14] C. Tournes, Y. Shtessel, and E. Wells, "Upper stage rocket guidance and control using discontinuous reaction control thrusters via sliding modes," in *American Control Conference, 1997. Proceedings of the 1997*, vol. 4, pp. 2547–2551 vol.4, Jun 1997.
- [15] A. Kater, "Attitude Control of Upper Stage Launcher During Long Coasting Period," Master's thesis, Lehrstuhl für Automatisierungs- und Regelungstechnik, Christian-Albrechts-University Kiel, Germany, 2013.
- [16] S. Beatty, "Comparison of pd and lqr methods for spacecraft attitude control using star trackers," in *WAC '06. World Automation Congress*, pp. 1–6, July 2006.
- [17] U. Kalabic, R. Gupta, S. Di Cairano, A. Bloch, and I. Kolmanovsky, "Constrained spacecraft attitude control on  $SO(3)$  using reference governors and nonlinear model predictive control," in *American Control Conference (ACC)*, pp. 5586 – 5593, IEEE, June 2014.
- [18] A. Richards, J. How, T. Schouwenaars, and E. Feron, "Plume avoidance maneuver planning using mixed integer linear programming," in *Proc. of AIAA Guidance, Navigation, and Control Conf.*, (Montreal, Canada), Aug. 2001.
- [19] D. Mellinger, A. Kushleyev, and V. Kumar, "Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 477–483, May 2012.
- [20] P. Patrinos, A. Guiggiani, and A. Bemporad, "Fixed-point dual gradient projection for embedded model predictive control," in *Proc. 12th European Control Conference*, (Zurich, Switzerland), 2013.
- [21] A. Guiggiani, I. Kolmanovsky, P. Patrinos, and A. Bemporad, "Fixed-point constrained model predictive control of spacecraft attitude," arXiv:1411.0479, 2014.
- [22] D. Frick, A. Domahidi, and M. Morari, "Embedded Optimization for Mixed Logical Dynamical Systems," *Computers & Chemical Engineering*, July 2014. to appear.
- [23] D. Wells, *Lagrangian Dynamics*. McGraw-Hill, 1967.
- [24] P. Costa, "Adaptive model architecture and extended Kalman-Bucy filters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 30, pp. 525–533, 1994.
- [25] A. Weiss, I. Kolmanovsky, M. Baldwin, and R. S. Erwin, "Model predictive control of three dimensional spacecraft relative motion," in *Proceedings of the American Control Conference*, (Montreal, Canada), pp. 173–178, June 2012.
- [26] A. S. Mohammed, M. Benyettou, A. Boudjemai, Y. Hashida, and M. Sweeting, "Simulation of microsatellite attitude using kalman filtering in orbit results," *Simulation Modelling Practice and Theory*, vol. 16, no. 3, pp. 257 – 277, 2008.
- [27] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407 – 427, 1999.
- [28] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009.
- [29] E. Camacho and C. Bordons, *Model predictive control*. London: Springer-Verlag, 2004.
- [30] M. Leomanni, A. Garulli, A. Giannitrapani, and F. Scortecci, "An MPC-based attitude control system for all-electric spacecraft with

on/off actuators,” in *Proc. 52th IEEE Conf. on Decision and Control*, (Florence, Italy), pp. 4853–4858, 2013.

- [31] J. Löfberg, “YALMIP : a toolbox for modeling and optimization in matlab,” in *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pp. 284–289, Sept 2004.
- [32] Gurobi Optimization, Inc., “Gurobi optimizer reference manual,” 2014. Online at <http://www.gurobi.com>.