Preliminary Results on Waypoint Tracking for Spacecraft with Actuator Constraints

Espen Oland
Department of Electrical Engineering
UiT - The Arctic University of Norway
Narvik, Norway
espen.oland@uit.no

Tom Stian Andersen
Department of Electrical Engineering
UiT - The Arctic University of Norway
Narvik, Norway
tom.s.andersen@uit.no

***Abstract*—This paper presents preliminary results on how to perform waypoint tracking with spacecraft with actuator constraints. It considers a simplified spacecraft model and can be considered a deep space model, and shows how to perform waypoint tracking with only one main thruster together with full attitude control. As the spacecraft reaches close to the waypoint during a deceleration phase that makes the speed go towards zero, reaction control thrusters are used to make the remaining velocity error go to zero achieving the control objective.**

***Keywords*—Rendezvous, waypoint tracking, spacecraft, actuator constraints**

I. Introduction

The rendezvous problem was ranked first among the top technology challenges in the 2011 technology roadmap by NASA on the topic of "Robotics, Tele-Robotics, and Autonomous Systems" [1]. To that end there is a need for research on this topic, which is important to pave the way for future space exploration.

One of the most basic problems is to perform a translational motion from a point A to a point B. Assuming a fully actuated vehicle with full translational control in all axes, this can easily be achieved using a standard PID controller using position feedback. By including actuator constraints this becomes a little more involved. From a design point of view, it does not make sense to fit a spacecraft with large translational thrusters along each axis, such that the most natural design will include one main thruster together with attitude control actuators (e.g. reaction wheels) as well as reaction control thrusters for station-keeping and small translational maneuvers.

This means that in order to perform a translational maneuver from a point A to point B using one main thruster, the first objective is to point the thrust direction towards the waypoint. Then after accelerating to a desired velocity (or simply for a given time), the spacecraft must be rotated 180° such that the thruster points along the direction of motion and must be used to decelerate until the waypoint is reached. For terrestial waypoint tracking in cases such as unmanned aerial vehicles (UAVs), aircraft, ships, and underwater vehicles, there will always be a convenient viscous damping that helps controlling the speed and bounding the velocity components during the maneuver. In space, nothing helps you brake; such that special maneuvers are required to achive the control objective.

The topic of waypoint tracking has received much attention throughout the ages with applications such as ships [2], aircraft [3], underwater vehicles [4], UAVs [5], missiles [6], as well as spacecraft [7], [8], [9], [10].

Phillips et al. show in [7] how to perform close proximity maneuvers while accounting for propellant impingement. This is achieved using a series of waypoints that is tracked by a fully actuated spacecraft. Guo et al. show in [8] how to perform a waypoint-optimized Mars landing and the authors apply the Zero-Effort-Miss/Zero-Effort-Velocity method and account for nonlinear actuator constraints. Furfaro and Linares apply in [9] the ZEM/ZEV feedback approach in conjunction with reinforcement learning to perform obtain a precise planetary landing. In many ways, the problem of performing a precise landing can be considered similar to the problem of performing waypoint tracking with constrained actuation.

This paper builds on the previous research mentioned above, as well as work performed by the authors [5], which show how to perform waypoint tracking and collision avoidance by properly defining desired orientation parameterized as quaternions. This approach is here applied for waypoint guidance for spacecraft.

This paper is structured as follows: Section II presents the modeling used in this paper, where the spacecraft dynamics is simplified. Section IV explains the main ideas of finding the desired orientation and how to perform the waypoint maneuvers. Section V detail the controllers used in this work. Section VI presents simulations of a spacecraft that tracks a series of waypoints using one main thruster for large translational maneuvers, where small reaction control thrusters are activated close to the waypoint. The paper is then wrapped up with a conclusion discussing the results, and future work.

A. Problem Statement

Given a spacecraft with one main thruster for translational control, reaction wheels for attitude control and reaction control thrusters for final position control, design a guidance and control system that enables the spacecraft to track a series of waypoints.

