

Spacecraft Attitude Control using Explicit Model Predictive Control [★]

Øyvind Hegrenæs ^{a,b}, Jan Tommy Gravdahl ^a, Petter Tøndel ^a

^aDepartment of Engineering Cybernetics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway

^bUniversity Graduate Center at Kjeller, N-2027 Kjeller, Norway

Abstract

In this paper, an explicit model predictive controller for the attitude of a satellite is designed. Explicit solutions to constrained linear MPC problems can be computed by solving multi-parametric quadratic programs (mpQP), where the parameters are the components of the state vector. The solution to the mpQP is a piecewise affine (PWA) function, which can be evaluated at each sample to obtain the optimal control law. The on-line computation effort is restricted to a table-lookup, and the controller can be implemented on inexpensive hardware as fixed-point arithmetics can be used. This is useful for systems with limited power and CPU resources. An example of such systems is micro-satellites, which is the focus of this paper. In particular, the explicit MPC (eMPC) approach is applied to the SSETI/ESEO micro-satellite, initiated by the European Space Agency (esa). The theoretical results are supported by simulations.

Key words: Attitude control; Predictive control; Constraints; Linear quadratic regulators; Piecewise linear controllers.

1 Introduction

The purpose of this paper is twofold. First, we establish a nonlinear model of a micro-satellite, with thrusters and a reaction wheel as actuators. Secondly, we propose a strategy to solve the attitude control problem for this satellite. However, unlike earlier work, typically carried out using PD- or LQ-control [11],[14], Lyapunov design procedures [6], sliding mode [2], quaternion feedback techniques [10],[15], or \mathcal{H}_∞ [3],[12], the focus of this paper will be on explicit model predictive control. This approach should be considered if constraints need to be taken into account, and real-time optimization is impossible due to computational limitations. To the best knowledge of the authors, this approach has not yet been applied to attitude control of spacecrafts.

Stability proofs are not considered at this point. A potential approach is to search for piecewise quadratic Lyapunov functions by solving a convex optimization prob-

lem. In [5] this was done using linear matrix inequalities (LMIs).

When implementing the controller, it is important to keep in mind that the actuating thrusters are on-off by nature. A bang-bang modulation scheme with dead-zone will be utilized to address this problem.

The structural data and satellite model is based on the SSETI/ESEO micro-satellite, initiated by esa, and the results in this paper are based on the work in [7].

1.1 Explicit Model Predictive Control

When solving an MPC problem the control action, or equally, the solution, is obtained by computing an open-loop optimal sequence of control inputs on a predefined horizon, once for each time sample. The first control input in the sequence is then applied to the plant, and the optimization is repeated with the new initial conditions and on the new horizon, shifted one step ahead. Due to the shifted horizon, the term *receding horizon control* is commonly used interchangeably with MPC. For the remainder of this section, the process to be controlled can be described by a *discrete-time*, deterministic linear

[★] This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by -X- under the direction of -X-. Corresponding author Ø. Hegrenæs. Tel. +47 64844780. Fax +47 63818146.

Email addresses: hegrenas@unik.no (Øyvind Hegrenæs), tommy.gravdahl@itk.ntnu.no (Jan Tommy Gravdahl), petter.tondel@itk.ntnu.no (Petter Tøndel).

state space model, that is

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k),\end{aligned}\quad (1)$$

where $\mathbf{x}(k) \in \mathbb{R}^n$ is the state variable, $\mathbf{u}(k) \in \mathbb{R}^m$ is the input variable, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{m \times m}$, and (\mathbf{A}, \mathbf{B}) is a stabilizable pair. If we now consider the regulator problem, that is, the problem of driving the state vector to the origin, the traditional MPC solves the following optimization problem for the current $\mathbf{x}(k)$

$$\begin{aligned}\min_{\mathbf{U}, \mathbf{s}} \{ \mathcal{J}(\mathbf{U}, \mathbf{s}, \mathbf{x}(k)) \} \quad \text{subject to:} \\ \mathbf{y}_{\min} - \mathbf{s} \leq \mathbf{y}_{k+i|k} \leq \mathbf{y}_{\max} + \mathbf{s}, \quad i = 1, \dots, N \\ \mathbf{u}_{\min} \leq \mathbf{u}_{k+i} \leq \mathbf{u}_{\max}, \quad i = 0, \dots, M-1 \\ \mathbf{u}_{k+i} = \mathbf{K}\mathbf{x}_{k+i|k}, \quad M \leq i \leq N-1 \\ \mathbf{x}_{k|k} = \mathbf{x}(k) \\ \mathbf{x}_{k+i+1|k} = \mathbf{A}\mathbf{x}_{k+i|k} + \mathbf{B}\mathbf{u}_{k+i}, \quad i \geq 0 \\ \mathbf{y}_{k+i} = \mathbf{C}\mathbf{x}_{k+i|k}, \quad i \geq 0\end{aligned}\quad (2)$$

where the cost function to be minimized is given as

$$\begin{aligned}\mathcal{J} &= \rho \|\mathbf{s}\|_2^2 + \mathbf{x}_{k+N|k}^T \mathbf{P} \mathbf{x}_{k+N|k} \\ &+ \sum_{i=0}^{N-1} \left\{ \mathbf{x}_{k+i|k}^T \mathbf{Q} \mathbf{x}_{k+i|k} + \mathbf{u}_{k+i}^T \mathbf{R} \mathbf{u}_{k+i} \right\},\end{aligned}\quad (3)$$

