This is a repository copy of *Model predictive control system design and implementation for spacecraft rendezvous*.

Article:

Model Predictive Control System Design and Implementation for Spacecraft Rendezvous

E. N. Hartley[a,*], P. A. Trodden[b,1], A. G. Richards[b], J. M. Maciejowski[a]

[a]*University of Cambridge Department of Engineering, Trumpington Street, Cambridge. CB2 1PZ. United Kingdom.*
[b]*Dept. of Aerospace Engineering, University of Bristol, Queens Building, University Walk, Bristol. BS8 1TR. United Kingdom.*

Abstract

This paper presents the design and implementation of a model predictive control (MPC) system to guide and control a chasing spacecraft during rendezvous with a passive target spacecraft in an elliptical or circular orbit, from the point of target detection all the way to capture. To achieve an efficient system design, the rendezvous manœuvre has been partitioned into three main phases based on the range of operation, plus a collision-avoidance manœuvre to be used in event of a fault. Each has its own associated MPC controller. Linear time varying models are used to enable trajectory predictions in elliptical orbits, whilst a variable prediction horizon is used to achieve finite-time completion of manœuvres, and a 1-norm cost on velocity change minimises propellant consumption. Constraints are imposed to ensure that trajectories do not collide with the target. A key feature of the design is the implementation of non-convex constraints as switched convex constraints, enabling the use of convex linear and quadratic programming. The system is implemented using commercial-off-the-shelf tools with deployment through the use of automatic code generation in mind, and validated by closed-loop simulation. A significant reduction in total propellant consumption in comparison with a baseline benchmark solution is observed.

Keywords: Autonomous vehicles; constraints; embedded systems; model-based control; optimization; path planning; predictive control; spacecraft autonomy

1. Introduction

Rendezvous is an important part of current and future space missions. For example, in the Mars Sample Return (MSR) mission (Régnier et al., 2005; Mura, 2007; Beaty et al., 2008) there will be a rendezvous and capture manœuvre, where a soil sample, enclosed in a passive container, must be captured by an orbiting chaser craft. Due to the distances involved, and the resulting communication delays (up to 20 min in each direction), in combination with the levels of accuracy required to correctly perform the capture (tolerance of approximately 20 cm) there is a particularly strong motivation to perform this operation autonomously.

---

[*]Corresponding author
Email addresses: `enh20@eng.cam.ac.uk` (E. N. Hartley), `paul.trodden@ed.ac.uk` (P. A. Trodden), `arthur.richards@bristol.ac.uk` (A. G. Richards), `jmm@eng.cam.ac.uk` (J. M. Maciejowski)
[1]P. A. Trodden is now at School of Mathematics, University of Edinburgh, The King's Buildings, Edinburgh. EH9 3JZ. United Kingdom

This paper describes the design and implementation of a Model Predictive Control (MPC) system (e.g. Maciejowski (2002); Camacho & Bordons (2004); Rawlings & Mayne (2009)) for the rendezvous and capture associated with the MSR mission, carried out as part of the ESA ORCSAT (On-line Reconfiguration Control System and Avionics Technologies) project. The MPC control system is designed to be used from the point of target detection to the point of target capture and to function in both circular and elliptical orbits.

Whilst MPC has its origins in the process industries (Qin & Badgwell, 2003), it has recently been applied to vehicle manœuvre trajectory planning problems (including short-range spacecraft rendezvous scenarios (Richards & How, 2003a; Larsson et al., 2006; Breger & How, 2006), UAV scenarios (Shim et al., 2003; Richards & How, 2006; Almeida, 2008), spacecraft formation flying scenarios (Manikonda et al., 1999; Breger & How, 2007), and spacecraft attitude control applications (Manikonda et al., 1999; Hegrenæs et al., 2005; Wood & Chen, 2008)). In addition, Bodin et al. (2011) indicates that the MPC formulation of Larsson et al. (2006) has been used successfully for trajectory tracking on a real test mission.

Unlike the works mentioned above, which have focused on application of MPC to individual spacecraft manœuvres or to tracking problems, this work considers the use of MPC end-to-end from the point of target detection all the way until target capture. The design presented explicitly considers implementation of MPC within the constraints of a spacecraft avionics architecture, although this paper focuses purely on the MPC design aspects of the ORCSAT project.

Since minimising propellant consumption is critical to space operations, a controller based on optimising fuel usage is attractive. It follows naturally from early work on trajectory optimisation (Hohmann, 1925; Kaplan, 1976; Prussing, 1991; Battin, 1999) to move the optimiser onboard the spacecraft in order to circumvent communication delays. The HARVD control system (Kerambrun et al., 2008) goes some way towards this by selecting guidance trajectories and nominal control actuation sequences from a large library of pre-computed optimal manœuvres. These are augmented with corrective manœuvres based on current navigation information to ensure faithful tracking of the desired paths.

Model predictive control possesses some attractive characteristics as a candidate guidance and control framework. Firstly, because model predictive control re-plans the optimal trajectory at each sampling instance, it is conceivable that propellant consumption in the presence of uncertainty could be reduced in comparison to tracking a pre-computed, nominal trajectory as performed by HARVD and on the state-of-the-art Automated Transfer Vehicle (ATV) (Ganet et al., 2002). Secondly, due to the implicit way in which the MPC problem is specified, carrying an exhaustive library of manœuvres for every possible initial configuration is not necessary. Thirdly, rendezvous operations typically have many constraints, including limited thruster capability, sensing regions, requirements for safe operation limits and so on. The intrinsic ability of MPC to handle constraints, and the ability to consider the control objective to be achievement of a trajectory end-point rather than a complete trajectory, alongside the availability of well-researched models of relative dynamics that can be used for prediction (Clohessy & Wiltshire, 1960; Tschauner, 1967; Carter, 1998; Battin, 1999; Melton, 2000; Schaub et al., 2000; Yamanaka & Ankersen, 2002; Gim & Alfriend, 2003; Breger

& How, 2007) therefore makes it a logical choice for rendezvous.

Other welcome characteristics that can be obtained through the MPC design process include robust finite time completion (Richards & How, 2003b, 2006) and guaranteed safe passive abort trajectories (Breger & How, 2006, 2008). MPC is also inherently reconfigurable. Constraints (e.g. thrust availability, passive safety requirements), and model parameters (e.g. target orbit) can be modified on-line — an attractive feature if these are not known exactly until the target has been detected.

Of course, the flexibility of MPC comes at a cost, and the main challenge of deploying MPC for spacecraft is its complexity. Numerical optimisers are comparatively complex pieces of software, at odds with the limited computing available on board a spacecraft. One of the key considerations of this work is the integration of MPC into an avionics architecture and software toolchain.

2. Mission Information

The mission scenario that will be considered in this paper is representative of one possible realisation of the Mars Sample Return mission (Beaty et al., 2008), where the "target" vehicle contains the soil sample to be captured and returned by the "chaser". MPC is applied to provide trajectory guidance and control for the whole rendezvous. During the final 100 m prior to capture, MPC will also provide small-angle attitude control to maintain the target pointing necessary for the short-range relative navigation sensor (LIDAR) to function correctly. Before this point, it is assumed that attitude control will be performed by an independent control system using reaction wheels.

2.1. Nominal Initial Conditions

The control system is designed using two nominal reference scenarios based on possible outcomes of the launch of the sample capsule. The initial conditions for each of these scenarios are expressed in terms of Keplerian orbital elements (Kaplan, 1976; Sidi, 1997; Fehse, 2003) in Table 1. These drive the design, as they force the candidate control system to handle both circular and elliptical orbits. In each case, the chaser starts approximately 300 km away from the target — a distance corresponding to the approximate operating range of the long-range RF relative navigation sensors.