II. Modeling

A. Notation

This section is similar to the first author's previous works such as [11]. The time derivative of a vector is denoted as $\dot{\mathbf{x}} = d\mathbf{x}/dt$ and the Euclidean length is written as $||\mathbf{x}|| = \sqrt{\mathbf{x}^\top \mathbf{x}}$. Superscript denotes the reference frame of a vector. The rotation matrix is denoted $\mathbf{R}_a^c \in \mathcal{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3\times 3} : \mathbf{R}^\top \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\}$, which rotates a vector from frame a to frame c and where $\mathbf{I}$ denotes the identity matrix. The angular velocity vector is denoted $\boldsymbol{\omega}_{a,c}^e$, which represents the angular velocity of frame c relative to frame a referenced in frame e. Angular velocities between different frames can be added together as $\boldsymbol{\omega}_{a,f}^e = \boldsymbol{\omega}_{a,c}^e + \boldsymbol{\omega}_{c,f}^e$. The time derivative of the rotation matrix is found as $\dot{\mathbf{R}}_a^c = \mathbf{R}_a^c \mathbf{S}(\boldsymbol{\omega}_{c,a}^a)$ where the cross product operator $\mathbf{S}(\cdot)$ is such that for two vectors $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^3$, $\mathbf{S}(\mathbf{v}_1)\mathbf{v}_2 = \mathbf{v}_1 \times \mathbf{v}_2$, $\mathbf{S}(\mathbf{v}_1)\mathbf{v}_2 = -\mathbf{S}(\mathbf{v}_2)\mathbf{v}_1$, $\mathbf{S}(\mathbf{v}_1)\mathbf{v}_1 = \mathbf{0}$ and $\mathbf{v}_1^\top \mathbf{S}(\mathbf{v}_2)\mathbf{v}_1 = 0$.

The rotation matrix can be parameterized using quaternions, where the quaternion that represents a rotation from frame a to frame c is denoted $\mathbf{q}_{c,a} \in \mathcal{S}^3 = \{\mathbf{q} \in \mathbb{R}^4 : \mathbf{q}^\top \mathbf{q} = 1\}$ and can be written as $\mathbf{q}_{c,a} = \begin{bmatrix}\eta_{c,a} & \boldsymbol{\epsilon}_{c,a}^\top\end{bmatrix}^\top = \begin{bmatrix}\cos\left(\frac{\vartheta_{c,a}}{2}\right) & \mathbf{k}_{c,a}^\top \sin\left(\frac{\vartheta_{c,a}}{2}\right)\end{bmatrix}^\top$ which performs a rotation of an angle $\vartheta_{c,a}$ around the unit vector $\mathbf{k}_{c,a}$, and the inverse quaternion is defined as $\mathbf{q}_{a,c} = \begin{bmatrix}\eta_{c,a} & -\boldsymbol{\epsilon}_{c,a}^\top\end{bmatrix}^\top$. The scalar part is denoted $\eta_{c,a}$ and the vector part as $\boldsymbol{\epsilon}_{c,a} \in \mathbb{R}^3$, enabling the rotation matrix to be constructed as $\mathbf{R}_a^c = \mathbf{I} + 2\eta_{c,a}\mathbf{S}(\boldsymbol{\epsilon}_{c,a}) + 2\mathbf{S}^2(\boldsymbol{\epsilon}_{c,a})$. Composite rotations are found using the quaternion product as $\mathbf{q}_{c,e} = \mathbf{q}_{c,a} \otimes \mathbf{q}_{a,e} = \mathbf{T}(\mathbf{q}_{c,a})\mathbf{q}_{a,e}$ with

$$\mathbf{T}(\mathbf{q}_{c,a}) = \begin{bmatrix}\eta_{c,a} & -\boldsymbol{\epsilon}_{c,a}^\top \\ \boldsymbol{\epsilon}_{c,a} & \eta_{c,a}\mathbf{I} + \mathbf{S}(\boldsymbol{\epsilon}_{c,a})\end{bmatrix}, \tag{1}$$

which ensures that the resulting quaternion maintains the unit length property

III. Translational Dynamics

Let the translational dynamics of the spacecraft be defined as

$$\dot{\mathbf{p}}^i = \mathbf{v}^i \tag{2}$$

$$\dot{\mathbf{v}}^i = \frac{1}{m}\mathbf{R}_b^i \mathbf{f}_{Thruster}^b + \frac{1}{m}\mathbf{R}_b^i \mathbf{f}_{RCT}^b \tag{3}$$

where $\mathbf{p}^i$ denotes the position of the spacecraft in the inertial frame, $\mathbf{v}^i$ denotes the velocity, m is the spacecraft mass, $\mathbf{R}_b^i$ is the rotation matrix from the body frame to the inertial frame, $\mathbf{f}_{Thruster}^b = \begin{bmatrix}T & 0 & 0\end{bmatrix}^\top$ is the thruster in the body frame, with the total thrust, T, aligned with the $\mathbf{x}^b$ axis. Additionally, there are a set of reaction control thrusters (RCT), $\mathbf{f}_{RCT}^b = \begin{bmatrix}f_x & f_y & f_z\end{bmatrix}^\top$ that allows for small translational maneuvers such as station keeping and obtaining a perfect stop at a desired waypoint.