and $\mathbf{U} \triangleq [\mathbf{u}_k^T, \dots, \mathbf{u}_{k+M-1}^T]^T$, $\mathbf{s} \triangleq [\mathbf{s}_k^T, \dots, \mathbf{s}_{k+N-1}^T]^T$, $\mathbf{R} = \mathbf{R}^T > 0$, $\mathbf{Q} = \mathbf{Q}^T \geq 0$, $\mathbf{P} = \mathbf{P}^T > 0$, $\mathbf{x}_{k+i|k}$ is the prediction of $\mathbf{x}(k+i)$ at time k , and M and N are input and constraint horizons. When the final cost matrix \mathbf{P} and gain matrix \mathbf{K} are calculated from the algebraic Riccati equation, under the assumptions that the constraints are not active for $i \geq M$ and $i \geq N$, (2) exactly solves the constrained infinite horizon LQR problem for (1), with weight matrices \mathbf{R} and \mathbf{Q} . The additional variable $\mathbf{s} \in \mathbb{R}^{n_s}$ is a vector containing slack variables, while the term $\|\mathbf{s}\|_2$ is the \mathcal{L}_2 -norm of \mathbf{s} , and $\rho > 0$ is the penalty weight of the slack variables. Note that using the \mathcal{L}_2 -norm is only one way of including slack variables. The slack variables are defined such that they are nonzero only if the output constraints are violated, yet heavily penalized in the cost function, so that the optimizer has a strong incentive to keep them zero if possible. If we have $\rho \rightarrow \infty$, or equally $\|\mathbf{s}\|_2 \rightarrow 0$, the MPC problem in (2) involves only hard constraints. Hard constraints may imply infeasibility, which for instance can be the case if initial conditions are infeasible, if noise causes the output to go outside the feasible solution space in the next time step, or if there are serious model uncertainties. This needs to be addressed in real applications, and the introduction of slack variables is one possibility.

1.1.1 From linear MPC to mpQP

It is shown in [1], with $\rho \rightarrow \infty$, that the MPC problem (2) can be reformulated as

$$\begin{aligned}V_z(\mathbf{x}(k)) &= \min_{\mathbf{z}} \left\{ \frac{1}{2} \mathbf{z}^T \mathbf{H} \mathbf{z} \right\} \\ \text{subject to:} \quad \mathbf{G} \mathbf{z} &\leq \mathbf{W} + \mathbf{S} \mathbf{x}(k),\end{aligned}\quad (4)$$

where $\mathbf{z} \triangleq \mathbf{U} + \mathbf{H}^{-1} \mathbf{F}^T \mathbf{x}(k)$, \mathbf{U} is defined as in (2), and $\mathbf{x}(k)$ is the current state, which can be treated as a vector of parameters. We have that $\mathbf{z} \in \mathbb{R}^{n_z}$, $\mathbf{H} \in \mathbb{R}^{n_z \times n_z}$, $\mathbf{G} \in \mathbb{R}^{q \times n_z}$, $\mathbf{W} \in \mathbb{R}^{q \times 1}$, and $\mathbf{S} \in \mathbb{R}^{q \times n}$. Note that $\mathbf{H} > 0$ since $\mathbf{R} > 0$. This is a strong result, as the problem formulated in (4) is strictly convex, and the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient conditions for optimality, giving a unique solution.

As shown in [1], the mpQP in (4) can be solved by applying the KKT conditions

$$\begin{aligned}\mathbf{H} \mathbf{z} + \mathbf{G}^T \lambda &= 0, \quad \lambda \in \mathbb{R}^q, \\ \lambda_r (\mathbf{G}^r \mathbf{z} - \mathbf{W}^r - \mathbf{S}^r \mathbf{x}(k)) &= 0, \quad r = 1, \dots, q, \\ \lambda &\geq 0, \\ \mathbf{G} \mathbf{z} - \mathbf{W} - \mathbf{S} \mathbf{x}(k) &\leq 0,\end{aligned}\quad (5)$$

where the superscript r on a matrix denotes the r^{th} row, while q is the number of inequalities in the optimization problem. The number of free variables is $n_z = mN$.

Even though not derived for the case of including slack variables, both (4) and (5) can easily be extended to cover this situation, by defining the augmented matrices $\tilde{\mathbf{H}} \in \mathbb{R}^{\tilde{n}_z \times \tilde{n}_z}$, $\tilde{\mathbf{G}} \in \mathbb{R}^{q \times \tilde{n}_z}$, and $\tilde{\mathbf{z}} \triangleq [\mathbf{z}, \mathbf{s}]^T \in \mathbb{R}^{\tilde{n}_z}$. The number of free variables now becomes $\tilde{n}_z = n_z + n_s$.

A key observation is that (4) is solved explicitly for *all* $\mathbf{x}(k)$. It is shown in [1] that the solution $\mathbf{z}^*(\mathbf{x}(k))$, hence $\mathbf{U}^*(\mathbf{x}(k))$, is a continuous piecewise affine (PWA) function defined over a polyhedral partition. Consequently, the on-line effort is limited to evaluating this PWA function. This is illustrated in the following example.

1.1.2 Explicit MPC example

A simple example is given to illustrate some features of explicit MPC. Similar analysis is applicable for the satellite system to be studied subsequently, but visualization becomes harder in this case. For the remainder of this paper, the explicit MPC controllers are derived using the algorithms in [13].

Consider the double integrator [9],

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d u(k),\quad (6)$$

where

$$\mathbf{A}_d = \begin{bmatrix} 0 & T_s \\ 0 & 1 \end{bmatrix}, \quad \mathbf{B}_d = \begin{bmatrix} T_s^2 \\ T_s \end{bmatrix}, \quad \mathbf{x}(k) \triangleq \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}. \quad (7)$$

If x_1 is interpreted as position, x_2 as speed and u as force, the objective is to control the position under constraints on the speed and force, which we define as $|x_2| \leq 0.5$ and $|u| \leq 1$. By setting the sampling time T_s to 0.05 [sec], and excluding slack variables, the mpQP problem for (6), over the horizon $N = M = 2$, with cost matrices $R = 1$ and $\mathbf{Q} = \text{diag}(1, 0)$, gives a polyhedral partition over the parameter space $-[3, 0.6]^T \leq \mathbf{x} \leq [3, 0.6]^T$, consisting of 13 regions. The parameter space is the space in which we solve the mpQP. If we denote each of these polyhedra as \mathcal{X}_i , where i is the specific region, then $\mathcal{X}_i \subset \mathbb{R}^2$. Each polyhedron contains an optimal state feedback control law such that if $\mathbf{x}(k) \in \mathcal{X}_i$ then

$$\mathbf{u}(k) = \mathbf{K}_i \mathbf{x}(k) + \mathbf{k}_i, \quad (8)$$

where \mathbf{K}_i is the gain matrix for region i , and \mathbf{k}_i is a constant vector. Consequently, once the mpQP has been solved, finding the optimal control law is restricted to a table-lookup, depending on the current state vector. For (6), with initial conditions $\mathbf{x}(0) = [-2, 0]^T$, we get the response shown in Fig. 1, which clearly shows the relation between the trajectory in state space and the search for an optimal control law in the parameter space.