2.2. Chaser Characteristics

The chaser mass, moment of inertia and maximum thrust are shown in Table 2. Note that maximum thrust guaranteed in every direction is less than the individual thrust on a single thruster due to the geometric alignment of the thrusters and the requirement that a zero net torque must be commanded for a pure translation manœuvre.

2.3. Navigation Uncertainty

To be of practical use, the MPC controller must cope with the relative navigation uncertainties shown in Table 3 (with directions aligned to a local-vertical local-horizontal (LVLH) reference frame centred on the target). Division

Scenario	Circular		Elliptical	
Vehicle	Target	Chaser	Target	Chaser
Semi-major axis, a (km)	3893	3843	4643	4593
Eccentricity, e	0	0.003	0.20441	0.20741
Inclination, i (deg)	30.0	30.3	15.0	15.3
RAAN, Ω (deg)	0	0.3	323.4	323.7
Argument of periapsis, ω (deg)	0	0.3	0	0.3
True anomaly, ν (deg)	0	-8	0	-5

Table 1: Scenario initial conditions (nominal)

Property	Value
Chaser mass (kg)	1575
Chaser moment of inertia ($kg \cdot m^2$)	$\begin{bmatrix} 1450 & -20 & 5 \\ -20 & 1800 & -5 \\ 5 & -5 & 1200 \end{bmatrix}$
Chaser thruster config	8×22 N thrusters
Chaser max thrust in single direction (N)	20

Table 2: Chaser characteristics

of error in individual directions by $\sqrt{3}$ implies the error is defined in terms of 3D Euclidean distance. Division by $\sqrt{2}$ implies that the error is defined in terms of 2D distance and a worst case error in the third co-ordinate. At short range, linear interpolation is used between minimum and maximum uncertainty levels.

Direction	V. Short	Short Min	Short Max	Long (as % of range)
	(< 100 m)	(100 m)	(5 km)	(> 5 km)
x (m)	$0.035/\sqrt{2}$	$0.035/\sqrt{2}$	$3/\sqrt{3}$	$(0.1/\sqrt{3})\%$
y (m)	$0.035/\sqrt{2}$	$0.035/\sqrt{2}$	$3/\sqrt{3}$	$(0.1/\sqrt{3})\%$
z (m)	$0.035/\sqrt{2}$	$0.035/\sqrt{2}$	$3/\sqrt{3}$	$(0.1/\sqrt{3})\%$
v_x (m/s)	0.009	0.009	$0.1/\sqrt{3}$	$(0.003/\sqrt{3})\%$
v_y (m/s)	0.007	0.007	$0.1/\sqrt{3}$	$(0.003/\sqrt{3})\%$
v_z (m/s)	0.007	0.007	$0.1/\sqrt{3}$	$(0.003/\sqrt{3})\%$

Table 3: Navigation uncertainty (3σ values)

2.4. Simulation

The MPC control system has been developed and tested using a Simplified Rendezvous Performance Simulator (SRPS) provided by ORCSAT project partners GMV. This includes a full non-linear model of the rigid body dynamics of the target and chaser in a Mars orbit, and representative thruster models. Navigation signals are simplified, and the values presented in Table 3 are used to model navigation uncertainty as zero-mean Gaussian noise. At ranges between 100 m and 5 km, the navigation uncertainty is interpolated between the values presented in Table 3.

A further level of validation has been obtained by integrating the final MPC control system developed into a higher order Functional Engineering Simulator (FES) (Le Peuvédic et al., 2008a) with more representative sensor models and authentic navigation filters.

3. System-Level Design

Following detection of the target at a range of approximately 300 km, the control system must transfer the chaser spacecraft to a "blinding point" approximately 3 m from the target, from which a passive drift trajectory must be used to complete the capture. The approach must be performed in a passively safe manner — if control authority is removed (i.e. if the thrusters are disabled due to a fault in the control or navigation systems), the resulting passive drift trajectory must remain outside of a "safety sphere" centred on the target. An additional operational requirement is that the chaser must visit a sequence of pre-specified holding points in the neighbourhood of the target and await authorisation to proceed from ground control.

In order to take greatest advantage of the ability of MPC to perform optimisation and actively handle constraints as a means to increase levels of autonomy, the MPC control system should provide both guidance and control func-

tionality. Little benefit over use of a classical feedback controller would be expected if MPC were used purely within an inner-loop to regulate to a pre-determined trajectory. Similarly, tracking a trajectory generated by MPC using a classical inner-loop controller is not an appealing solution for this application, because "bang-off-bang" actuation patterns are known to be fuel-optimal under ideal conditions (Sidi, 1997; Battin, 1999; Fehse, 2003), and there is a need to constrain transient responses as well as steady-state conditions.

Fulfilment of the objective described above using a single MPC controller would require a prediction model complex enough to perform accurate trajectory propagations over long periods of time, a prediction horizon at least long enough to predict a trajectory one orbital period into the future and a sampling frequency high enough to achieve the required accuracy prior to capture. Given that to be a useful engineering solution, the MPC controller must be evaluated in real-time at each sampling instance, it is evident that these requirements are in conflict. At the other extreme, it would be possible to have an MPC implementation for a large number of individual manœuvres. However, this would push up the complexity of development and implementation, and reduce the number of degrees of freedom with which the MPC could provide improvements.

To obtain a usable and practically realisable control system with good closed loop performance, it was decided to subdivide the rendezvous into a small number of phases. This allows the prediction model, cost function, constraint set, sampling period and prediction horizon to be chosen to suit the mission requirements.

3.1. Candidate Prediction Models

There is a significant amount of literature concerning simplified state transition matrices (STM) for propagation of the relative trajectories of two orbiting objects. The most well-known model is the Hill-Clohessy-Wiltshire (HCW) equations (Clohessy & Wiltshire, 1960), which propagate the relative position and velocity of the chaser with respect to the target in a Cartesian reference frame centred on a target object in a circular orbit. The HCW equations can be used to form a linear time-invariant (LTI) state-space model of the relative dynamics. However, the HCW equations rely on the assumption that the target orbit is near-circular (orbital eccentricity $e = 0$), and that radial and out-of-plane separations are small. As observed by Inalhan et al. (2002), the assumption of circular orbit can cause significant prediction error for orbits of eccentricity as little as $e = 0.005$. In this work, consideration of elliptical target orbits is essential, meaning that the HCW equations are not directly applicable.

As exploited by (Tschauner, 1967; Carter, 1998; Inalhan et al., 2002; Yamanaka & Ankersen, 2002), an otherwise linear model can be parameterised by the true anomaly of the target ν_{tgt}. Assuming no additional forces perturb the target orbit, a mapping can be made between ν_{tgt} and time using Kepler's equation, thus allowing construction of an LTV model (Breger, 2002). These methods predict in a Cartesian or cylindrical local reference frame and allow for elliptical orbits, but still assume small radial and out-of-plane separations. The model of Yamanaka & Ankersen (2002) has the appealing property that, unlike previous models, it is numerically identical to the HCW equations for circular orbits.

The Gim-Alfriend STM (Gim & Alfriend, 2003) offers an alternative LTV approach. It uses a linearised geometric mapping between relative curvilinear coordinates (Melton, 2000) and differences in a non-singular set of orbital elements. It is specifically designed to accurately consider the mean and osculating effects of J_2 — the effect of gravitational variation due to the oblateness of the central body of the orbit.