A. Rotational Dynamics

The rotational dynamics can be described using quaternions, as this paper utilizes several different reference frames. The attitude dynamics of the body frame of the spacecraft relative to the inertial frame can be described as

$$\dot{\mathbf{q}}_{i,b} = \frac{1}{2}\mathbf{q}_{i,b} \otimes \begin{bmatrix}0 \\ \boldsymbol{\omega}_{i,b}^b\end{bmatrix} = \frac{1}{2}\mathbf{T}(\mathbf{q}_{i,b}) \begin{bmatrix}0 \\ \boldsymbol{\omega}_{i,b}^b\end{bmatrix}. \tag{4}$$

with the attitude dynamics as

$$\mathbf{J}\dot{\boldsymbol{\omega}}_{i,b}^b = -\mathbf{S}(\boldsymbol{\omega}_{i,b}^b)\mathbf{J}\boldsymbol{\omega}_{i,b}^b + \boldsymbol{\tau}^b \tag{5}$$

where $\mathbf{J} \in \mathbb{R}^{3\times 3}$ is the inertia matrix, $\boldsymbol{\omega}_{i,b}^b$ is the angular velocity of the spacecraft body frame relative to the inertial frame, while $\boldsymbol{\tau}^b$ is the moments acting on the body, which can be achieved using reaction wheels or similar actator for attitude control.

IV. Waypoint Guidance

A. Acceleration Phase

Now assume that the spacecraft has a position defined by $\mathbf{p}^i$ and it is desired to reach the waypoint $\mathbf{p}_{wp}^i$. Let an error function be defined as

$$\mathbf{e}^e = \mathbf{R}_i^e(\mathbf{p}_{wp}^i - \mathbf{p}^i) \tag{6}$$

$$\mathbf{e}^e = \mathbf{R}_i^e \mathbf{e}^i \tag{7}$$

where superscript e denotes a position error frame and $\mathbf{e}^e := \begin{bmatrix}||\mathbf{e}^i|| & 0 & 0\end{bmatrix}^\top$. To make the position error go to zero, the objective is simply to point the body frame along the $\mathbf{e}^e$ vector and move with a positive speed. Figure 1(a) shows the basic vectors and reference frames needed for for achieving waypoint tracking. The quaternion, angular velocity and rotation matrix that describes which direction to move can now be found as

$$\mathbf{q}_{i,e} = \begin{bmatrix}\eta_{i,e} & \boldsymbol{\epsilon}_{i,e}^\top\end{bmatrix}^\top = \begin{bmatrix}\sin\left(\frac{\vartheta_{i,e}}{2}\right) & \mathbf{k}_{i,e}^\top \cos\left(\frac{\vartheta_{i,e}}{2}\right)\end{bmatrix}^\top \tag{8}$$

$$\boldsymbol{\omega}_{i,e}^e = -\mathbf{S}^\dagger(\mathbf{e}^e)\mathbf{R}_i^e \mathbf{e}^i \tag{9}$$

$$\vartheta_{i,e} = \cos^{-1}\left(\frac{\mathbf{e}^e \cdot \mathbf{e}^i}{||\mathbf{e}^i||^2}\right) \tag{10}$$

$$\mathbf{k}_{i,e} = \frac{\mathbf{e}^e \times \mathbf{e}^i}{||\mathbf{e}^e \times \mathbf{e}^i||} \tag{11}$$

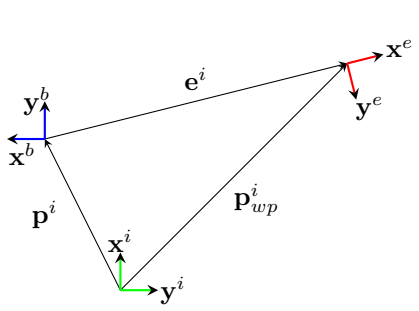
$$\mathbf{R}_e^i = \mathbf{I} + 2\eta_{i,e}\mathbf{S}(\boldsymbol{\epsilon}_{i,e}) + 2\mathbf{S}^2(\boldsymbol{\epsilon}_{i,e}). \tag{12}$$

It has been shown in [5] that a set around the origin of $\mathbf{e}^i$ is uniformly asymptotically stable by following this quaternion. The angular velocity (9) is found by differentiating (7) and noting that $\mathbf{S}^\dagger(\mathbf{e}^e)\dot{\mathbf{e}}^e = \mathbf{0}$ due to definition of reference frames.