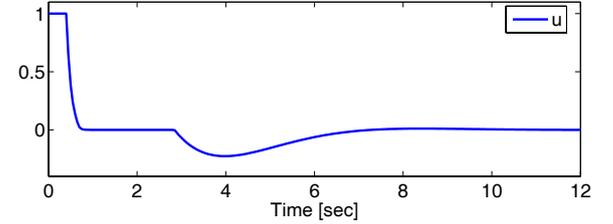
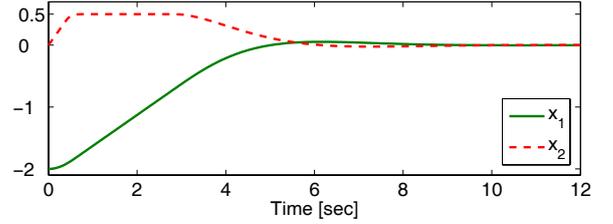
In the case where no constraints are active ($i = 1$), the explicit MPC solution equals that of the stationary discrete LQR, that is, $\mathbf{K}_1 \equiv -\mathbf{K}_{LQ}$ and $\mathbf{k}_1 \equiv \mathbf{0}$.

1.2 SSETI/ESEO

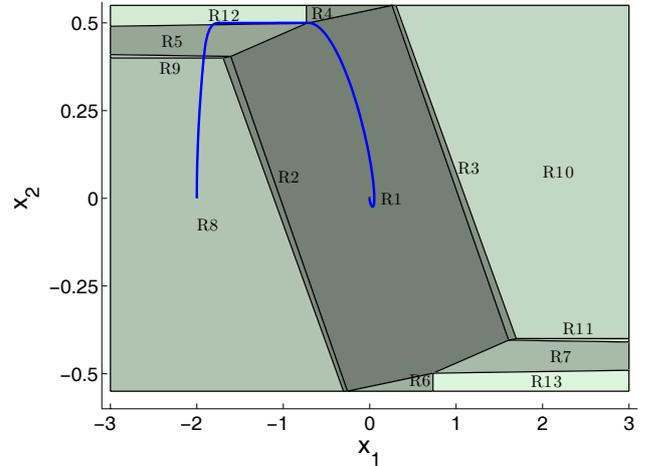
The Student Space Exploration and Technology Initiative (SSETI) comprises several satellite projects. The specific satellite to be studied in this paper is the European Student Earth Orbiter (ESEO). Through the project, students from different European universities participate in designing, building and operating a micro-satellite. A short summary of substantial SSETI/ESEO data is given in Table 1.

Table 1
SSETI/ESEO parameters

Parameter	Value
Satellite inertia matrix, \mathbf{I}	$\text{diag}(4.250, 4.337, 3.664)$ [kgm ²]
Axial wheel inertia, I_w	$4 \cdot 10^{-5}$ [kgm ²]
Axial wheel placement, \mathbf{A}	$[0, 1, 0]^T$
Thruster torque, \mathbf{K}_{nom}	$[0.0484, 0.0484, 0.0398]^T$ [Nm]
Maximum wheel torque	0.0020 [Nm]
Maximum wheel velocity	527 [rad/s] \approx 5032 rpm



(a) Simulation in state space



(b) Simulation in parameter space

Fig. 1. Set-point control of the double integrator, $N = 2$

The attitude control system for ESEO comprises several subsystems, including attitude determination and propulsion. For propulsion, a cold gas system using gaseous nitrogen (GN_2) was chosen since it is simple and affordable, and it can be assembled by students without too strict safety regulations. A reaction wheel is also included for actuation, by means of providing torque about the principal body y-axis. Attitude determination is done using an extended Kalman filter, where available measurements are provided by sun and horizon sensors, magnetometers, and a star tracker. For the remainder we will assume that the attitude is known.

The performance requirements for the attitude control system are closely related to various mission modes. In this paper only the nominal mode is considered, which refers to general spacecraft operations when in orbit. This will be apparent in later sections. In nominal operation, the pointing error is not to exceed $\pm 1^\circ$ about the principal x-axis (roll) and y-axis (pitch), while pointing the z-axis (yaw) within $\pm 5^\circ$ of desired value.

2 Modeling

In this section, a model of a satellite with thrusters and an L -wheel cluster is derived. The notation is based on [4] and [8].

2.1 Kinematics

Due to their nonsingular parametrization, the Euler parameters are chosen to represent the kinematics. The Euler parameters are defined in terms of the angle axis parameters θ and \mathbf{k} , and the mapping is defined as

$$\eta = \cos \frac{\theta}{2}, \quad \boldsymbol{\epsilon} = \mathbf{k} \sin \frac{\theta}{2} \quad (9)$$

which gives the corresponding rotation matrix

$$\mathbf{R}(\eta, \boldsymbol{\epsilon}) = \mathbf{1} + 2\eta\boldsymbol{\epsilon}^\times + 2\boldsymbol{\epsilon}^\times\boldsymbol{\epsilon}^\times, \quad (10)$$

where $(\cdot)^\times$ denotes the skew-symmetric, or cross product, operator. From the properties of $\mathbf{R} \in SO(3)$, it can be shown that

$$\dot{\mathbf{R}}_o^b = (\boldsymbol{\omega}_{bo}^b)^\times \mathbf{R}_o^b = -(\boldsymbol{\omega}_{ob}^b)^\times \mathbf{R}_o^b, \quad (11)$$