A further approach is to use Gauss' variational equalities (GVEs) (Schaub et al., 2000; Breger & How, 2007) — models based on Gauss' expressions relating an acceleration vector expressed in a local frame to changes in classical orbital elements. A linear time-varying model is formed by linearising differences in these elements. Relatively small changes in the orbital elements can lead to relatively large separations in Cartesian space, which is helpful in minimising the effect of linearisation error using Taylor expansions. Breger & How (2007) show how the Gim-Alfriend approach can be used with GVEs to include the effects of J_2.

3.2. Rendezvous Phase Apportionment

In order to achieve the best compromise between flexibility, performance and complexity, the phases chosen for MPC design are apportioned in a similar way to that done in the baseline control system, HARVD (Kerambrun et al., 2008). By partitioning the control problem in this way, each MPC controller can be tuned for its respective rendezvous phase (Figure 1), accounting for factors such as the relative importance of control accuracy, maximum planned trajectory duration and computational demands. The designs are also influenced by the anticipated levels of navigation uncertainty, and by higher level mission requirements.

3.2.1. Intermediate Range

The first phase, OSTG (Orbit Synchronisation Translational Guidance) uses the J_2 modified GVEs as a prediction model, to accurately capture long range dynamics. Preliminary studies found that for the present application, this model provided the lowest error between predicted open-loop trajectories, and trajectories simulated using the SRPS. Short-term control accuracy is not critical during this phase, so a relatively long sampling period can be used to obtain a relatively long prediction horizon without an excessive number of decision variables in the optimisation problem. OSTG should bring the chaser from a distance of approximately 300 km (of which the radial and out-of-plane components will be significant, so would lead to high levels of linearisation error for models based upon Cartesian or cylindrical co-ordinates) into an orbit with the same Keplerian orbital elements (e.g. Kaplan (1976); Sidi (1997); Fehse (2003)) as the target, with the exception of the true anomaly, ν_{chs}, which should attain a value corresponding to an in-track separation of between 10 km and 30 km. Unlike in HARVD, in-plane and out-of-plane corrections are considered simultaneously.

3.2.2. Short Range

The second phase, INTG (Impulsive Nominal Translational Guidance) starts once the OSTG phase has been completed. At this point, out-of-plane and radial separations are smaller, but there is still a significant in-track separation. The remit of the INTG controller is to reduce this separation by performing passively-safe impulsive manœuvres

7

between a series of prescribed holding points up until a target approach point (TAP) approximately 100 m from the target. At each holding point, the chaser must wait for a signal from ground control indicating permission to proceed. If the INTG phase commences at a point closer than the furthest-out holding point, the first holding point to be scheduled should be the nearest one in the direction of the target.

At short range, short-term accuracy is more important than accuracy of long-term trends. The linearised geometric transformation $\Sigma(\mathbf{e}_{\text{tgt}})$ that translates from orbital element separations to distances is only an approximation. It is also important for the MPC controller to function in an elliptical orbit. In order to accommodate more accurate passive safety constraints and accurate attainment of holding points, a shorter sampling period is required. To accommodate these requirements, the prediction model is provided by the Yamanaka-Ankersen equations (Yamanaka & Ankersen, 2002). This is a simpler model than the J_2-modified GVEs, and linearisation error is not significant once the out-of-plane and radial separations have been mostly corrected.

3.2.3. Very Short Range

The final phase, FTTG (Forced Terminal Translational Guidance) uses the Yamanaka-Ankersen equations and a linearised quaternion-based attitude model to allow high precision control up until the final capture. Because at this range the primary relative navigation sensor is LIDAR, it is critical that an attitude orientation is maintained so that the target is within the LIDAR's field of view. This can be achieved by regulating to a setpoint which maintains the target in the centre of the LIDAR field of view. Similarly, the closed-loop trajectory must reach a point so that a capture error of less than 20 cm is achieved after a subsequent period of free drift. It is therefore clear that during this phase a far shorter sampling period is required in order to facilitate the necessary level of disturbance rejection.

3.2.4. Collision Avoidance Manœuvre (CAM)

A fourth phase (CAM) for active collision avoidance in case of contingency is included in the design. If a fault occurs during the FTTG phase, or if it is detected that the velocity or position close to the "blinding point" puts the chaser on a collision-course with the target, an active avoidance manœuvre must be performed to move the chaser away from the target. In the case of the scenario considered in this work, the chaser must be moved to a distance at least 500 m from the target in at most 3 orbits. At this point, the INTG mode can resume operation.

Being a contingency manœuvre, the CAM might need to operate with reduced navigation performance. If target-pointing is lost, the navigation estimate will degrade over time. Furthermore, to successfully avoid collision, the CAM must calculate and command a control action in a short period of time from the instant it is triggered. Therefore, for speed, and because the CAM operates at relative close proximity to the target, the Yamanaka-Ankersen equations are preferred.

3.3. System Architecture

The MPC subsystems are implemented separately, each in an individual Simulink subsystem, with a set of common input and output signals. Whilst each of these contains the core MPC controller, each subsystem also contains
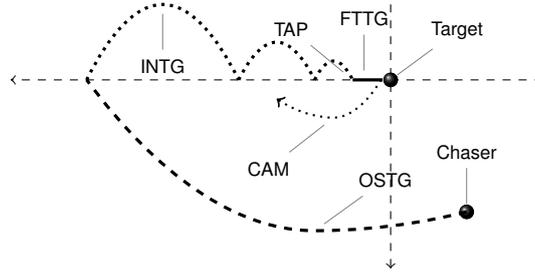
Figure 1: Rendezvous phases (not to scale)

any support functions required to detect whether the subsystem should be active or not, perform co-ordinate transformations, detect completion of the mission phase and raise a "success" flag, and, in the case of the INTG mode, choose which holding point should be achieved next.

Each of the MPC controller subsystems is responsible for raising a "success" flag when its terminal conditions have been achieved. An external scheduling algorithm then enables the MPC corresponding to the next mission phase. The schedule of phases is presented in Figure 2. It should be noted that the CAM is only applicable during the final phase of the rendezvous. In the OSTG and INTG phases the trajectories are naturally passively safe so a "do nothing" approach is sufficient.
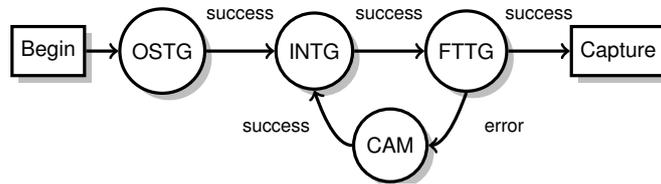

Figure 2: MPC Controller Scheduling

A separate output preparation block, common to all modes and operating at a higher sampling frequency, is also implemented to convert impulsive velocity change commands generated by the MPC subsystems to finite duration acceleration pulses and to rotate these pulses from the co-ordinate system preferred by the active MPC subsystem to an inertial reference frame as determined by the interface with the simulator.

3.4. Optimisation Formulation

One of the key requirements of the work presented here is that the final design must be able to run in real-time on hardware representative of what could be included on board a future spacecraft. As part of the high-level system design the MPC for all phases was restricted to linear time-varying dynamics models based upon convex linear programming (LP) or quadratic programming (QP) with linear constraints. LTI models are unsuitable as they cannot capture relative dynamics in elliptical orbits — of the available dynamics models, only HCW is LTI and that is only applicable to the circular case. At the other extreme, nonlinear MPC would permit best fidelity, but this was judged not to be worth

9

the additional complexity in the optimiser. Similarly, solver software for mixed-integer optimisation (Bemporad & Morari, 1999) was judged too complex for this application, despite the potential to handle the non-convex constraints.

Explicit MPC (Bemporad et al., 2002b,a), in which the optimisation is pre-solved offline as a parametric function of the state was ruled out due to the complex nature of the control law. Explicit MPC can be appealing, because the control law is typically a piecewise affine function of the state, which is readily implemented. However, with the LTV dynamics, many complex constraints, six or more states and a large region of operation, that control law would become prohibitively large in terms of the number of partitions.