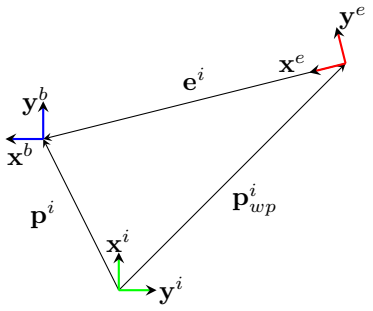
A simple translational thrust controller for the acceleration phase can be created as

$$T = \begin{cases} T_{max} & \text{if } ||\mathbf{v}^i|| \leq V_d \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

where $V_d \in \mathcal{L}_\infty$ is a desired speed, and T_{max} is the maximum thrust from the main thruster (i.e. bang-bang modulation).



(a) Acceleration phase.



(b) Deceleration phase

Fig. 1. Basic vectors and reference frames during waypoint tracking. Subfigure (a) shows the vectors during the acceleration phase, where the objective is to align the body frame along the position error frame, and translate along the $\mathbf{e}^i$ vector. Subfigure (b) shows the decelaration phase, where the objective is to align the body frame with the rotated error frame and decelerate until reaching the desired waypoint.

B. Deceleration Phase

After achieving a desired speed or accelerating for a given time, the orientation of the spacecraft must be rotated 180° to be able to use the main thruster for deceleration. This can be achieved by simply redefining the error vector $\mathbf{e}^i$ as shown in Figure 1(b). By pointing the body frame along the rotated position error frame, the main thruster will enable deceleration until the waypoint is reached. By just firing the main thruster several times will eventually make the speed go to zero, but there is also a desire to reach and stop at the waypoint. To that end, consider the basic equation of motion

$$2ad = v^2 - v_0^2 \tag{14}$$

where a is the acceleration, d is the distance, v is the speed, and v_0 is the intial speed. From this, the desired acceleration to decelerate can be found as

$$a_d = -\frac{||\mathbf{v}^i||}{2||\mathbf{e}^i||} \tag{15}$$

which will increase as $||\mathbf{e}^i|| \to 0$. This notion, enables the thrust to be found as

$$T = \begin{cases} T_{max} & \text{if } |ma_d| \geq T_{max} \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

This method enables the thrust only to be fired after the desired acceleration exceeds what can be produced by the main thruster, and has been shown to be quite effective in slowing down the spacecraft.

C. Misalignment

In a perfect world the waypoint tracking problem would now have been solved. However, misalignment during thruster firings will have a great impact on large translational maneuvers. In the following section, the attitude controller uses an integral term for the sole purpose of making the misalignment go close to zero during thruster firings. Due to this, there tends to be small velocity components that will make the spacecraft miss the desired waypoint. One way to remedy this, is to include another reference frame that helps to project all the velocity components onto the $\mathbf{x}^b$ axis. This can be achieved by defining

$$\mathbf{v}^d = \mathbf{R}^d_i \mathbf{v}^i = \mathbf{R}^d_e \mathbf{R}^e_i \mathbf{v}^i \tag{17}$$

here superscript d denotes the desired frame where $\mathbf{v}^d := \begin{bmatrix} \pm||\mathbf{v}^i|| & 0 & 0 \end{bmatrix}^\top$. One important notion with this definition, is that the rotation matrix is defined relative to the position error frame, something that allows the rotation to make the velocity components go to zero subsume (or take priority over) the position error quaternion. This is discussed in great detail in [5]. The rotation matrix from the desired frame to the position error frame can now be constructed as

$$\mathbf{q}_{e,d} = \begin{bmatrix} \eta_{e,d} & \boldsymbol{\epsilon}^\top_{e,d} \end{bmatrix}^\top = \begin{bmatrix} \sin\left(\frac{\vartheta_{e,d}}{2}\right) & \mathbf{k}^\top_{e,d} \cos\left(\frac{\vartheta_{e,d}}{2}\right) \end{bmatrix}^\top \tag{18}$$

$$\boldsymbol{\omega}^d_{e,d} = \mathbf{S}^\dagger(\mathbf{v}^d)\left(\mathbf{R}^d_e \mathbf{S}(\boldsymbol{\omega}^e_{i,e})\mathbf{R}^e_i \mathbf{v}^i - \mathbf{R}^d_i \dot{\mathbf{v}}^i\right) \tag{19}$$

$$\vartheta_{e,d} = \cos^{-1}\left(\frac{\mathbf{v}^d \cdot \mathbf{v}^e}{||\mathbf{v}^e||^2}\right) \tag{20}$$

$$\mathbf{k}_{e,d} = \frac{\mathbf{v}^d \times \mathbf{v}^e}{||\mathbf{v}^d \times \mathbf{v}^e||} \tag{21}$$

$$\mathbf{R}^e_d = \mathbf{I} + 2\eta_{e,d}\mathbf{S}(\boldsymbol{\epsilon}_{e,d}) + 2\mathbf{S}^2(\boldsymbol{\epsilon}_{e,d}). \tag{22}$$

