Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

An Implementation of Predictive Functional Control for Image-Based Satellite Attitude Control

Andrej Zdešar [*] **Gregor Klančar** [*,†] **Gašper Mušič** [*,†]
Drago Matko [*,†] **Igor Škrjanc** [*]

[*] *Faculty of Electrical Engineering, University of Ljubljana*
Tržaška 25, 1000 Ljubljana, Slovenia
(e-mail: andrej.zdesar@fe.uni-lj.si)
[†] *Centre of Excellence for Space Sciences and Technologies*
Aškerčeva 12, 1000 Ljubljana, Slovenia

Abstract: This paper presents an implementation of predictive functional control for image-based satellite attitude control. The approach considers Low Earth Orbit small satellites equipped with an on-board camera for interactive Earth observation. The control goal is to keep a target on the Earth surface in the satellite's camera field of view during orbiting of the satellite above the target. The control error is given in the image frame, therefore in-image interaction with the system is possible. Two types of control problems are considered: target tracking and oriented-target tracking. The system contains a delay that is taken into account in development of the predictive control law. The control problem is solved in the framework of predictive functional control that is based on the kinematic model of the satellite attitude. The presented control algorithm is validated in simulation environment.

Keywords: predictive control, model based control, visual servoing, satellite, kinematic model

1. INTRODUCTION

The development of small, micro, nano and pico satellites is expanding. These satellites normally operate in Earth's low-orbit and their main task is Earth observation and monitoring. For this purpose the attitude control has to be developed that enables appropriate alignment of the satellite with respect to the Earth surface. The attitude of the satellite can be controller by different actuators, e.g. reaction wheels, small thrusters, magnetic coils, permanent magnets, etc. And the satellite's attitude can be measured by a range of sensors, e.g. sun sensor, earth magnetic sensor, star trackers, gyroscopes, etc. In this paper a camera is used as a sensor for determining the satellite attitude. This configuration has already been considered in Klančar et al. (2012); Zdešar et al. (2013), but in this paper the main emphasis is given on development of the control law. The quaternion notation is used to represent rotations between different coordinate frames. The control error, also given in quaternion form, can be set in way that enables arbitrary in-image alignment of the target and reference vectors.

Since the considered configuration uses an image sensor (camera) in a feedback loop for motion control of a mobile system (satellite), this is a visual servoing (VS) problem (Corke (2011)). The VS techniques can also be found in a variety of other applications: ground mobile vehicles (Klančar et al. (2004); De Luca et al. (2008); von Hundelshausen (2004)), (under)water mobile systems (Kim et al. (2011)), unmanned aerial vehicles (Meingast et al. (2004); Eberli et al. (2010); Bošnak et al. (2011, 2012); Shakernia et al. (1999); Bourquardez and Chaumette (2007)), robotic manipulators (Zhang (2009); Kosmopoulos (2011); Kelly et al. (2000)), etc. Based on the definition of the error, the VS approaches are normally divided into three main groups: position based visual servoing (PBVS), image based visual servoing (IBVS) and hybrid visual servoing. In this paper the IBVS approach is considered, in which the control error between the desired and current satellite attitude is defined directly in image. Since the camera is mounted on a satellite this is an eye-in-hand VS configuration.

The huge amount of data the images can provide present a challenging task, when it comes to extracting the relevant information for a specific task, especially when the task is to be executed in real-time as is the case of visual servoing of satellite attitude. Presentation of machine vision approaches for target tracking is out of the scope of this paper. We suppose that the result of in-image tracking algorithm is available. Since image processing is time-consuming operation, we take delays into account in the design of the control algorithm. We also suppose that the image-grabbing frequency is limited.

In this paper we consider to solve the presented control problem in the framework of predictive functional control (PFC). PFC belongs to a class of predictive control (Morari and Lee (1999); Maciejowski (2002); Brosilow and Joseph (2002); Camacho and Bordons (2004)). Over the years a diverse range of predictive control algorithms have been developed: model algorithmic control (MAC) (Rouhani and Mehra (1982)), model-based predictive control (MPC) (Richalet (1993)), generalized predictive control (GPC) (Normey-Rico et al. (1999)), predictive functional control (PFC) (Škrjanc and Matko (2000); Lepetič et al. (2003); Richalet and O'Donovan (2009)), etc. There are some nice properties about PFC that make this type of control approach appealing. The basic principles of the PFC are easy to understand, and since the control law can be derived in analytical form, this makes it also suitable for implementation on a real-time system with fast dynamics.

This paper is structured as follows. Section 2 introduces the notation used and presents the coordinate frames for description of the satellite and camera in space, and gives the model of the satellite and the camera. Section 3 presents the derivation of two types of the control errors. This is followed by section 4 where PFC algorithm for satellite attitude control is presented. In section 5 experimental results are given and in the final section 6 some conclusions are made.