where $\boldsymbol{\omega}_{ob}^b$ is defined as the angular velocity of the body frame \mathcal{F}_b relative to the orbit frame \mathcal{F}_o , measured in \mathcal{F}_b , and \mathbf{R}_o^b is the rotation matrix from \mathcal{F}_b to \mathcal{F}_o . The orbit frame has its origin located at the center of mass of the satellite. Its z-axis is always nadir pointing (towards the center of Earth), while its x-axis is pointing in the direction of the forward velocity. The y-axis completes a right-hand coordinate system. From (10) and (11), the kinematic differential equations can be found as

$$\dot{\eta} = -\frac{1}{2}\boldsymbol{\epsilon}^T \boldsymbol{\omega}_{ob}^b \quad (12a)$$

$$\dot{\boldsymbol{\epsilon}} = \frac{1}{2}[\eta\mathbf{1} + \boldsymbol{\epsilon}^\times] \boldsymbol{\omega}_{ob}^b. \quad (12b)$$

2.2 Dynamics

The equations of motion for an L -wheel gyrost are written as

$$\dot{\mathbf{h}}_b = \boldsymbol{\tau}_e - [\mathbf{J}^{-1}(\mathbf{h}_b - \boldsymbol{\Lambda}\mathbf{h}_w)] \times \mathbf{h}_b \quad (13a)$$

$$\dot{\mathbf{h}}_w = \boldsymbol{\tau}_w, \quad (13b)$$

where \mathbf{h}_w is the $L \times 1$ vector of the axial angular momenta of the wheels, $\boldsymbol{\tau}_e$ is the 3×1 vector of the external torque acting on the body, not including wheel torques, $\boldsymbol{\tau}_w$ is the $L \times 1$ vector of the internal axial torques applied by the platform to the wheels, and $\boldsymbol{\Lambda}$ is the $3 \times L$ matrix whose columns contain the axial unit vectors of the L momentum exchange wheels. Let $\boldsymbol{\omega}_{ib}^b$ denote the angular

velocity of the body frame \mathcal{F}_b relative to an inertial frame \mathcal{F}_i , measured in \mathcal{F}_b . Then, the vector \mathbf{h}_b is the total angular momentum of the spacecraft in the body frame, given as

$$\mathbf{h}_b = \mathbf{J}\boldsymbol{\omega}_{ib}^b + \boldsymbol{\Lambda}\mathbf{h}_w, \quad (14)$$

where \mathbf{J} is the inertia-like matrix defined as

$$\mathbf{J} \triangleq \mathbf{I} - \boldsymbol{\Lambda}\mathbf{I}_w\boldsymbol{\Lambda}^T. \quad (15)$$

The matrix \mathbf{I} is the inertia of the spacecraft, including wheels, and the matrix $\mathbf{I}_w = \text{diag}\{I_{w1}, I_{w2}, \dots, I_{wL}\}$ contains the axial moments of inertia of the wheels. The axial angular momenta of the wheels can be written in terms of the body angular velocity and the axial angular velocities of the wheels relative to the body, $\boldsymbol{\omega}_w$, as

$$\mathbf{h}_w = \mathbf{I}_w\boldsymbol{\Lambda}^T\boldsymbol{\omega}_{ib}^b + \mathbf{I}_w\boldsymbol{\omega}_w. \quad (16)$$

Note that $\boldsymbol{\omega}_w = [\omega_{w1}, \omega_{w2}, \dots, \omega_{wL}]^T$ is an $L \times 1$ vector, and that these relative angular velocities are those that would for instance be measured by tachometers fixed to the platform.

Equation (13) can also be written in terms of angular velocities. By defining $\boldsymbol{\mu} \triangleq [\mathbf{h}_b, \mathbf{h}_w]^T$ and $\mathbf{v} \triangleq [\boldsymbol{\omega}_{ib}^b, \boldsymbol{\omega}_w]^T$, we can write (14) and (16) in the compact form

$$\boldsymbol{\mu} = \boldsymbol{\Gamma}\mathbf{v}, \quad \text{where } \boldsymbol{\Gamma} = \begin{bmatrix} \mathbf{I} & \boldsymbol{\Lambda}\mathbf{I}_w \\ \mathbf{I}_w\boldsymbol{\Lambda}^T & \mathbf{I}_w \end{bmatrix}. \quad (17)$$

Clearly, we can find $\boldsymbol{\omega}_{ib}^b$ and $\boldsymbol{\omega}_w$ from $\mathbf{v} = \boldsymbol{\Gamma}^{-1}\boldsymbol{\mu}$, or equally, we can write $\dot{\mathbf{v}} = \boldsymbol{\Gamma}^{-1}\dot{\boldsymbol{\mu}}$. By utilizing the matrix inversion lemma, together with (17), we get

$$\begin{bmatrix} \dot{\boldsymbol{\omega}}_{ib}^b \\ \dot{\boldsymbol{\omega}}_w \end{bmatrix} = \begin{bmatrix} \mathbf{J}^{-1} & -\mathbf{J}^{-1}\boldsymbol{\Lambda} \\ -\boldsymbol{\Lambda}^T\mathbf{J}^{-1} & \boldsymbol{\Lambda}^T\mathbf{J}^{-1}\boldsymbol{\Lambda} + \mathbf{I}_w^{-1} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{h}}_b \\ \dot{\mathbf{h}}_w \end{bmatrix} \quad (18)$$

which can be written as

$$\dot{\boldsymbol{\omega}}_{ib}^b = \mathbf{J}^{-1} [-(\boldsymbol{\omega}_{ib}^b)^\times (\mathbf{I}\boldsymbol{\omega}_{ib}^b + \boldsymbol{\Lambda}\mathbf{I}_w\boldsymbol{\omega}_w) + \boldsymbol{\tau}_e] - \boldsymbol{\Lambda}\boldsymbol{\tau}_w \quad (19a)$$

$$\dot{\boldsymbol{\omega}}_w = \boldsymbol{\Lambda}^T\mathbf{J}^{-1} [(\boldsymbol{\omega}_{ib}^b)^\times (\mathbf{I}\boldsymbol{\omega}_{ib}^b + \boldsymbol{\Lambda}\mathbf{I}_w\boldsymbol{\omega}_w) - \boldsymbol{\tau}_e] + [\boldsymbol{\Lambda}^T\mathbf{J}^{-1}\boldsymbol{\Lambda} + \mathbf{I}_w^{-1}] \boldsymbol{\tau}_w. \quad (19b)$$