The restriction to only use convex LP or QP means that non-convex constraints such as collision avoidance must be approximated in a convex manner. However, there is no requirement that only a single QP or LP may be solved at each sampling instance. For example, in order to include a measure of time in the cost function for the OSTG and INTG phases a variable horizon is used. This can be implemented using *switched controllers*. At each time step each element of a set of convex, finite horizon MPC controllers is evaluated, each with a different prediction horizon up to a maximum horizon N_{max}. The control action from the controller with the lowest overall cost (calculated as a weighted sum of the prediction horizon and the optimum cost from the finite horizon MPC controller) is chosen and applied to the plant.

4. Intermediate Range (OSTG)

The objective of the OSTG MPC controller is to synchronise the orbit of the chaser with that of the target in finite time, whilst minimising propellant usage. The chaser should be brought onto the target orbit, with a difference in true anomaly Δv such that the "in-track" separation is between 10 km and 30 km. This may be on either side of the target.

4.1. Prediction Model

For the intermediate range MPC controller (OSTG), a prediction model based on linearised J_2-modified Gauss's Variation Equations (GVEs) (Breger & How, 2007) is used. Non-singular orbital elements are used rather than the basic Keplerian orbital elements, to avoid ambiguities between ω and v when $e = 0$, and to avoid numerical ill condition when calculating the J_2 modification (Gim & Alfriend, 2003). Denoting the vector of orbital elements of chaser and target using subscripts "tgt" and "chs" as

$$\mathbf{e}_{chs} = \begin{bmatrix} a_{chs} & e_{chs} & i_{chs} & \Omega_{chs} & \omega_{chs} & v_{chs} \end{bmatrix}^T \tag{1}$$

and

$$\mathbf{e}_{tgt} = \begin{bmatrix} a_{tgt} & e_{tgt} & i_{tgt} & \Omega_{tgt} & \omega_{tgt} & v_{tgt} \end{bmatrix}^T \tag{2}$$

10

respectively, the relative non-singular orbital elements (Gim & Alfriend, 2003) are expressed as:

$$
\Delta \bar{\mathbf{e}} = \begin{bmatrix} \Delta a \\ \Delta \theta \\ \Delta i \\ \Delta q_1 \\ \Delta q_2 \\ \Delta \Omega \end{bmatrix} = \begin{bmatrix} a_{\text{chs}} - a_{\text{tgt}} \\ (\nu_{\text{chs}} + \omega_{\text{chs}}) - (\nu_{\text{tgt}} + \omega_{\text{tgt}}) \\ i_{\text{chs}} - i_{\text{tgt}} \\ e_{\text{chs}} \cos \omega_{\text{chs}} - e_{\text{tgt}} \cos \omega_{\text{tgt}} \\ e_{\text{chs}} \sin \omega_{\text{chs}} - e_{\text{tgt}} \sin \omega_{\text{tgt}} \\ \Omega_{\text{chs}} - \Omega_{\text{tgt}} \end{bmatrix}.
\tag{3}
$$

The state vector during the OSTG phase is

$$
x(j) = \Delta \bar{\mathbf{e}}(j).
\tag{4}
$$

An impulsive input discretisation is used, with the plant inputs being expressed as impulsive velocity changes expresssed in the chaser orbital frame (COF). The chaser orbital frame is an LVLH (local vertical, local horizontal) reference frame centred on the centre of mass of the chaser. The COF axes are defined as z_{cof} pointing towards the focus of the orbit, y_{cof} parallel to the chaser orbital angular velocity, and x_{cof} completing the right-hand set.

4.2. Variable Horizon Cost Function

It has been demonstrated in the literature (e.g. Tillerson et al. (2002)) that a 1-norm cost function offers better conservation of fuel than a quadratic or sum-of-squares cost function. Therefore, in order to minimise absolute propellant consumption whilst reaching the desired terminal conditions in finite time, a 1-norm cost function on the ΔV applied in the chaser orbital frame is used, along with a weighted variable horizon (Richards & How, 2006, 2003a). In accordance with the use of variable horizon MPC, the control objectives are encoded in the terminal constraints (which will be specified later). The chosen cost function to be minimised is

$$
J(\mathbf{u}, \alpha, N) = N + w_u \sum_{k=1}^{N-1} \|u(t + kT_s | t)\|_1
$$

where $u(t + kT_s|t)$ is interpreted as the predicted input at time $t + kT_s$ given the measurements at time t, w_u is a scalar tuning parameter weighting the propellant consumption against manœuvre time, $N \leq N_{\text{max}}$ is the prediction horizon (in sampling instances), and

$$
\mathbf{u} \triangleq \begin{bmatrix} u(t + T_s|t)^T & \dots & u(t + (N-1)T_s|t)^T \end{bmatrix}^T.
\tag{5}
$$

Variable α represents the angle of a passive safety constraint. Section 4.4 describes how multiple values are attempted via switched MPC laws. It should be observed from this structure that the first element of the predicted input sequence is $u(t + T_s|t)$, not $u(t|t)$, because the prediction model includes a unit delay to account for finite computation time. The

LTV prediction model used conforms to the following structure:

$$x(t + T_s|t) = A(t)x(t|t) + B(t)u(t|t - T_s) \quad \text{(from prev. time step)}$$

$$x(t + 2T_s|t) = A(t + T_s)x(t + T_s|t) + B(t + T_s)u(t + T_s|t)$$

$$x(t + 3T_s|t) = A(t + 2T_s)x(t + 2T_s|t) + B(t + 2T_s)u(t + 2T_s|t)$$

$$\vdots$$

$$x(t + NT_s|t) = A(t + (N - 1)T_s)x(t + (N - 1)T_s|t) \tag{6}$$

$$+ B(t + (N - 1)T_s)u(t + (N - 1)T_s|t).$$

4.3. Input Constraints

Constraints on the maximum deliverable ΔV are placed on each element of the input vector

$$-u_{\text{lim}} \leq u_i(j|t) \leq u_{\text{lim}}, \quad i \in \{1, 2, 3\},$$

where u_{lim} is defined by the force capacity F_c (in Newtons), mass m (in kilograms), sampling period T_s (in seconds) and duty cycle D_τ (as a fraction of the sampling period):

$$u_{\text{lim}} = D_\tau \frac{F_c}{m} T_s.$$

By setting $D_\tau \leq 1/\sqrt{2}$, this imposed ∞-norm constraint can be used to conservatively guarantee satisfaction of a 2-norm constraint on maximum deliverable thrust in any direction.

4.4. Safety Constraints

The orbit synchronisation must be performed in a passively safe manner. The free drift trajectories following any ΔV command by the MPC must remain outside of a *safety sphere* centred on the target for at least one orbital period. This safety sphere is specified to vary in size as a function of the current separation between target and chaser. This, again is clearly a non-convex constraint. The non-convex safety sphere avoidance constraint can be approximated by using a rotating half-space constraint (Figure 3).