Here, the angular acceleration vector becomes a little more involved due to the dependency on the position error frame, but the reader can easily derive the same expression by following the proof in [5] for Lemma 2. The desired quaternion and angular velocity can now be constructed as

$$\mathbf{q}_{i,d} = \mathbf{q}_{i,e} \otimes \mathbf{q}_{e,d} \tag{23}$$

$$\boldsymbol{\omega}^d_{i,d} = \mathbf{R}^d_e \boldsymbol{\omega}^e_{i,e} + \boldsymbol{\omega}^d_{e,d}. \tag{24}$$

where the misalignment method can be suppressed when not used by setting $\mathbf{q}_{e,d} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^\top$ and $\boldsymbol{\omega}^d_{e,d} = \mathbf{0}$.

V. Controllers

A. Attitude Controller

The output from the waypoint guidance method is a desired quaternion and angular velocity, denoted as $\mathbf{q}_{i,d} = \begin{bmatrix} \eta_{i,d} & \boldsymbol{\epsilon}_{i,d} \end{bmatrix}^\top$ and $\boldsymbol{\omega}^d_{i,d}$. To track these signals, a simple PID

controller can be desigined. First, let the quaternion error be found as

$$\mathbf{q}_{d,b} = \mathbf{q}_{d,i} \otimes \mathbf{q}_{i,d} \tag{25}$$

which has the kinematics as

$$\dot{\mathbf{q}}_{d,b} = \frac{1}{2}\mathbf{T}(\mathbf{q}_{d,b}) \begin{bmatrix} 0 \\ \boldsymbol{\omega}^b_{d,b} \end{bmatrix} \tag{26}$$

where $\boldsymbol{\omega}^b_{d,b} = \boldsymbol{\omega}^b_{i,b} - \mathbf{R}^b_d \boldsymbol{\omega}^d_{i,d}$.

It is well-known that quaternions have unit lenght, meaning that as $\boldsymbol{\epsilon}_{d,b} \to \mathbf{0}$, it follows that $\eta_{d,b} \to \pm 1$. Hence, the vector part of the quaternion can be chosen as the error signal for the PID controller, which therefore can be designed as

$$\boldsymbol{\tau}^b = -k_p \boldsymbol{\epsilon}_{d,b} - k_i \int_0^t \boldsymbol{\epsilon}_{d,b} d\tau - k_d \dot{\boldsymbol{\epsilon}}_{d,b}, \tag{27}$$

where k_p, k_i, k_d are three positive gains to be chosen. Note that the quaternion derivative is extracted from the quaternion kinematics. This control law will enable the quaternion error go to zero, and by properly defining the desired quaternion, the control objective will be achieved. Actuation limitations for the attitude controller has not been considered, as the main contribution of this work lies in the translational control method. The control effort can easily be reduced by reducing the gains, at the cost of longer rotational maneuvers.

B. Fine Position Controller

After performing the main translational motion using the main thruster, there will always be some perturbations that calls for a position controller to make the position and velocity converge to zero. Assuming that the spacecraft have full translational control using reaction control thrusters, a control law that can make the final position error go to zero can be proposed as

$$\mathbf{f}^b_{RCT} = m\mathbf{R}^b_i(-\kappa_p \mathbf{e}^i - \kappa_d \mathbf{v}^i) \tag{28}$$

where $\kappa_p, \kappa_d > 0$ are two positive gains. The main motivation for including this control law, is that with a very large thruster it is only possible to make the speed go below a given threshold, which will make the spacecraft drift with e.g. 0.1 m/s away from the desired waypoint, such that the additional sets of thrusters allows the speed to go to zero.