2. SYSTEM MODELLING

2.1 Notation

Throughout the paper the following notation is used. Small-caps bold-face letters are used for column vector (e.g. $\boldsymbol{x}$), and big-caps bold-face letter are used for matrices (e.g. $\boldsymbol{X}$). The N-dimensional point $\boldsymbol{p}$ expressed in the frame F is denoted as ${}^{F}\boldsymbol{p} \in \mathbb{R}^N$. A quaternion, $\mathring{q} \in \mathbb{H}$, may be expressed as a tuple $(q_0, \boldsymbol{q})$, where $q_0 \in \mathbb{R}$ and $\boldsymbol{q} \in \mathbb{R}^3$. In quaternion notation, the transformation of a point ${}^{A}\boldsymbol{p} \in \mathbb{R}^3$ expressed in the frame A to the frame B is defined as follows:

$$\mathring{q}({}^{B}\boldsymbol{p}) = {}^{B}\mathring{q}_A \cdot \mathring{q}({}^{A}\boldsymbol{p}) \cdot {}^{B}\mathring{q}_A^{*} + \mathring{q}({}^{B}\boldsymbol{t}_A)\,, \tag{1}$$

where $\mathring{q}(\boldsymbol{p}) = (0, \boldsymbol{p})$ represents the vector $\boldsymbol{p}$ in quaternion form. The ${}^{B}\mathring{q}_A$ is the rotation quaternion from the frame A to the frame B and ${}^{B}\boldsymbol{t}_A \in \mathbb{R}^3$ is the translation vector between the origins of the frames A and B, expressed in the frame B. The rotation quaternion ${}^{B}\mathring{q}_A$ in (1) can be expressed through the frame C:

$${}^{B}\mathring{q}_A = {}^{B}\mathring{q}_C \cdot {}^{C}\mathring{q}_A\,. \tag{2}$$

Note that multiplication of quaternions in (2) is not a commutative operation.

2.2 Camera model

For the camera model a simple pin-hole camera model is used, without any lens distortions or other artefacts. The point ${}^{C}\boldsymbol{p} \in \mathbb{R}^3$ expressed in the camera frame is transformed into the homogeneous point ${}^{P}\tilde{\boldsymbol{p}} \in \mathbb{R}^3$ in the picture frame according to:

$${}^{P}\tilde{\boldsymbol{p}} \simeq {}^{P}\boldsymbol{M}_C \cdot {}^{C}\boldsymbol{p}\,, \tag{3}$$

where ${}^{P}\boldsymbol{M}_C \in \mathbb{R}^3 \times \mathbb{R}^3$ is the camera model. Therefore, the points that lie on a ray in the camera frame that is passing through the camera centre and the image point satisfy the following equation:

$${}^{C}\boldsymbol{p} = \lambda \cdot {}^{P}\boldsymbol{M}_C^{-1} \cdot {}^{P}\tilde{\boldsymbol{p}}\,, \tag{4}$$

where $\lambda \in \mathbb{R}$ is the scalling factor. According to the figure 1, the points on the ray are in front of the camera image plane when the scalling factor is negative ($\lambda < 0$).

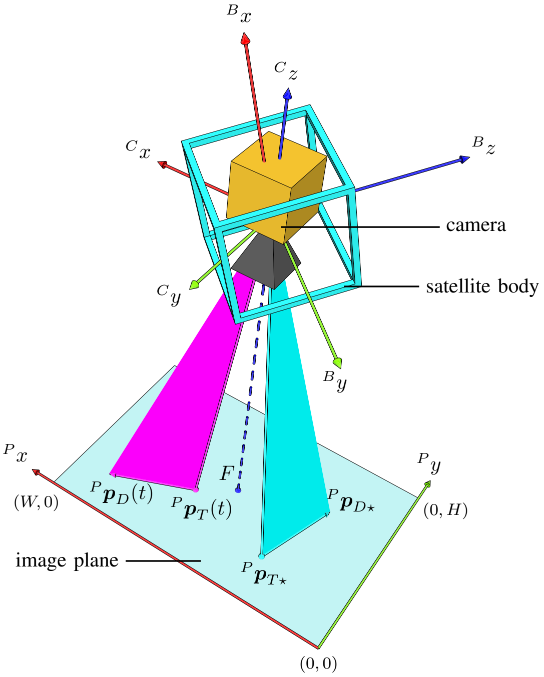


Fig. 1. Geometrical relations between the satellite body frame B, the camera frame C and the image (picture) frame P.

2.3 Satellite model

For description of the satellite pose the conventional coordinate frames are used: ECI (Earth-centered inertial), ECEF (Earth-centered Earth-fixed), ECO (Earth-centered orbital), LVLH (local-vertical local-horizontal), O (orbital) and B (body). To simulate the position of the satellite in orbit, which is defined by the Kepler elements that describe the satellite position at the initial time, the *Simplified General Perturbations* (SGP) model is used (Klančar et al. (2012)). The Kepler elements of the orbit are obtained from the NORAD two-element set. For more information on how to calculate the position of the satellite at some arbitrary time ${}^{ECI}\boldsymbol{t}_B(t)$ see Klančar et al. (2012).