As can be seen from (19), the angular velocities are given in \mathcal{F}_b relative to \mathcal{F}_i , while the kinematics in (12) are relative to \mathcal{F}_o . However, it would be preferable if we in the model could describe the attitude of \mathcal{F}_b relative to \mathcal{F}_o . This can be done by utilizing the relation

$$\boldsymbol{\omega}_{ib}^b = \boldsymbol{\omega}_{ob}^b + \mathbf{R}_o^b\boldsymbol{\omega}_{io}^o \quad \text{and} \quad \dot{\boldsymbol{\omega}}_{ib}^b = \dot{\boldsymbol{\omega}}_{ob}^b + \dot{\mathbf{R}}_o^b\boldsymbol{\omega}_{io}^o, \quad (20)$$

where $\boldsymbol{\omega}_{io}^o = [0, -\omega_0, 0]^T$, and ω_0 is assumed constant and equal to the mean angular velocity of \mathcal{F}_o , given in \mathcal{F}_i .

This implies circular orbits. Now, the gravity gradient is included as a disturbance, that is, $\boldsymbol{\tau}_e = \boldsymbol{\tau} + \boldsymbol{\tau}_g$, where $\boldsymbol{\tau}$ is the torque provided from thrusters, and the gravity gradient is given as

$$\boldsymbol{\tau}_g = 3\omega_0^2 [\mathbf{c}_3 \times (\mathbf{I}\mathbf{c}_3)], \quad (21)$$

where \mathbf{c}_i denotes the i -th column of the rotation matrix \mathbf{R}_o^b . By utilizing (11) and (20), we can rewrite (19) as

$$\dot{\boldsymbol{\omega}}_{ob}^b = \hat{f}_{inert} + \hat{f}_\tau + \hat{f}_g + \hat{f}_{add} \quad (22a)$$

$$\dot{\boldsymbol{\omega}}_w = \bar{f}_{inert} + \bar{f}_\tau + \bar{f}_g, \quad (22b)$$

where the terms are given as

$$\hat{f}_{inert} = \mathbf{J}^{-1} [-(\boldsymbol{\omega}_{ob}^b - \omega_o \mathbf{c}_2)^\times (\mathbf{I}[\boldsymbol{\omega}_{ob}^b - \omega_o \mathbf{c}_2] + \boldsymbol{\Lambda} \mathbf{I}_w \boldsymbol{\omega}_w)] \quad (23a)$$

$$\bar{f}_{inert} = \boldsymbol{\Lambda}^T \mathbf{J}^{-1} [(\boldsymbol{\omega}_{ob}^b - \omega_o \mathbf{c}_2)^\times (\mathbf{I}[\boldsymbol{\omega}_{ob}^b - \omega_o \mathbf{c}_2] + \boldsymbol{\Lambda} \mathbf{I}_w \boldsymbol{\omega}_w)] \quad (23b)$$

$$\hat{f}_\tau = \mathbf{J}^{-1} \boldsymbol{\tau} - \mathbf{J}^{-1} \boldsymbol{\Lambda} \boldsymbol{\tau}_w \quad (23c)$$

$$\bar{f}_\tau = -\boldsymbol{\Lambda}^T \mathbf{J}^{-1} \boldsymbol{\tau} + [\boldsymbol{\Lambda}^T \mathbf{J}^{-1} \boldsymbol{\Lambda} + \mathbf{I}_w^{-1}] \boldsymbol{\tau}_w \quad (23d)$$

$$\hat{f}_g = \mathbf{J}^{-1} [3\omega_0^2 \mathbf{c}_3 \times (\mathbf{I}\mathbf{c}_3)] \quad (23e)$$

$$\bar{f}_g = -\boldsymbol{\Lambda}^T \mathbf{J}^{-1} [3\omega_0^2 \mathbf{c}_3 \times (\mathbf{I}\mathbf{c}_3)] \quad (23f)$$

$$\hat{f}_{add} = \omega_o \dot{\mathbf{c}}_2. \quad (23g)$$

3 Attitude control by means of explicit MPC

In the following, the eMPC controller is computed, and some aspects considering implementation are discussed.

The complete nonlinear model (12) and (22) can be written in compact form as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = [\dot{\boldsymbol{\omega}}_{ob}^b, \dot{\boldsymbol{\omega}}_w, \dot{\boldsymbol{\eta}}, \dot{\boldsymbol{\epsilon}}]^T, \quad (24)$$

where $\boldsymbol{\omega}_{ob}^b \triangleq [\omega_1, \omega_2, \omega_3]^T$ is the angular velocity of the body frame relative to the orbit frame, $\boldsymbol{\omega}_w \triangleq \omega_w$ is the angular velocity of the wheels about their spin axes (scalar in our case), and $\boldsymbol{\epsilon} \triangleq [\epsilon_1, \epsilon_2, \epsilon_3]^T$, which together with $\boldsymbol{\eta}$, make up the Euler parameters. The control input is given as $\mathbf{u} \triangleq [\boldsymbol{\tau}^T, \boldsymbol{\tau}_w^T]^T = [\tau_1, \tau_2, \tau_3, \tau_w]^T$.