Assume that there is a configuration of six thrusters (or sets of thrusters) mounted such that they can produce forces in each axis. Let $\mathbf{T}^b_{RCT} = \begin{bmatrix} f_1 & f_2 & f_3 & f_4 & f_5 & f_6 \end{bmatrix}^\top$, then the desired thruster firings from (28) can be mapped to the individual thrusters as

$$\mathbf{T}^b_{RCT} = \mathbf{B}^\dagger \mathbf{f}^b_{RCT} \tag{29}$$

where $\dagger$ denotes the Moore-Penrose pseudoinverse, and the allocation matrix can be defined as

$$\mathbf{B} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}, \tag{30}$$

then each thruster can be modulated using bang bang modulation.

VI. Simulations

It is now time to put everything together. Consider N waypoints contained in an array $\mathbf{P}^i_{wp} \in \mathbb{R}^{N\times 3}$. Let V_d denote the desired (or maximum) speed, $\delta > 0$ is the radius of a sphere around the desired waypoint, and $\delta_p > 0$ and δ_v are the desired position and velocity accuracy when performing fine position control. Then an algorithm showing the main ideas can be proposed as shown in Algorithm 1.

Algorithm 1: Waypoint Tracking

```
for i ≤ N do
    Select i'th waypoint;
    while ||e^i|| > δ do
        if ||e^i(t)|| ≥ 0.8||e^i(0)|| then
            Acceleration←True;
        else
            Acceleration←False;
        Find desired attitude;
        Control attitude;
        if ||ε_{d,b}|| < 1 · 10^{-6} then
            if Acceleration = True then
                if ||v^i|| ≤ V_d then
                    T ← T_max;
                else
                    T ← 0;
            else
                if m||v^i|| / (2||e^i||) ≥ T_max then
                    T ← T_max;
                else
                    T ← 0;
        else
            T ← 0;
    while ||e^i|| ≥ δ_p and ||v^i|| ≥ δ_v do
        Fine position control using RCT;
    i ← i + 1;
```

Now consider a spacecraft that shall follow a series of waypoints with the parameters and physical properties as given in Table I. For this simulation, we ignore the fine position control, and only make the velocity component go to zero close to the waypoint before switching to the next waypoint. The list of waypoints are given in Table II. The switching mechanism to determine the acceleration state, is simply set to 20% of the initial position error. This means that the first 20% of translational motion is spent on accelerating the spacecraft, while the remaining 80% is spent on coasting and breaking down the velocity to zero.

Figure 2 shows a 3D figure of the behavior of the spacecraft during the waypoint tracking, where the red line shows the $\mathbf{x}^b$ axis (or thrust direction), which is changed throughout the maneuvers. Note that it able to change its orientation such

TABLE I. Parameters and values used for the simulation.

Parameter	Value	Unit
m	100	kg
$\mathbf{J}$	$20\mathbf{I}$	kgm^2
k_p	20	
k_i	2	
k_d	20	
κ_p	0	
κ_d	5	
T_{max}	20	N
f_{max}	0.5	N

TABLE II. List of waypoints.

Waypoint	x	y	z
WP-1	40000	0	0
WP-2	40000	40000	0
WP-3	40000	40000	40000

that it breaks down the speed before converging close to the desired waypoint.

The thrusters are shown in Figure 3. The top subplot shows the main thruster which is able to produce 0 or 20N. The six following subplots are the reaction control thrusters, which are only activated close to the desired waypoint and only has 0.5 N thrust.

The velocity and position error during the operation are given in Figure 4 and Figure 5, which goes (close) to zero during each maneuver. Even better position tracking can be achieved by activating the position tracking controller during fine position control, but it will require more time to properly converge before switching to the next waypoint.

The rotational dynamics is shown in Figure 6, where the top subplot shows the quaternion, the middle subplot shows the angular velocity while the bottom plot shows the torques produced by the attitude control system.

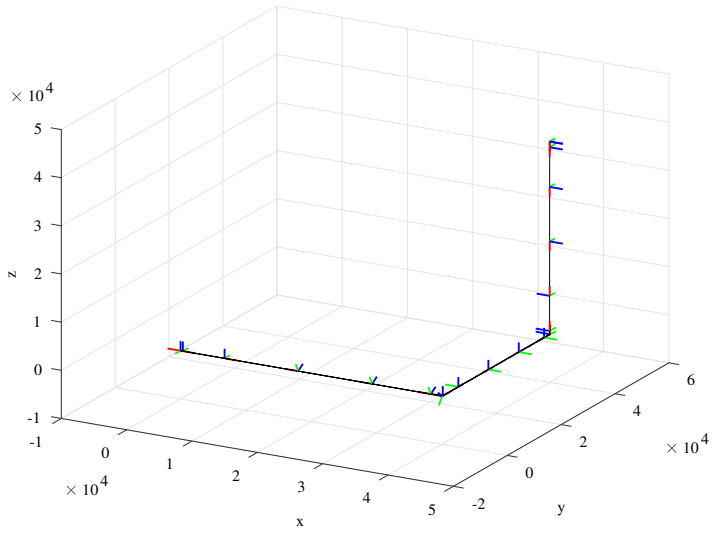


Fig. 2. Position and orientation visualization during the maneuver, where the thrust direction is shown as the the red arrow.