The dynamic and kinematic models of the satellite are as follows (Wertz (1978)):

$$\frac{d^{B}\boldsymbol{\omega}(t)}{dt} = {}^{B}\boldsymbol{J}^{-1}\left({}^{B}\boldsymbol{m}(t) + \left[{}^{B}\boldsymbol{J}\,{}^{B}\boldsymbol{\omega}(t)\right]_{\times}{}^{B}\boldsymbol{\omega}(t)\right), \tag{5}$$

where ${}^{B}\boldsymbol{J}$ is the tensor of the satellite's moments of inertia, ${}^{B}\boldsymbol{\omega}(t) \in \mathbb{R}^3$ is the vector of the angular velocities of the satellite with respect to the ECI frame (but expressed in the satellite body frame B), ${}^{B}\boldsymbol{m}(t)$ are the moments applied to the satellite (expressed in the satellite body frame B). The kinematic model of the satellite is as follows:

$$\frac{d^{B}\mathring{\boldsymbol{q}}_{ECI}(t)}{dt} = \frac{1}{2}\begin{bmatrix} 0 & -{}^{B}\omega_x(t) & -{}^{B}\omega_y(t) & -{}^{B}\omega_z(t) \\ {}^{B}\omega_x(t) & 0 & {}^{B}\omega_z(t) & -{}^{B}\omega_y(t) \\ {}^{B}\omega_y(t) & -{}^{B}\omega_z(t) & 0 & {}^{B}\omega_x(t) \\ {}^{B}\omega_z(t) & {}^{B}\omega_y(t) & -{}^{B}\omega_x(t) & 0 \end{bmatrix} {}^{B}\mathring{\boldsymbol{q}}_{ECI}(t)\,. \tag{6}$$

Note that the quaternions in (6) are written as four element vectors (a quaternion $\mathring{q}$ is represented as $\mathring{\boldsymbol{q}}^T = [q_0\ \boldsymbol{q}^T]$).

In addition to the aforementioned coordinate frames an additional frame for describing the target is defined. The target frame T is fixed to a point on the Earth's surface with z-axis pointing away from the Earth's centre and y-axis in-plane with the Earth's rotation axis. The target frame T can be expressed in the $ECEF$ frame with quaternion ${}^{ECEF}\mathring{q}_T$ and position of the target frame origin ${}^{ECEF}\boldsymbol{t}_T$. The position of a point on the Earth's surface is normally given in terms of geodetic latitude, longitude and altitude. To be able to convert these values to Cartesian coordinates in the $ECEF$ frame, the model of the Earth's surface needs be known. One of the most accurate model of the Earth's surface is the WGS84 (NIMA (2000)). However, in our case we consider a more simple model: we assume that the Earth is a sphere. For our case this is a good approximation, but if one would require higher accuracy, the conversion between geodetic coordinates to Cartesian coordinates would need to be made wherever appropriate.

3. CONTROL ERROR CALCULATION

A camera is attached to the satellite in a way that the camera focus point is at the centre of the satellite body frame (figure 1). Assume that the orientation between the camera and body frame is represented by quaternion ${}^{B}\mathring{q}_C$. The control goal is to rotate the camera view to a specific target. Two cases are considered (Zdešar et al. (2013)): in the first case the view needs to point towards the target point, and in the second case the view needs to point towards the target point at specific orientation. In both cases the error in the body frame of the satellite has to be determined, since the actuators that control the orientation of the satellite are mounted on the body of the satellite. In the following subsections a summary of how this error can be obtained from the image, and also how this error is obtained from spatial relations between the bodies is presented.

3.1 Target tracking

In this case the target is given with the position of the target point ${}^{T}\boldsymbol{p}_T$. We are seeking for the rotation ${}^{B\star}\mathring{q}_B(t)$ that would make the reference vector in the camera frame ${}^{C}\boldsymbol{p}_{T\star}$ point towards the target point ${}^{T}\boldsymbol{p}_T$. In other words, the goal is to align the vector ${}^{C}\boldsymbol{p}_T(t)$ with the reference vector ${}^{C}\boldsymbol{p}_{T\star}$ (see figure 1) — notice that this two vectors are both defined in the camera frame. The required camera rotation quaternion is obtained as follows:

$${}^{C\star}\mathring{q}_C(t) = \left({}^{C}\boldsymbol{p}_T^T(t)\cdot{}^{C}\boldsymbol{p}_{T\star} + \left\|{}^{C}\boldsymbol{p}_T(t)\right\|\left\|{}^{C}\boldsymbol{p}_{T\star}\right\|, \left[{}^{C}\boldsymbol{p}_T(t)\right]_\times {}^{C}\boldsymbol{p}_{T\star}\right). \quad (7)$$

Note that the lengths of all the vectors in (7) are not important as long as they are all non-zero. Therefore, the vectors (rays) ${}^{C}\boldsymbol{p}_T(t)$ and ${}^{C}\boldsymbol{p}_{T\star}$ can be obtained in the image, using the relation (4). Note also that the solution (7) is only one of the pencil of solutions that achieves the desired alignment towards the target point, since one degree of freedom was left unconstrained.