The MPC problem is initially solved with output constraints ensuring that the free-drift trajectories remain inside the half-space denoted $\mathcal{R}$ in Figure 3. This is done by including explicit open-loop predictions of the free drift trajectories at each time step, as originally proposed by (Breger & How, 2008, 2006). Letting $j = t + kT_s$:

$$x^{\text{free}}(j|j|t) = x(j|t)$$

$$x^{\text{free}}(j + T_{\text{safe}}|j|t) = A(j; j + T_{\text{safe}})x(j|t)$$

$$x^{\text{free}}(j + 2T_{\text{safe}}|j|t) = A(j; j + 2T_{\text{safe}})x(j|t)$$

$$\vdots$$

$$x^{\text{free}}(j + N_{\text{safe}}T_{\text{safe}}|j|t) = A(j; j + N_{\text{orb}}T_{\text{safe}})x(j|t). \tag{7}$$
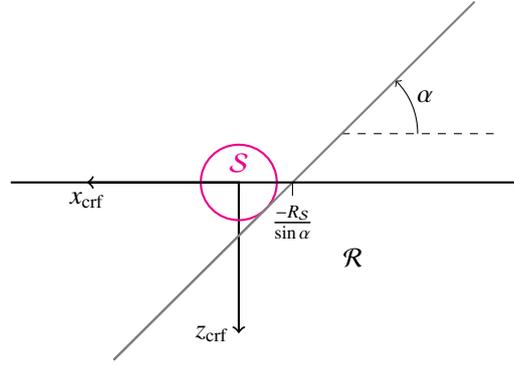
Figure 3: Rotating safety sphere constraint

The notation $A(j; m)$ denotes the STM providing an open-loop propagation from time j to time m, and the notation $x^{\text{free}}(m|j|t)$ indicates the free drift prediction at time m, propagated from the open-loop controlled prediction at time j, based upon the navigation data from time t. Define $\mathbf{e}_{\text{tgt}}(j)$ to be the vector of target orbital elements at time j, and $\Sigma\left(\mathbf{e}_{\text{tgt}}(j)\right)$ as a linearised geometric transformation matrix converting the relative orbital elements to relative positions in the cylindrical reference frame as defined in (Gim & Alfriend, 2003). The passive safety constraints are then expressed as

$$\begin{bmatrix} \cos\alpha(j|t) & 0 & \sin\alpha(j|t) & 0 & 0 & 0 \end{bmatrix} \times$$

$$\Sigma(\mathbf{e}_{\text{tgt}}(j + iT_{\text{safe}}))x^{\text{free}}(j + iT_{\text{safe}}|j|t) \leq -R_{\mathcal{S}}(t)$$

$$\forall i, \ j \in (\{0, \ldots, N_{\text{safe}}\} \otimes \{t + T_s, \ldots, t + (N-1)T_s\}) \quad (8)$$

where N_{safe} is the number of points at which the free-drift trajectory is constrained, and T_{safe} is the time period between these predicted free-drift trajectory samples. Given an angle parameter $\alpha(t)$ passed to the optimiser, the angles for each time step are given by

$$\alpha(j|t) = \begin{cases} \alpha(t) + k/N \times 45° & \text{if } \mod(\alpha(t), 180°) = 0 \\ \alpha(t) & \text{otherwise} \end{cases} \quad (9)$$

noting that the angle is held constant unless the chaser starts close to the x_{crf} axis. In the latter case, a fixed constraint can lead to infeasibility. Finally the whole optimisation is solved for a range of choices of $\alpha(t)$

$$J^*(N) = \min_{\alpha(t)} \min_{\mathbf{u}, \mathbf{N}} J(\mathbf{u}, \alpha, N) \quad (10)$$

subject to

$$\alpha(t) \in \{\alpha_0(t) - 45°, \ \alpha_0(t), \ \alpha_0(t) + 45°\}. \quad (11)$$

The value $\alpha_0(t)$ is calculated as the angle between the chaser and the z_{crf} axis at time t, rounded to the nearest integer multiple of $45°$. The rounding improves performance by preventing large numbers of small changes in constraints from one optimisation to the next, which would lead to many small corrections in trajectory and wasted fuel.

13

By determining the angle of the half-space constraint, the minimising angle $\alpha(t)$ also determines which side of the target the terminal constraint must be placed.

4.5. Terminal Constraint

The control objective is to bring all relative non-singular orbital elements apart from $\Delta\theta$ to zero. The variable $\Delta\theta$ should instead be brought into one of two boxes. This is a non-convex constraint (Figure 4). The side from which the final approach is performed is determined by $\alpha(t)$ as described above.
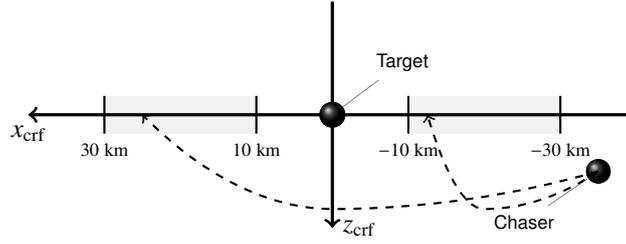

Figure 4: Choice of OSTG terminal set (not to scale)

Now, letting $j_N = t + NT_s$, the terminal constraint is a combination of

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} x(j_N|t) =
\begin{bmatrix}
0 \\ 0 \\ 0 \\ 0 \\ 0
\end{bmatrix}
\tag{12}
$$

(implemented as symmetric inequality constraints for compatibility with the available solver) and one of

$$
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0
\end{bmatrix} \Sigma\left(\mathbf{e}_{\text{tgt}}(j_N)\right) x(j_N|t) \leq
\begin{bmatrix}
30 \times 10^3 \\
-10 \times 10^3
\end{bmatrix}
\tag{13}
$$

or

$$
\begin{bmatrix}
0 & -1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0
\end{bmatrix} \Sigma\left(\mathbf{e}_{\text{tgt}}(j_N)\right) x(j_N|t) \leq
\begin{bmatrix}
30 \times 10^3 \\
-10 \times 10^3
\end{bmatrix}
\tag{14}
$$

depending on the side from which the approach has been determined to occur, based on the angle $\alpha(t)$.

4.6. Tuning

The maximum prediction horizon during the OSTG phase must be long enough for the predicted open-loop controlled trajectory to be feasible with respect to the terminal constraint and constraints imposed at other time steps during the prediction horizon. The sampling period T_s and maximum prediction horizon $N_{\max}$ must be chosen to accommodate this, even if due to the cost function weightings the closed loop trajectory takes longer. In addition, T_s

must be short enough to allow sufficient flexibility of control, but long enough that corrections due to navigation error does not cause excessive propellant consumption. By performing batch simulations over a selection of different prediction horizons, sampling times, and input weightings it was found that $N_{\max} = 25$ and $T_s = 600$ s, with $w_u = 10$ was a suitable compromise, leading to a total prediction horizon of 15000 s or approximately 1.5 orbits for the elliptical scenario and 2 orbits for the circular scenario.

5. Short Range (INTG)

Once the OSTG MPC submodule has brought the chaser craft into the same orbit as the target, the separation between them is still in the range of 10 km–30 km (Figure 4) due to a difference in true anomaly between the two vehicles. The INTG MPC submodule has the task of reducing this by performing a series of (passively safe) impulsive transfers between a sequence of pre-specified holding points up until a point 100 m from the target.

Minimisation of propellant consumption is still critical, so variable horizon MPC with a 1-norm cost function is considered for this phase also. Because it is unknown *a priori* when the flag permitting exit from a holding point will be raised, there is no point designing an MPC that predicts past the next holding point in the sequence. Whilst there is not a strict prescription on the shape of the trajectories between holding points, there is an expectation that in the absence of uncertainties these trajectories should resemble the conventional "bang-off-bang" hopping trajectories (Fehse, 2003), each lasting approximately half an orbit. For a given prediction horizon N, this allows a shorter sampling period T_s to be used than for the OSTG phase whilst maintaining an optimisation problem of similar size.