3.1 Control design for the SSETI/ESEO satellite

As we consider a linear MPC approach in this paper, it is necessary to linearize the nonlinear model (24). By choosing the equilibrium point p equal to $\mathbf{x}^p = [\mathbf{0}^4, 1, \mathbf{0}^3]^T$, $\mathbf{u}^p = \mathbf{0}^4$, which equals the scenario where \mathcal{F}_b coincides with \mathcal{F}_o (nominal mode) and the angular velocity of the wheel is zero, it can be found that the linearized model can be written as

$$\Delta \dot{\mathbf{x}} = \mathbf{A}_c \Delta \mathbf{x} + \mathbf{B}_c \Delta \mathbf{u}, \quad (25)$$

where the matrices \mathbf{A}_c and \mathbf{B}_c are given as

$$\mathbf{A}_c = \begin{bmatrix} 0 & 0 & (1-k_x)\omega_0 & 0 \\ 0 & 0 & 0 & 0 \\ (k_z-1)\omega_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & -8k_x \omega_0^2 & 0 & 0 \\ 0 & 0 & \frac{-6k_y i_{22} \omega_0^2}{\kappa} & 0 \\ 0 & 0 & 0 & -2k_z \omega_0^2 \\ 0 & 0 & \frac{6k_y i_{22} \omega_0^2}{\kappa} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (26a)$$

$$\mathbf{B}_c = \begin{bmatrix} \frac{1}{i_{11}} & 0 & 0 & 0 \\ 0 & \frac{1}{\kappa} & 0 & -\frac{1}{\kappa} \\ 0 & 0 & \frac{1}{i_{33}} & 0 \\ 0 & -\frac{1}{\kappa} & 0 & \frac{i_{22}}{\kappa I_w} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (26b)$$

where $\mathbf{I} = \text{diag}(i_{11}, i_{22}, i_{33})$, $k_x = \frac{i_{22}-i_{33}}{i_{11}}$, $k_y = \frac{i_{11}-i_{33}}{i_{22}}$, $k_z = \frac{i_{22}-i_{11}}{i_{33}}$, and $\kappa = i_{22} - I_w$.

From the system matrix in (26a), we immediately con-

Table 2
Summary of tuning parameters

Parameter	Value
\mathbf{Q}	diag(500, 500, 500, $1 \cdot 10^{-7}$, 1, 1, 1)
\mathbf{R}	diag(200, 200, 200, 1)
N (horizon)	2
ρ (slack)	10

clude that the linearized system is uncontrollable, as all the terms corresponding to the state η equal zero. However, the linearized system is found to be stabilizable, and omitting η , also controllable. Also note that we can utilize the fact that the Euler parameters satisfy $\eta^2 + \epsilon^T \epsilon = 1$, making us able to keep track of, and update η in an open-loop manner.

Before we can apply the mpQP algorithm, (25) is converted into an equivalent *discrete-time* form by utilizing a zero-order hold approach, where in order to be well below the smallest time constant of the system, the sampling time was chosen to be $T_s = 0.1$ [sec]. Also, when deriving the controller, η is omitted, introducing the new state vector $\tilde{\mathbf{x}} \in \mathbb{R}^7$.

The tuning parameters used for deriving the explicit MPC controller are summarized in Table 2. Furthermore, the parameter space, in which we solve the mpQP, is chosen as

$$-[1, 1, 1, 800, 1, 1, 1]^T \leq \tilde{\mathbf{x}} \leq [1, 1, 1, 800, 1, 1, 1]^T \quad (27)$$

and the constraints are given as

$$\mathbf{u}_{max} = -\mathbf{u}_{min} = \begin{bmatrix} 0.0484 \\ 0.0484 \\ 0.0398 \\ 0.0020 \end{bmatrix}, \quad |\omega_w| \leq 527. \quad (28)$$

The constraints on \mathbf{u} are chosen based on the nominal thruster torques and maximum wheel torque, given in Table 1, while the constraint on the wheel angular velocity was defined by the SSETI project due to limited power supply.

The solution of the mpQP, obtained from the discrete-time version of (25), Table 1 and 2, and (28), gives a polyhedral partition over the parameter space in (27), consisting of 1679 regions. As in the double integrator example, if we denote each of these polyhedra as \mathcal{X}_i , where i is the specific region, then $\mathcal{X}_i \subset \mathbb{R}^7$, and each control law is given as in (8), $i \in \{1, \dots, 1679\}$.

3.2 Bang-bang modulation

Due to the on-off nature of the actuating thrusters, a bang-bang modulation scheme is applied. The technique is best explained through Fig. 2, where \mathbf{K}_{nom} represents the nominal thruster torques, and \mathbf{u}_* is given as

$$\mathbf{u}_* : \text{sign}(\mathbf{u}) = \begin{cases} -\mathbf{1} & \text{if } \mathbf{u} \leq -d\mathbf{z}, \\ \mathbf{1} & \text{if } \mathbf{u} \geq d\mathbf{z}, \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (29)$$

where the dead-zone is chosen based on performance as well as fuel consumption. Other techniques also exist in solving this problem, one being pulse-width pulse-frequency (PWPF) modulation [15].

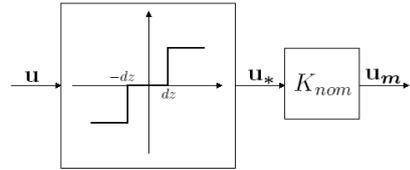


Fig. 2. Bang-bang modulation with dead-zone

4 Simulation results

The closed-loop simulations in this section have been performed with the complete nonlinear model (24), where initial conditions for the dynamics and kinematics, as well as initial Keplerian orbital elements, are given as in Table 3.

As mentioned in the introduction, only the nominal mode will be considered at this point, which means that the best obtainable result is whenever the body frame \mathcal{F}_b coincides with the orbit frame \mathcal{F}_o . In the plots the Euler parameters have been transformed into Euler angles (XYZ) [deg].

4.1 Case I

No noise or bang-bang modulation is applied. The performance of the eMPC controller is compared to that of the stationary discrete LQR (similar cost matrices). For the latter, we include saturation on the control inputs, equal to the input constraints of the eMPC. The results are shown in Fig. 3, where the dotted lines represent the constraints. The response is comparable for both controllers (only the eMPC response is shown when minor differences occurred), and when constraints are inactive the eMPC equals the stationary LQR. However, during the transient response, where the state constraint for the wheel velocity is satisfied when using eMPC, there is a large overshoot in the LQR case. Finally, we observe that the input constraints for the eMPC controller are satisfied throughout the simulation.