Based on the spatial relations between the frames (objects), the vector ${}^{C}\boldsymbol{p}_T(t)$ can also be derived in the following way:

$$\mathring{q}({}^{C}\boldsymbol{p}_T(t)) = {}^{C}\mathring{q}_B \cdot \mathring{q}({}^{B}\boldsymbol{p}_T(t)) \cdot {}^{C}\mathring{q}_B^{*}\,, \quad (8)$$

$$\begin{aligned}\mathring{q}({}^{B}\boldsymbol{p}_T(t)) &= {}^{B}\mathring{q}_{ECEF}(t)\cdot\mathring{q}({}^{ECEF}\boldsymbol{p}_T)\cdot{}^{B}\mathring{q}_{ECEF}^{*}(t)-\\ &\quad - {}^{B}\mathring{q}_{ECI}(t)\cdot\mathring{q}({}^{ECI}\boldsymbol{t}_B(t))\cdot{}^{B}\mathring{q}_{ECI}^{*}(t)\,, \quad (9)\end{aligned}$$

$$\begin{aligned}\mathring{q}({}^{ECEF}\boldsymbol{p}_T) &= {}^{ECEF}\mathring{q}_T\cdot\mathring{q}({}^{T}\boldsymbol{p}_T)\cdot{}^{ECEF}\mathring{q}_T^{*}+\\ &\quad + \mathring{q}({}^{ECEF}\boldsymbol{t}_T)\,, \quad (10)\end{aligned}$$

$${}^{ECEF}\mathring{q}_B(t) = {}^{ECEF}\mathring{q}_{ECI}(t)\cdot{}^{ECI}\mathring{q}_B(t)\,. \quad (11)$$

These equations enable simulation of the system and calculation of the control error. However, in the final implementation the control error is calculated from image data using machine vision.

3.2 Oriented-target tracking

In this case we do not only want that the camera is pointing towards a specific target, but also that some vector in the image is aligned with the image of some vector on the ground. In other words, the goal is to align two semi-planes as it is also shown in the figure 1. For the purpose of the following derivations, the vector $\boldsymbol{n}^T = [0\ 0\ -1]$ is introduced. The orientation of the reference direction in the camera frame is given as follows:

$${}^{D}\mathring{q}_{C\star} = \left({}^{C}\boldsymbol{t}_{T\star}^T\cdot\boldsymbol{n} + \left\|{}^{C}\boldsymbol{t}_{T\star}\right\|, \left[{}^{C}\boldsymbol{t}_{T\star}\right]_\times\boldsymbol{n}\right). \quad (12)$$

The overall rotation that aligns with the target at specific angle can be written as combination of rotations:

$${}^{C\star\star}\mathring{q}_C(t) = {}^{C\star\star}\mathring{q}_{C\star}(t)\cdot{}^{C\star}\mathring{q}_C(t)\,. \quad (13)$$

The rotation required to align the reference semi-plane with the target semi-plane when the camera is already pointing towards the target point is:

$${}^{C\star\star}\mathring{q}_{C\star}(t) = {}^{D}\mathring{q}_{C\star}\cdot{}^{D\star}\mathring{q}_D(t)\cdot{}^{D}\mathring{q}_{C\star}^{*}\,, \quad (14)$$

where

$${}^{D\star}\mathring{q}_D(t) = \left(\boldsymbol{v}_1^T(t)\cdot\boldsymbol{v}_2 + \|\boldsymbol{v}_1(t)\|\,\|\boldsymbol{v}_2\|, [\boldsymbol{v}_1(t)]_\times\boldsymbol{v}_2\right). \quad (15)$$

The vectors $\boldsymbol{v}_1(t)$ and $\boldsymbol{v}_2$ are obtained from projection of the vectors ${}^{D}\boldsymbol{p}_D(t)$ and ${}^{D}\boldsymbol{p}_{D\star}$ onto the plane $\boldsymbol{n}$, respectively:

$$\mathring{q}(\boldsymbol{v}_1(t)) = \frac{\mathring{q}({}^{D}\boldsymbol{p}_D(t)) + \mathring{q}(\boldsymbol{n})\cdot\mathring{q}({}^{D}\boldsymbol{p}_D(t))\cdot\mathring{q}(\boldsymbol{n})}{2}\,, \quad (16)$$

$$\mathring{q}(\boldsymbol{v}_2) = \frac{\mathring{q}({}^{D}\boldsymbol{p}_{D\star}) + \mathring{q}(\boldsymbol{n})\cdot\mathring{q}({}^{D}\boldsymbol{p}_{D\star})\cdot\mathring{q}(\boldsymbol{n})}{2}\,, \quad (17)$$

$$\mathring{q}({}^{D}\boldsymbol{p}_D(t)) = {}^{D}\mathring{q}_C(t)\cdot\mathring{q}({}^{C}\boldsymbol{p}_D(t))\cdot{}^{D}\mathring{q}_C^{*}(t)\,, \quad (18)$$

$$\mathring{q}({}^{D}\boldsymbol{p}_{D\star}) = {}^{D}\mathring{q}_{C\star}\cdot\mathring{q}({}^{C}\boldsymbol{p}_{D\star})\cdot{}^{D}\mathring{q}_{C\star}^{*}\,, \quad (19)$$

where

$${}^{D}\mathring{q}_C(t) = {}^{D}\mathring{q}_{C\star}\cdot{}^{C\star}\mathring{q}_C(t)\,. \quad (20)$$