5.1. Holding Point Schedule

Holding points are periodic free-drift trajectories centred on a point a fixed distance away from the target. Because it is unknown how long the chaser will be required to stay at any particular hold point, the MPC controller is designed with the goal of reaching the "next" hold-point, with the position of the next hold-point provided to the MPC controller as an input parameter. It is not guaranteed that the OSTG MPC controller will deliver the chaser to exactly the first holding-point (in fact, from the perspective of fuel consumption and completion time, it is preferable for the chaser to be delivered into the same orbit as the target with a reduced in-track separation). In such cases the first hold-point is chosen as the closest hold-point in the direction of the target from the chaser at the instant that the INTG MPC controller is enabled (Figure 5). Let x_{hp} denote the distance (in kilometres) from the target to the centre of the currently scheduled holding point.

5.2. Prediction Model

For the short range (INTG) phase the Yamanaka-Ankersen (Yamanaka & Ankersen, 2002) STM (state transition matrix) is used. This directly propagates the relative dynamics between chaser and target in terms of relative positions
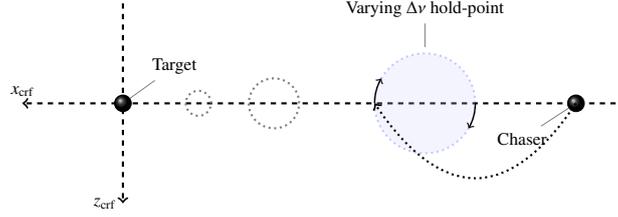
Figure 5: Choosing the first hold point (example in elliptical orbit)

and velocities in a cylindrical reference frame (CRF) and is a generalisation of the classical Hill-Clohessy-Wiltshire STM (Kaplan, 1976; Sidi, 1997; Fehse, 2003) to elliptical orbits. The state vector is

$$x = \begin{bmatrix} x_{\mathrm{crf}} & y_{\mathrm{crf}} & z_{\mathrm{crf}} & \dot{x}_{\mathrm{crf}} & \dot{y}_{\mathrm{crf}} & \dot{z}_{\mathrm{crf}} \end{bmatrix}^T. \tag{15}$$

The cylindrical co-ordinate frame (CRF) is a reference frame with its origin centred on the target (Figure 6). Value z_{crf} is the relative position in the radial direction (positive values towards the focus of the orbit), y_{crf} is the relative position in the direction normal to the plane of the orbit, and x_{crf} is in the direction that completes the right-hand set. The direction labelled $\overline{V}$ in Figure 6 corresponds to the instantaneous direction of the velocity of the target. It should be noted that this direction is time-varying with respect to the CRF for elliptical orbits, and aligned with x_{crf} for circular orbits. The cylindrical reference frame is used in preference to a Cartesian target orbital frame (TOF) because
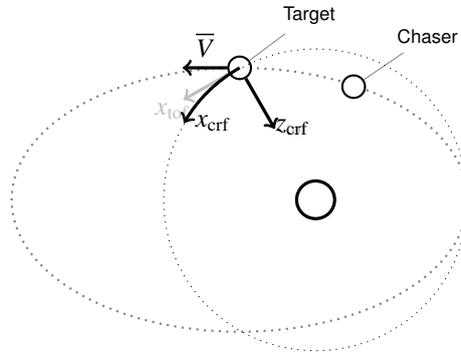

Figure 6: Cylindrical reference frame (CRF)

it allows for relatively large in-track separations to be possible whilst keeping the linearisation error of a model of relative dynamics model small. Linearisations from radial and out-of-plane separations are larger (Kaplan, 1976; Sidi, 1997; Melton, 2000), but nevertheless not significant over the ranges considered during the INTG phase.

The Yamanaka-Ankersen STM is linear parameter varying (LPV) in terms of the true anomaly of the target ν_{tgt}. However, because in this scenario the target is passive, under the assumption of perfect conditions, the target mean anomaly can be propagated as a function of time. The propagated true anomaly can be recovered from the propagated mean anomaly using the trigonometric expansion of *L'Equation du Centre* (Tisserand, 1889), or by solving

16

Kepler's equation iteratively using Newton's method (Sidi, 1997). This allows the Yamanaka-Ankersen STM to be implemented as an LTV model.

For the INTG phase, an impulsive discretisation is used to best represent the nature of the input trajectory expected, and scaling is performed so that x_{crf}, y_{crf} and z_{crf} are measured in kilometres, the velocities in the CRF are expressed in metres per second, and that the "input" is an impulsive ΔV is expressed in units of 0.1 ms^{-1}. The scaling is to improve numerical stability by ensuring that all decision variables and constraints are of similar magnitude. If $\Phi(t)$ is the unmodified Yamanaka-Ankersen STM with states expressed in m and ms^{-1} then matrix $A(t)$ is calculated as

$$
A(t) = \mathrm{diag}\left(\begin{bmatrix} 10^{-3} \\ 10^{-3} \\ 10^{-3} \\ 1 \\ 1 \\ 1 \end{bmatrix}\right) \Phi(t) \mathrm{diag}\left(\begin{bmatrix} 10^{3} \\ 10^{3} \\ 10^{3} \\ 1 \\ 1 \\ 1 \end{bmatrix}\right)
\tag{16}
$$

and $B(t)$ is calculated as

$$
B(t) = A(t)\begin{bmatrix} \mathbf{0}_{3\times3} \\ 0.1 I_{3\times3} \end{bmatrix}.
\tag{17}
$$

5.3. Variable Horizon Cost Function

Like for OSTG, in order to ensure finite time completion whilst minimising propellant consumption, a 1-norm cost function is used in conjunction with a variable horizon, however unlike OSTG, INTG does not apply a weighting on planned time to finish *i.e.* the horizon N. Instead, a weighting on distance is employed, as it provides a more consistent response over the range of distances that INTG has to handle. A fixed tuning of time weight would provide overly aggressive control at short distances. Letting $j = t + kT_s$, the cost function to be minimised is chosen to be:

$$
J = \sum_{k=1}^{N-1} \left(\| E_c \left(x_{\mathrm{crf}}(j|t) - r(j) \right) \|_1 + w_u \| u(j|t) \|_1 \right).
\tag{18}
$$

where w_u is the weighting on input (ΔV) relative to distance, and E_c is a weighting matrix selecting the out-of-plane and in-track components of the state vector:

$$
E_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.
\tag{19}
$$

The reference $r(j)$ is chosen as

$$
r(j) = \begin{cases} -x_{hp}\left(1 + e\cos v(j)\right) & \text{if approach from } x < 0 \\ x_{hp}\left(1 + e\cos v(j)\right) & \text{if approach from } x > 0. \end{cases}
\tag{20}
$$

The reference trajectory traces the x_{crf} position of a *varying-Δv* holding point at distance x_{hp} on the appropriate side of the target. The summation from $k = 1$ rather than $k = 0$ is, as for the OSTG mode, because the prediction mode in the optimisation problem assumes the structure (6) with a modelled unit delay.