Table 3
Summary of simulations

Case I and II	Initial condition	Set-point	Unit
ω_{ob}^b	$\{-0.05, 0.15, -0.08\}$	$\{0, 0, 0\}$	rad/s
ω_w	300	0	rad/s
Euler angles	$\{-25, 60, 90\}$	$\{0, 0, 0\}$	deg
Keplerian	Initial condition		Unit
$[i, \omega, \Omega, \nu]$	$\{7, 178, -10, 0\}$		deg
a	17125		km
e	0.0		-

Table 4
RMS errors in states

States	Errors	Unit
ω_{ob}^b	$\{0.0035, 0.0052, 0.0035\}$	rad/s
ω_w	0.5	rad/s
Euler angles (XYZ)	$\{0.1, 0.1, 0.1\}$	deg

While LQR (or other control schemes) with forced saturation might work for some situations, it should be clear that this is an ad-hoc solution, which provides no control of potential output or state constraints.

Furthermore, we know that LQR (and any other non-predictive schemes) only considers the present state. When using eMPC we can compensate for control constraints some time steps ahead, which in general will decrease the accumulated control effort. In our case, it is found that the eMPC controller uses slightly less propulsion than the LQR. However, it is found that when using eMPC, the accumulated wheel torque is in the order of one half of that using LQR, hence decreasing power usage considerably.

4.2 Case II

Similar scenario as in *Case I*, but also including noise according to Table 4. Only the eMPC controller is considered, and bang-bang modulation is used for realizing the on-off nature of the thrusters. The results are shown in Fig. 4. As in *Case I*, we obtain a desired response, while none of the constraints are violated.

5 Further work

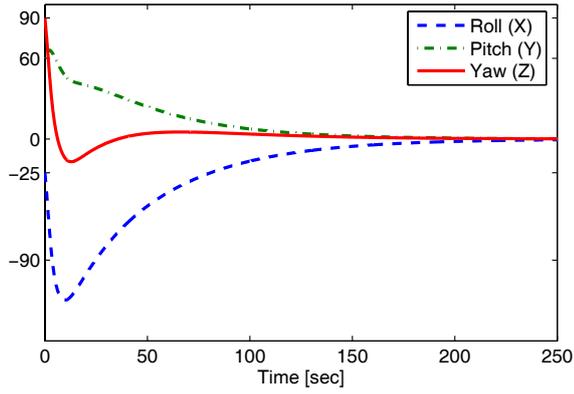
Due to the complexity of the mpQP solution, the prediction horizon is quite short. A longer horizon would be preferable, and should be attempted. Furthermore, by solving the standard MPC problem, it is indicated that having constraints on the angular rates decreases fuel usage considerably. This should also be incorporated in the eMPC controller.

6 Conclusions

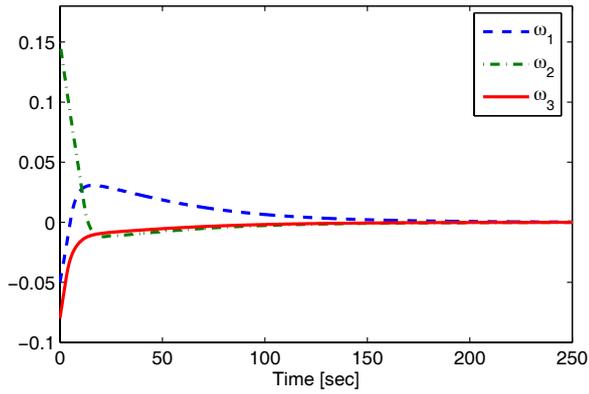
It has been shown that explicit solutions to constrained linear MPC problems can be computed for the attitude control problem by solving multi-parametric quadratic programs (mpQP). This approach should be considered when constraints need to be taken into account. The theoretical results have been supported by simulations.

References

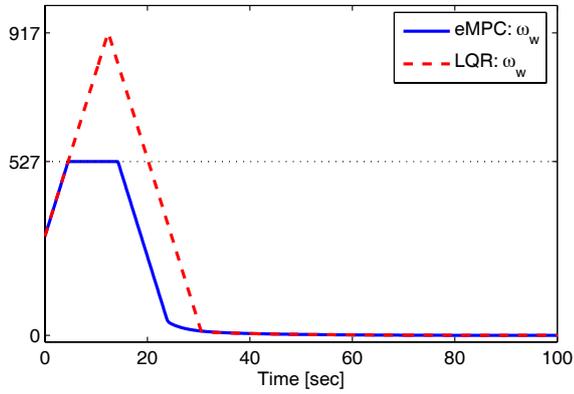
- [1] A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.
- [2] Y.-P. Chen and S.-C. Lo. Sliding-mode controller design for spacecraft attitude tracking maneuvers. *IEEE Transactions on Aerospace and Electronic Systems*, 29(4):1328–1333, 1993.
- [3] M. Dalsmo and O. Egeland. State feedback H_∞ control of a rigid spacecraft. *Proceedings of the 34th IEEE Conference on Decision and Control*, 4:3968–73, 1995.
- [4] O. Egeland and J.T. Gravdahl. *Modeling and Simulation for Automatic Control*. Marine Cybernetics, Trondheim, Norway, 2002.
- [5] G. Ferrari-Trecate, F.A. Cuzzola, D. Mignone, and M. Morari. Analysis and control with performance of piecewise affine and hybrid systems. *Proceedings of the American Control Conference*, 1:200–205, 2001.
- [6] C.D. Hall, P. Tsiotras, and H. Shen. Tracking rigid body motion using thrusters and momentum wheels. *Journal of the Astronautical Sciences*, 50(3):311–323, 2003.
- [7] Ø. Hegrenæs. Attitude control by means of explicit model predictive control, via multi-parametric quadratic programming. Master’s thesis, Norwegian University of Science and Technology, 2004.
- [8] P.C. Hughes. *Spacecraft attitude dynamics*. John Wiley & Sons, 1986.
- [9] T.A. Johansen, I. Petersen, and O. Slupphaug. On explicit suboptimal lqr with state and input constraints. *Proceedings of the 39th IEEE Conference on Decision and Control*, pages 662–7, 2000.
- [10] S.M. Joshi, A.G. Kelkar, and J.T.-Y. Wen. Robust attitude stabilization of spacecraft using nonlinear quaternion feedback. *IEEE Transactions on Automatic Control*, 40(10):1800–3, 1995.
- [11] L.-L. Show, J.-C. Juang, C.-T. Lin, and Y.-W. Jan. Spacecraft robust attitude tracking design: Pid control approach. *Proceedings of the 2002 American Control Conference*, 2:1360–5, 2002.
- [12] L.-L. Show, J.-C. Juang, and C.-D. Yang. Nonlinear H_∞ robust control for satellite large angle attitude maneuvers. *Proceedings of the American Control Conference*, 2:1357–1362, 2001.
- [13] P. Tøndel, T.A. Johansen, and A. Bemporad. An algorithm for multi-parametric quadratic programming and explicit mpc solutions. *Automatica*, 39(3):489–97, 2003.
- [14] J. T.-Y. Wen and K. Kreutz-Delgado. The attitude control problem. *IEEE Transactions on Automatic Control*, 36(10):1148–1162, 1991.
- [15] B. Wie and P.M. Barba. Quaternion feedback for spacecraft large angle maneuvers. *Journal of Guidance, Control, and Dynamics*, 8(3):360–5, 1985.