Again it should be noted that the points ${}^{C}\boldsymbol{p}_D(t)$ and ${}^{C}\boldsymbol{p}_{D\star}$ can be determined in the image frame P using (4). Therefore this control problem can be solved using the image-based control. Based on the spatial relations between the frames, the vector ${}^{C}\boldsymbol{p}_D(t)$ can be derived in the same way as vector ${}^{C}\boldsymbol{p}_T(t)$ in (8).

4. CONTROL LAW

The required satellite rotation ${}^{B\star}\mathring{q}_B(t)$ (or ${}^{B\star\star}\mathring{q}_B(t)$) that achieves the desired camera orientation is related to the camera rotation ${}^{C\star}\mathring{q}_C(t)$ (or ${}^{C\star\star}\mathring{q}_C(t)$) as follows:

$${}^{B\star}\mathring{q}_B(t) = {}^{B}\mathring{q}_C\cdot{}^{C\star}\mathring{q}_C(t)\cdot{}^{B}\mathring{q}_C^{*}\,. \quad (21)$$

From the required rotation given in quaternion $\mathring{q}(t)$, ($q_0(t) > 0\ \forall t > 0$) — which can be either ${}^{B\star}\mathring{q}_B(t)$ or ${}^{B\star\star}\mathring{q}_B(t)$ — the control error vector $\boldsymbol{e}^T(t)$ is calculated:

$$\boldsymbol{e}^T(t) = 2\arccos(q_0(t))\frac{\boldsymbol{q}(t)}{\|\boldsymbol{q}(t)\|}\,. \quad (22)$$

The control error is considered to be defined in the image-space, although this is not clearly observed from (22). A closer examination of the equations in the section 3 reveals that the control error was actually defined from semi-lines (rays) in the camera frame. Since a ray in the camera frame can be presented as a point in the picture frame, and the reference orientation is also given in the image frame (using the same image to camera model), the control law is considered to belong to a class of image-based visual servoing approaches.

The control algorithm is developed in a PFC framework based on kinematic model of the system:

$$\boldsymbol{u}(t) = {}^B\dot{\boldsymbol{\omega}}(t) = {}^B\ddot{\boldsymbol{\varphi}}(t)\,, \tag{23}$$

where ${}^B\boldsymbol{\varphi} \in \mathbb{R}^3$ is a vector of the angles that are to be controlled. Based on the required input $\boldsymbol{u}(t)$, the moments that need to be applied to the actuators are obtained from the dynamic model of the system (5):

$${}^B\boldsymbol{m}(t) = {}^B\boldsymbol{J}\boldsymbol{u}(t) - \left[{}^B\boldsymbol{J}{}^B\boldsymbol{\omega}(t)\right]_\times {}^B\boldsymbol{\omega}(t)\,. \tag{24}$$

It should be noted that this approach requires known inertial tensor ${}^B\boldsymbol{J}$ of the system and measurement of the angular velocities ${}^B\boldsymbol{\omega}(t)$.

The system (23) consists of three univariate double integrators without any input-output cross coupling. Each univariate system can be written as a transfer function in discrete time-domain:

$$H(z) = \frac{T_s^2}{2}\frac{z^{-1}(1+z^{-1})}{(1-z^{-1})^2}\,, \tag{25}$$

where T_s is the sampling time.

In order to implement a PFC, the following discrete model in introduced (see figure 2):

$$\begin{aligned} Y(z) &= H_0(s)U(z) + (H_1(z)+H_2(z))\,Y(z)\,,\\ H_0(z) &= \frac{gz^{-1}(1+z^{-1})}{(1-a_1z^{-1})(1-a_2z^{-1})}\,,\\ H_1(z) &= \frac{z^{-1}(1-b_1z^{-1})}{1-a_1z^{-1}}\,,\\ H_2(z) &= \frac{z^{-1}(1-b_2z^{-1})}{1-a_2z^{-1}}\,, \end{aligned} \tag{26}$$

where $\|a_1\| < 1$ and $\|a_2\| < 1$. When the following set of conditions is met:

$$a_2 = -a_1 \bigwedge b_2 = b_1 \bigwedge b_1 = \frac{1+a_1^2}{2} \bigwedge g = \frac{T_s^2}{2}\,, \tag{27}$$

the model (26) takes the form of a discrete double integrator (25) — this is the decomposed model of the system (25). Notice, that since $a_1 \in \mathbb{R}$ and $\|a_1\| < 1$, the factor b_1 is limited to the real range $[0.5, 1)$.