17

5.4. Terminal Constraints

The terminal constraint can be split into five components. The constraint on the immediate position in the terminal constraint is given as:

$$\begin{bmatrix} I & 0 \\ -I & 0 \end{bmatrix} x(j_N) \leq c_{hp}(j_N). \tag{21}$$

This constrains the position of the chaser at time $j_N = t + NT_s$ to be inside a box, centred on a point x_{hp} away from the target on $\overline{V}$, of width $2x_{hp}e^+$ in the orbital plane and $0.6x_{hp}e^+$ in the out-of-plane direction (Figure 7), where $e^+ = e_{\text{tgt}} + 0.1$ to ensure that the box is non-empty for a circular orbit. The rationale for this constraint is that it is sufficient to contain a *varying* Δv periodic trajectory, whilst limiting out-of-plane motion without driving it completely to zero. Similar constraints can be given for the position after 1/4, 1/2 and 3/4 orbits. The constraints on the position
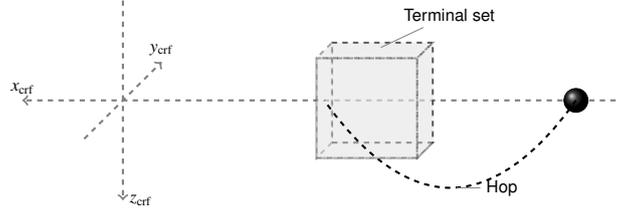

Figure 7: INTG terminal set (> 5 km)

1/4, 1/2 and 3/4 orbits after $t + NT_s$ are given as:

$$\begin{bmatrix} I & 0 \\ -I & 0 \end{bmatrix} A(j_N; j_N + \Delta T)x(j_N) \leq c_{hp}(j_N + \Delta T),$$

$$\forall \Delta T \in \{T_{1/4}, T_{1/2}, T_{3/4}\}. \tag{22}$$

Finally, in order that the holding point can be periodic, an invariance condition must be imposed. Due to the relative dynamic model only having one non-periodic mode, it is sufficient to only impose this constraint on the x_{crf} state:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} (A(j_N; j_N + T_{\text{orb}}) - I) \, x(j_N) = 0. \tag{23}$$

The MPC controller for INTG is used to provide combined guidance and control. Because a variable horizon is being used, the control objective must be encoded in the choice of terminal set. As described previously, the objective of the INTG MPC controller is to impulsively transfer the chaser from one holding point to the next. When the holding point is at a distance further than 5 km, the INTG MPC controller is designed with a terminal constraint that the chaser be on a periodic trajectory that is inside a box centred on the hold-point position on V-bar in the cylindrical reference frame after free-drift periods of 1/4, 1/2 and 3/4 orbits in terms of true anomaly (with corresponding times denoted as $T_{1/4}$, $T_{1/2}$, $T_{3/4}$). To calculate these periods, the true anomaly is propagated from its predicted value at the end of the

prediction horizon for 1/4, 1/2 and 3/4 orbits.

$$\nu(j_N + T_{1/4}) = \nu(j_N) + \pi/2$$

$$\nu(j_N + T_{1/2}) = \nu(j_N) + \pi/4$$

$$\nu(j_N + T_{3/4}) = \nu(j_N) + 3\pi/2$$

The mean anomaly $M(t)$ is obtained from the true anomaly $\nu(t)$ using the equation (Sidi, 1997):

$$M(t) = 2\tan^{-1}\left(\frac{\sqrt{1-e}\tan(\nu/2)}{e\sqrt{1-e^2} - \frac{\sqrt{1+e}\sin\nu}{1+e\cos\nu}}\right). \tag{24}$$

Noting that $M(t) = n\Delta t$, where Δt is defined to be the time since the target last passed the periapsis of its orbit, and n is the mean anomaly rate (Sidi, 1997), the times for 1/4, 1/2 and 3/4 orbits can be calculated as:

$$T_{1/4} = ((\Delta t(j_N + T_{1/4}) - \Delta t(j_N)) \mod T_{\text{orb}}) \tag{25a}$$

$$T_{1/2} = ((\Delta t(j_N + T_{1/2}) - \Delta t(j_N)) \mod T_{\text{orb}}) \tag{25b}$$

$$T_{3/4} = ((\Delta t(j_N + T_{3/4}) - \Delta t(j_N)) \mod T_{\text{orb}}). \tag{25c}$$

Remembering that x_{hp} is the absolute holding point distance from the target (in kilometers), and $e^+ = e_{\text{tgt}} + 0.1$, then define vector

$$c_{hp}(t) = \underbrace{\begin{bmatrix} -x_{hp}(1-e^+) \\ 0.3x_{hp}e^+ \\ x_{hp}e^+ \\ x_{hp}(1+e^+) \\ 0.3x_{hp}e^+ \\ x_{hp}e^+ \end{bmatrix}}_{x_{\text{crf}}(t)<0} \text{ or } \underbrace{\begin{bmatrix} x_{hp}(1+e^+) \\ 0.3x_{hp}e^+ \\ x_{hp}e^+ \\ -x_{hp}(1-e^+) \\ 0.3x_{hp}e^+ \\ x_{hp}e^+ \end{bmatrix}}_{x_{\text{crf}}(t)>0} \tag{26}$$

which is used in (22).

5.5. Terminal Constraints (Hopping < 5 km)

If the next hold point is equal to, or closer than 5 km, the terminal constraint is stricter. Instead of requiring the chaser to be on an arbitrary periodic trajectory that remains within a fixed box in the cylindrical reference frame, the terminal constraint requires that the chaser enters a trajectory in proximity to a *varying $\Delta\nu$* holding trajectory. This is so that at the final holding point (the TAP), there is little overshoot when switching to the FTTG mode which regulates to $\bar{V}$. Constraints (21) to (22) are modified by changing $c_{hp}(t)$ to be:

$$c_{hp}(t) = \begin{bmatrix} x_{href}(t) + |\delta_{x_{href}}| \\ 0.3x_{hp}e^+z_{href}(t) + |\delta_{z_{href}}| \\ -x_{href}(t) + |\delta_{x_{href}}| \\ 0.3x_{hp}e^+ - z_{href}(t) + |\delta_{z_{href}}| \end{bmatrix} \tag{27}$$

19

where

$$x_{href}(t) = \begin{cases} -x_{hp}(1 + e \cos v(t)) & \text{if } x_{\text{crf}} < 0 \\ x_{hp}(1 + e \cos v(t)) & \text{if } x_{\text{crf}} \geq 0 \end{cases} \tag{28a}$$

$$z_{href}(t) = \begin{cases} -x_{hp}e \cos v(t) & \text{if } x_{\text{crf}} < 0 \\ x_{hp}e \cos v(t) & \text{if } x_{\text{crf}} \geq 0 \end{cases} \tag{28b}$$

and $\delta_{x_{href}}$ and $\delta_{z_{href}}$ are tolerances based upon an upper bound on the largest propagation of the expected worst-case sensor uncertainty through the prediction model

$$\begin{bmatrix} \delta_{x_{href}} \\ \delta_{y_{href}} \\ \delta_{z_{href}} \end{bmatrix} = A^+(j; j + T_s)w_{\max}$$

where $w_{\max}$ is a vector consisting of the element-wise maximum absolute navigation errors and $A^+(t)$ is a matrix whose elements are the absolute values of the elements of $A(t)$, defined as:

$$A^+ \triangleq \left\{ A^+ \in \mathbb{R}^{6 \times 6} \,|\, A_{ij}^+ = |A_{ij}|, \; \forall i \in \{1, \ldots, 6\}, \; j \in \{1, \ldots, 6\} \right\}. \tag{29}$$
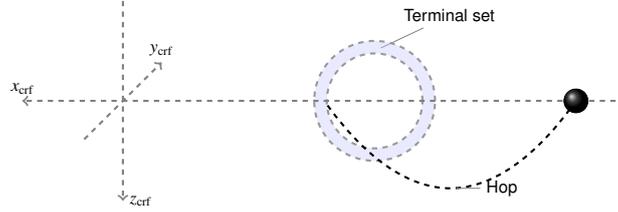

Figure 8: INTG terminal set (< 5 km)

5.6. Input Constraints

Nominal input constraints are imposed in exactly the same way as for OSTG, except they are imposed with respect to the cylindrical reference frame (CRF) rather than the chaser orbital frame (COF).