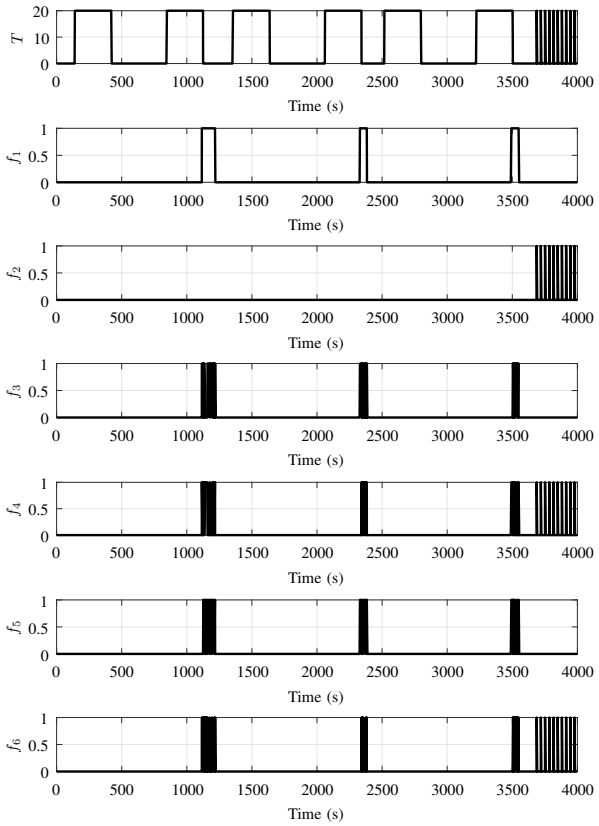


Fig. 3. Thruster firings to control the attitude.

Conclusion

This paper has presented preliminary results on performing waypoint tracking for spacecraft constrained to have one main thruster for performing large translational motions. The main idea of using the main thruster to break the spacecraft during the maneuvers has been achieved through the definition of the desired attitude. Future work on improving this method is to re-cast the algorithm into a state machine environment to handle the switching between the different modes, as well as augment the mathematical model with basis in spacecraft formation dynamics (with perturbations) to obtain more realistic results.

Acknowledgement

The authors would like to thank the reviewers for the good feedback and recommendations that helped improve this paper.

References

[1] M. Pavone, B. Açıkmeşe, I. A. Nesnas, and J. A. Starek, "Spacecraft autonomy challenges for next-generation space missions," *Advances in Control System Technology for Aerospace Applications*, vol. 460, pp. 1–48, 2015.

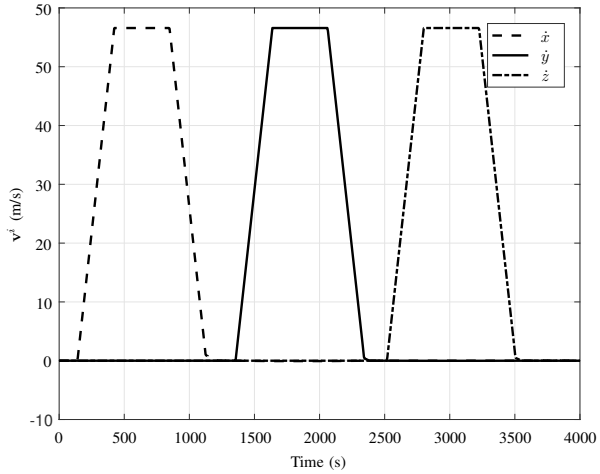


Fig. 4. Velocity profile during waypoint maneuvers.