The output of the model is therefore calculated as sum of outputs from the submodels:

$$y(k) = y_0(k) + y_1(k) + y_2(k)\,, \tag{28}$$

where the submodels are:

$$\begin{aligned} y_0(k) &= (a_1+a_2)y_0(k-1) - a_1a_2y_0(k-2)+\\ &\quad + gu(k-1) + gu(k-2)\,,\\ y_1(k) &= a_1y_1(k-1) + y^*(k-1) - b_1y^*(k-2)\,,\\ y_2(k) &= a_2y_2(k-1) + y^*(k-1) - b_2y^*(k-2)\,. \end{aligned} \tag{29}$$

To develop the PFC law, a model of the reference signal must be introduced. In our case, the reference signal is defined as follows:

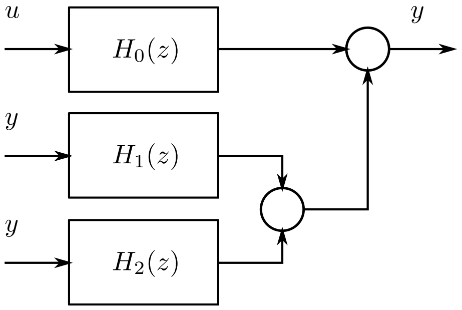


Fig. 2. Decomposition of the model.

$$y_r(k+h) = a_r^h y_r(k) + (1-a_r^h)w(k)\,, \tag{30}$$

where a_r is a parameter of the reference model, and $w(k)$ is the unfiltered reference signal.

Let us assume that all the h future inputs are equal to the input $u(k)$, that is $u(k) = u(k+1) = \ldots = u(k+h-1)$. The PFC is then obtained from the basic requirement:

$$\Delta_m(k) = \Delta_p(k)\,, \tag{31}$$

where $\Delta_m(k) = y(k+h) - y(k)$ and $\Delta_p(k) = y_r(k+h) - y_p(k)$ and $y_p(k)$ is the system output. The system input signals are contained in the term $\Delta_{m0}(k) = \delta_{m0}(k) + d_{m0}u(k)$. Therefore, from the relation (31) the PFC control law is obtained:

$$u(k) = \frac{(1-a_r^h)e(k) - \delta_{m0}(k) - \Delta_{m1}(k) - \Delta_{m2}(k)}{d_{m0}}\,, \tag{32}$$

where $e(k) = w(k) - y_p(k)$.

So far the delays have not been considered, but since the error signal is determined from time-consuming image processing, the delays must be taken into account in the design of the control law. The delay of the system can be compensated in the usual way by calculating the error factor in (32) as $e(k) = w(k) - y_p(k) + y_d(k) - y(k)$, where $y_d(k)$ is the delayed model output.

In the development of the control law, the position of the satellite was assumed to be constant with respect to the target. However, the satellite and Earth move. As the satellite travels about the orbit, the position of the camera with respect to the target changes with time. Therefore in the error-free target tracking the satellite angular velocity is not zero. The required angular velocity of the satellite in steady state can be determined form the spatial relations between satellite and the target introduced in section 3. Hovever, the presented control law should eliminate the steady state error.

5. EXPERIMENTS

The presented control algorithm is validated using the satellite simulator implemented in Matlab environment. In simulation, the Kepler elements of the Lapan-Tubsat satellite Renner and Buhl (2008) are used, whose sunsynchronous orbit is approximately 600 km above the Earth's surface and requires 97 min to travel one orbital period. The satellite's orbital speed is approximately $7.5\,\mathrm{km\,s^{-1}}$. The satellite's moment of inertia matrix $\boldsymbol{J}$ is set to be a unit matrix. The system contains a delay of 5 s. The absolute value of the control action is bounded to 5×10^{-4} Nm.

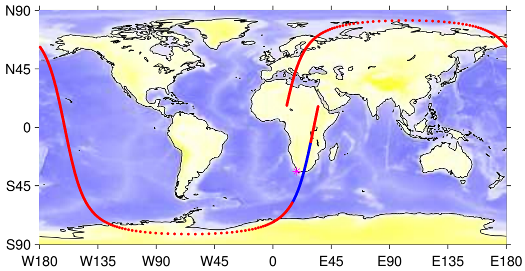


Fig. 3. Satellite footprint and target location (Cape Town).

The simulation experiments are designed in a way that enables observation of Cape Town. One of the passes close to the town will happen on August 30$^{\text{th}}$, 2014, when the satellite becomes available on sky at 4:57:24 and disappears below the horizon around twelve minutes later. The time in all the simulations is measured from this initial time. The figure 3 shows the satellite's footprint and the location of Cape Town.