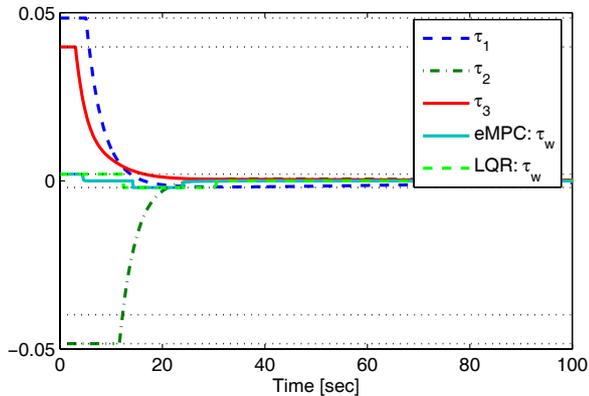
(a) Euler angles (only eMPC shown)



(b) Angular velocity (only eMPC shown)

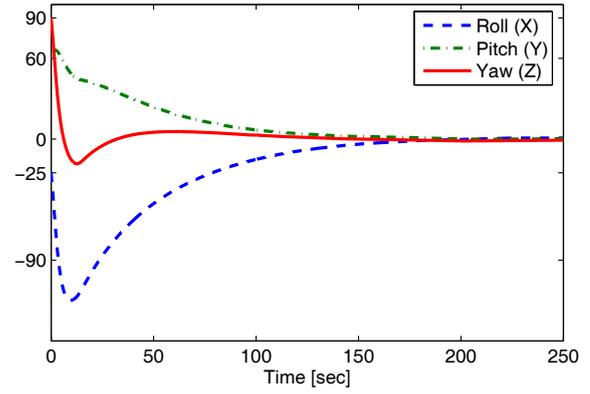


(c) Wheel velocity (magnified)

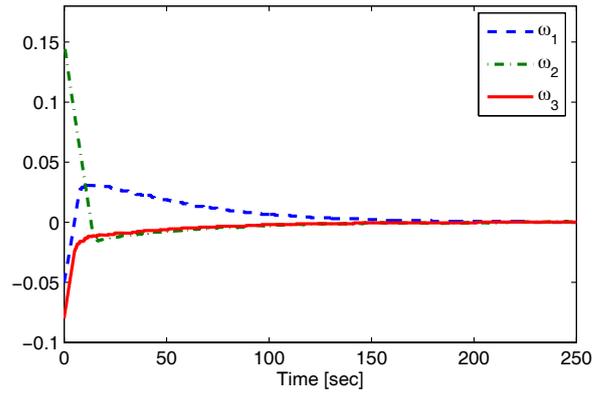


(d) Control input (magnified)

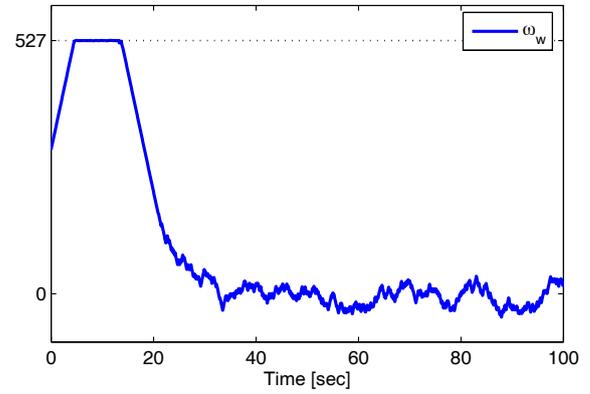
Fig. 3. Case I



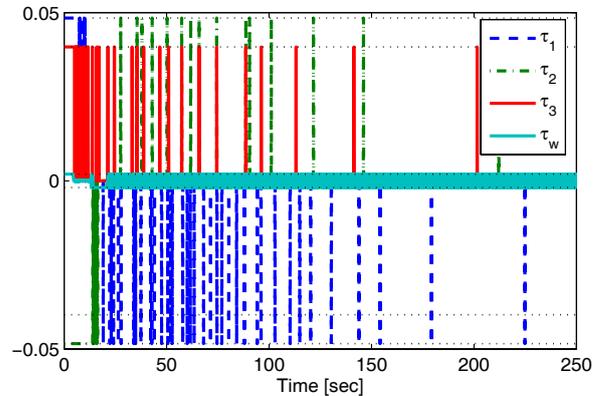
(a) Euler angles



(b) Angular velocity



(c) Wheel velocity (magnified)



(d) Control input

Fig. 4. Case II