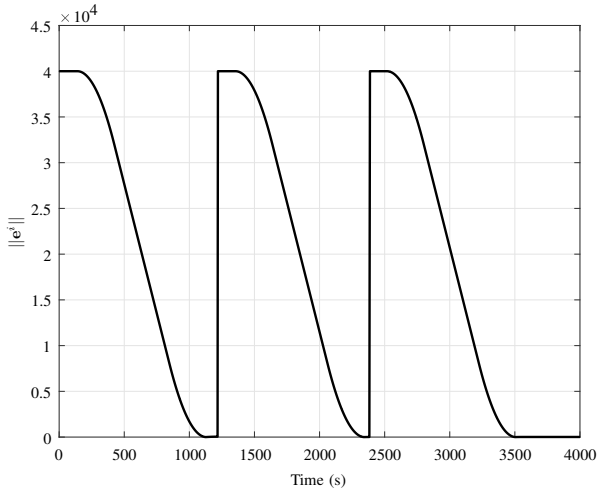


Fig. 5. Position error during waypoint tracking.

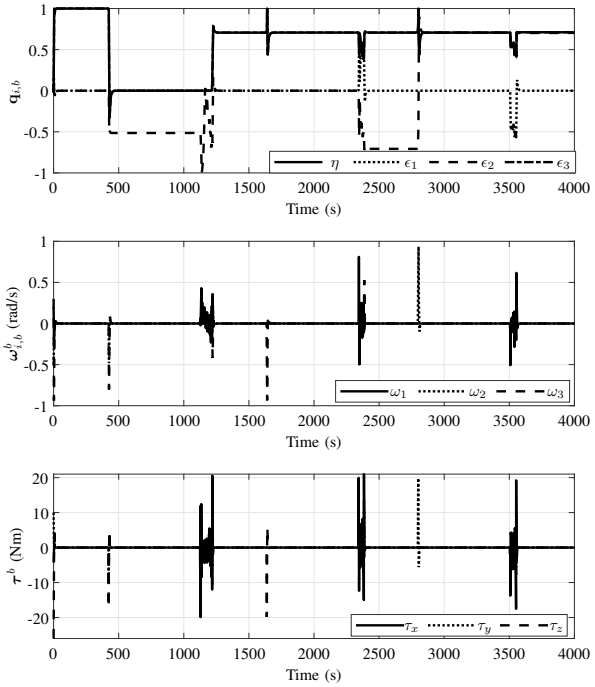


Fig. 6. Attitude dynamics during the operation.

[2] E. Børhaug and K. Y. Pettersen, "Adaptive way-point tracking control for underactuated autonomous vehicles," in *Proceedings of the 44th IEEE Conference on Decision and Control*, (Seville, Spain), 2005.

[3] A. Babaei and M. Mortazavi, "Three-Dimensional Curvature-Constrained Trajectory Planning Based on In-Flight Waypoints," *Journal of Aircraft*, vol. Vol. 47, No. 4, pp. pp. 1391–1398, 2010.

[4] A. P. Aguiar and A. M. Pascoal, "Dynamic positioning and way-point tracking of underactuated AUVs in the presence of ocean currents," in *Proceedings of the 41st IEEE Conference on Decision and Control*, (Las Vegas, NV, USA), 2002.

[5] E. Oland, T. S. Andersen, and R. Kristiansen, "Subsumption architecture applied to flight control using composite rotations," *Automatica*, vol. 69, pp. 195–200, 2016.

[6] H. Zhou, E. M. Khalil, T. Rahman, and W. Chen, "Waypoints following guidance for surface-to-surface missiles," *International Journal of Aeronautical & Space Science*, vol. 19, pp. 183–195, 2018.

[7] J. M. Phillips, L. E. Kavraki, and N. Bedrossian, "Spacecraft rendezvous and docking with real-time randomized optimization," in *Proceedings of AIAA Guidance, Navigation, and Control Conference*, (Austin, TX, USA), 2003.

[8] Y. Guo, M. Hawkins, and B. Wie, "Waypoint-optimized zero-effort-miss/zero-effort-velocityfeedback guidance for mars landing," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 3, pp. 799–809, 2013.

[9] R. Furfaro and R. Linares, "Waypoint-based generalized zem/zev feedback guidance for planetary landing via a reinforcement learning approach," *Dynamics and Control of Space Systems, DyCoSS 2017*, vol. 161, pp. 401–416, 2017.

[10] R. Furfaro, R. Ruggiero, F. Topputo, M. Lovera, and R. Linares, "Waypoint-optimized closed-loop guidance for spacecraft rendezvous in relative motion," *Advances in the Astronautical Sciences*, vol. 162, pp. 2651–2666, 2018.

[11] E. Oland, "A command-filtered backstepping approach to autonomous inspections using a quadrotor," in *Proceedings of the 24th Mediterranean Conference on Control and Automation*, (Athens, Greece), 2016.