The control objective is to align the tracked target on the surface of the Earth to the desired target — the control objective is supposed to be given through image interaction. The simulation scenario requires from both types of the controllers to align the target to the desired target. The situation in the case of target tracking is shown in the figures 4a and 4b, and the situation in the case of oriented-target tracking is shown in the figures 4a and 4b.

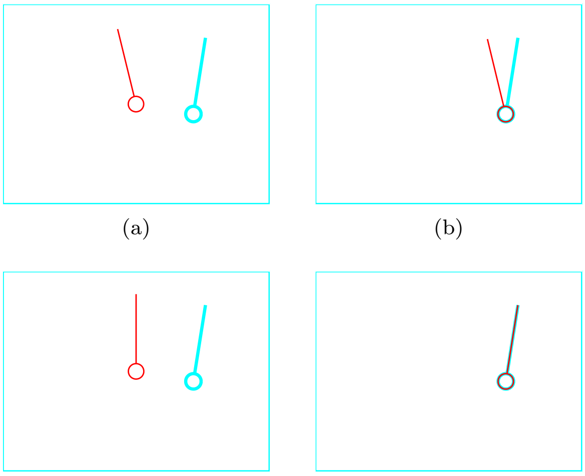
The figure 5 show the tracking performance of the target tracking controller to two changes of reference. In the figure 6 the applied torques to the system are shown. The results for oriented-target tracking controller are shown in the figures 7 and 8. The obtained results prove that the presented control algorithm can track the target, even though the control action is bounded and the system has an inherent 5 s delay.

6. CONCLUSION

In this paper an implementation of predictive functional control for image-based satellite attitude control problem was considered. Two different control problems were studied: target tracking problem and oriented-target tracking problem — in both cases the desired objective can be defined through in-image interaction. The PFC algorithm was developed based on the kinematic model of the satellite attitude. The results confirm that the predictive functional control can achieve satisfatory performance, even in the presence of input constraints and system delays. High performance was achieved without any time-consuming computation, and this makes a PFC an appealing approach for real-time applications.

ACKNOWLEDGEMENTS

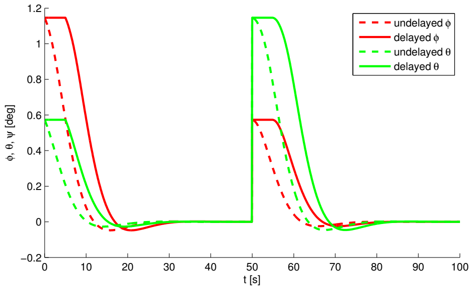
The Centre of Excellence for Space Sciences and Technologies SPACE-SI is an operation partly financed by the European Union, European Regional Development Fund and Republic of Slovenia, Ministry of Education, Science and Sport.



(a) (b)

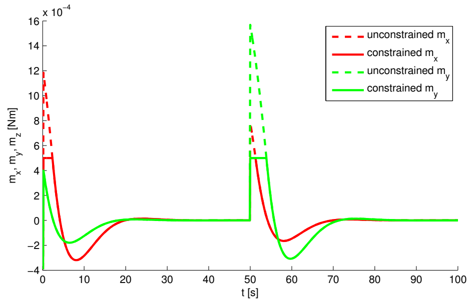
(c) (d)

Fig. 4. Target tracking (top row) and oriented-target tracking (bottom row). The control goal is to align the position of the tracked point on the ground, expressed in the image (slim red vector), to the desired position in image (thick cyan vector). Initial situation (left column) and the state after the transition time (right column). Notice that in case of target tracking (top right) only position (marked with a circle) is aligned.



(a)

Fig. 5. Error signals in the case of target tracking.

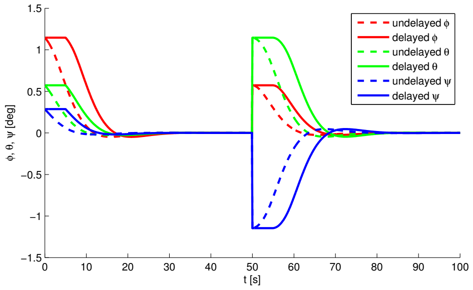


(a)

Fig. 6. Control torques in the case of target tracking.

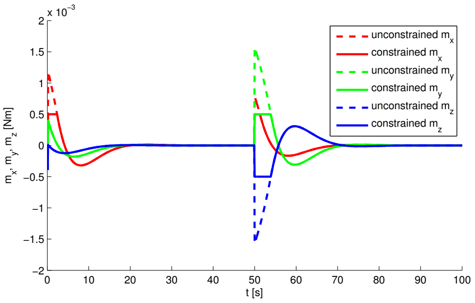
REFERENCES

Bourquardez, O. and Chaumette, F. (2007). Visual servoing of an airplane for alignment with respect to a runway. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 07, 1330–1335. IEEE, Roma, Italy.



(a)

Fig. 7. Error signals in the case of oriented-target tracking.



(a)

Fig. 8. Control torques in the case of oriented-target tracking.

Bošnak, M., Matko, D., and Blažič, S. (2011). Quadrocopter hovering using position-estimation information from inertial sensors and a high-delay video system. *Journal of Intelligent & Robotic Systems*, 67(1), 43–60.

Bošnak, M., Matko, D., and Blažič, S. (2012). Quadrocopter control using an on-board video system with off-board processing. *Robotics and Autonomous Systems*, 60(4), 657–667.

Brosilow, C. and Joseph, B. (2002). Two-degree of freedom internal model control. In *Techniques of Model-Based Control*, 65–84. Prentice Hall.

Camacho, E.F. and Bordons, C. (2004). *Model predictive control*. Springer, 2 edition.

Corke, P. (2011). *Robotics, vision and control: fundamental algorithms in MATLAB*. Springer, Berlin Heidelberg.

De Luca, A., Oriolo, G., and Robuffo Giordano, P. (2008). Feature depth observation for image-based visual servoing: theory and experiments. *The International Journal of Robotics Research*, 27(10), 1093–1116.

Eberli, D., Scaramuzza, D., Weiss, S., and Siegwart, R. (2010). Vision based position control for MAVs using one single circular landmark. *Journal of Intelligent and Robotic Systems*, 61(1-4), 495–512.

Kelly, R., Carelli, R., Nasisi, O., Kuchen, B., and Reyes, F. (2000). Stable visual servoing of camera-in-hand robotic systems. *IEEE/ASME Transactions on Mechatronics*, 5(1), 39–48.

Kim, Y.H., Lee, S.W., Yang, H.S., and Shell, D.A. (2011). Toward autonomous robotic containment booms: visual servoing for robust inter-vehicle docking of surface vehicles. *Intelligent Service Robotics*, 5(1), 1–18.

Klančar, G., Blažič, S., Matko, D., and Mušič, G. (2012). Image-based attitude control of a remote sensing satellite. *Journal of Intelligent and Robotic Systems*, 66(3), 343–357.

Klančar, G., Kristan, M., Kovačič, S., and Orqueda, O. (2004). Robust and efficient vision system for group of cooperating mobile robots with application to soccer robots. *ISA transactions*, 43(3), 329–342.

Kosmopoulos, D. (2011). Robust Jacobian matrix estimation for image-based visual servoing. *Robotics and Computer-Integrated Manufacturing*, 27(1), 82–87.

Lepetič, M., Škrjanc, I., Chiacchiarini, H.G., and Matko, D. (2003). Predictive functional control based on fuzzy model: comparison with linear predictive functional control and PID control. *Journal of Intelligent and Robotic Systems*, 36(4), 467–480.

Maciejowski, J.M. (2002). *Predictive control: with constraints*. Pearson Education.

Meingast, M., Geyer, C., and Sastry, S. (2004). Vision based terrain recovery for landing unmanned aerial vehicles. In *43rd Conference on Decision and Control (CDC)*, 1670–1675. IEEE.

Morari, M. and Lee, J.H. (1999). Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23(4-5), 667–682.

NIMA (2000). Department of defense world geodetic system 1984. Technical report, National Imagery and Mapping Agency (NIMA), St. Louis, MO, USA.

Normey-Rico, J.E., Gómez-Ortega, J., and Camacho, E.F. (1999). A Smith-predictor-based generalised predictive controller for mobile robot path-tracking. *Control Engineering Practice*, 7(6), 729–740.

Renner, U. and Buhl, M. (2008). High precision interactive Earth observation with Lapan-Tubsat. In *Proceedings of the 4S Symposium Small Satellites, Systems and Services*, 1–7. Rhodos, Greece.

Richalet, J. (1993). Industrial applications of model based predictive control. *Automatica*, 29(5), 1251–1274.

Richalet, J. and O'Donovan, D. (2009). *Predictive functional control: principles and industrial applications*. Springer.

Rouhani, R. and Mehra, R.K. (1982). Model algorithmic control (MAC); basic theoretical properties. *Automatica*, 18(4), 401–414.

Shakernia, O., Ma, Y., Koo, T.J., Hespanha, J., and Sastry, S.S. (1999). Vision guided landing of an unmanned air vehicle. In *38th Conference on Decision and Control (CDC)*, 4143–4148. IEEE, Phoenix, AZ, USA.

von Hundelshausen, F. (2004). *Computer vision for autonomous mobile robots*. Ph.D. thesis, Freien Universität Berlin.

Škrjanc, I. and Matko, D. (2000). Predictive functional control based on fuzzy model for heat-exchanger pilot plant. *IEEE Transactions on Fuzzy Systems*, 8(6), 705–712.

Wertz, J.R. (1978). *Spacecraft Attitude Determination and Control*. Springer.

Zdešar, A., Klančar, G., Mušič, G., Matko, D., and Škrjanc, I. (2013). Design of the image-based satellite attitude control algorithm. In *XXIV International Conference on Information, Communication and Automation Technologies (ICAT)*, 1–8. Sarajevo.

Zhang, G.L. (2009). *Image-based visual servoing with hybrid camera configuration for robust robotic grasping*. Msc, University of British Columbia.