5.7. Short-Term Passive Safety

Passive safety is considered in a similar manner as OSTG, with free drift propagations made for a whole orbit from each open-loop controlled prediction (7). Because the Yamanaka-Ankersen equations directly propagate the relative position dynamics in the CRF, the geometric transformation Σ is not required, and the safety constraint is:

$$\begin{bmatrix} -\text{sgn}(x_{\text{crf}}(t)) & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x^{\text{free}}(j + iT_{\text{safe}}|j|t) \leq -R_{\mathcal{S}}(t)$$

$$\forall i, \, j \in (\{0, \ldots, N_{\text{safe}}\} \otimes \{t + T_s, \ldots, t + (N - 1)T_s\}). \tag{30}$$

20

Unlike during the OSTG phase, the angle α of the halfspace approximation to the safety sphere may only take on values 0 and π. This is because during the INTG phase the expected trajectory should never pass over or underneath the target because in a capture scenario, the final approach can be performed from either direction.

5.8. Long-Term Passive Safety

In addition to ensuring that no passive drift trajectory intersects the safety sphere during the period of one orbit, an extra requirement is imposed at ranges < 5 km to ensure that the passive drift trajectory does not intersect the safety sphere within 10 orbits. Rather than explicitly predicting the full 10 orbits, a drift condition is imposed to ensure that after a single orbit, the chaser is at least as far away from the target as it is currently. This is expressed as

$$\begin{bmatrix} -\text{sgn}(x_{\text{crf}}(t)) & 0 & 0 & 0 & 0 & 0 \end{bmatrix} (A(j; j + T_{\text{orb}}) - I) x(j) \leq 0 \tag{31}$$

where in this case, $j = t + kT_s \quad \forall k \in \{1, \ldots, N-1\}$. By doing this, a free drift trajectory will always (on average) move away from the target. Thus, as long as no collision occurs during the first complete orbit of passive drift (enforced by the short-term passive safety constraints), no collision will occur during subsequent orbits.

5.9. Constraint Tightening

In order to account for navigation uncertainty, each of the running constraints are subsequently tightened using the nilpotent disturbance feedback policy proposed in (Richards & How, 2003b,a), adapted to the LTV system dynamics. It is assumed that most uncertainty comes from navigation error, therefore, the worst-case uncertainty is assumed to be within the 3σ values shown in Table 3 when closer than 5 km. To avoid infeasibility, at greater than 5 km the constraint tightening is relaxed to consider only 1σ noise values. Qualitatively, the effect of this is to make the passive safety constraints more conservative, and the drift condition equal to a predicted passive drift away from the target rather than merely not drifting towards the target. To avoid infeasibility, the terminal constraint is not tightened.

5.10. Tuning

The maximum prediction horizon during the INTG phase must be sufficient to feasibly encompass a complete transfer from one holding point to the next. In a circular orbit, a single impulsive radial or tangential "hopping" manœuvre (Fehse, 2003) takes exactly half an orbit. In an elliptical orbit there is a small amount of variability dependent on the target true anomaly. As for the OSTG phase, T_s should be short enough to allow impulsive thruster burns at the optimum times, but long enough that excessive decision variables are not required in the optimisation, and that propellant consumption is not dominated by corrective manœuvres caused by navigation uncertainty. By performing batch simulations over a selection of T_s and w_u, it was found that a setting of $T_s = 300$ s was suitable, alongside an input weighting $w_u = 0.2$. With $N_{\text{max}} = 20$, this corresponds to 6000 s, or 0.8 orbits for the circular scenario, or 0.6 elliptical scenario — both sufficient for the "optimal" open-loop manœuvre to be feasible.

6. Terminal to Capture (FTTG)

The objective of the final translation phase (FTTG) is to control the chaser along a straight-line trajectory a few centimetres from $\overline{V}$ towards the target. The FTTG phase ends at a point, approximately 3 m from the target, where the LIDAR view is blocked by parts of the chaser spacecraft itself. At this point, all control authority is relinquished, and the final capture occurs during a passive drift trajectory. If, due to a fault, or higher than expected navigation error the chaser arrives in the neighbourhood of the target at a higher velocity than expected, or on a trajectory that cannot be corrected in time, a collision avoidance manœuvre (CAM) must be commanded. During the FTTG phase, attitude control is performed using the thrusters rather than reaction wheels, and is included in the remit of the MPC controller. The MPC must maintain LIDAR target pointing in order to preserve relative navigation.

6.1. Thruster-Aligned Prediction Model

During the FTTG phase, attitude control is performed using thrusters rather than reaction wheels, and becomes part of the responsibility of the MPC controller. Due to the thruster geometry, there is cross-coupling between the inputs to the translational dynamics and the attitude dynamics (each thruster input provides a non-zero torque in combination with a force). This can be represented mathematically as

$$
\begin{bmatrix} \delta x_{\text{tr}}(j + T_s) \\ \delta x_{\text{at}}(j + T_s) \end{bmatrix} = \begin{bmatrix} A_{\text{tr}}(j) & \mathbf{0} \\ \mathbf{0} & A_{\text{at}}(j) \end{bmatrix} \begin{bmatrix} \delta x_{\text{tr}}(j) \\ \delta x_{\text{at}}(j) \end{bmatrix}
$$

$$
+ \begin{bmatrix} B_{\text{tr}}(j) & \mathbf{0} \\ \mathbf{0} & B_{\text{at}}(j) \end{bmatrix} \begin{bmatrix} M_{\text{tr}} \\ M_{\text{at}} \end{bmatrix} u(j) + \begin{bmatrix} 0 \\ f_{\text{at}}(j) \end{bmatrix} + \begin{bmatrix} d_{\text{tr}}(j) \\ d_{\text{at}}(j) \end{bmatrix} \quad (32)
$$

where $\delta x_{\text{tr}}(j)$ and $\delta x_{\text{at}}(j)$ are the state differences from the linearisation point at time j, $u(j) \in \mathbb{R}^8$ is a vector of thruster commands at time j, $M_{\text{tr}} \in \mathbb{R}^{3 \times 8}$ is a matrix whose columns comprise the unit vector force directions of each of the thrusters in the CRF, and $M_{\text{at}} \in \mathbb{R}^{3 \times 8}$ is a matrix whose columns comprise the unit vector torque directions of each of the thrusters in the chaser reference frame. The bias $f_{\text{at}}(j)$ arises from the linearisation of the attitude around a non-equilibrium setpoint. The biases $d_{\text{at}}(j)$ and $d_{\text{tr}}(j)$ arise from addition of a state disturbance estimate for offset-free control.

6.2. Trajectory Prediction Model

For the FTTG phase trajectory control, the LTV Yamanaka-Ankersen STM is also used, but with a shorter sampling period. Instead of an impulsive discretisation, a pulse discretisation is used, whereby a pulse of thrust is allowed for δT_s in every period of length T_s. It was found that a conventional zero-order hold discretisation did not improve accuracy. This was due to navigation errors. However, it was found that constantly commanding continuous low-level thrusts led to differential thrust being used to achieve the commanded forces, and thus excessive propellant consumption. The differential thrust occurs to counteract the effects of the minimum impulse bit (MIB). The MIB is the lowest non-zero level of thrust that can be commanded to the thrusters. The "$A_{\text{tr}}(t)$" matrix used remains

that described in (16), albeit discretised at a different rate, but the $B_{\text{tr}}(t)$ matrix is approximated by discretising the instantanous nonlinear continuous dynamics,

$$B_{\text{tr}}(t) = A_{\text{tr}}(t + \delta T_s; t + T_s)M_{12} \tag{33}$$

where

$$\begin{bmatrix} M_{11} & M_{12} \\ \mathbf{0} & I \end{bmatrix} = \exp\left(\begin{bmatrix} A_{\text{tr,c}}(t) & B_{\text{tr,c}}(t) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \delta T_s\right) \tag{34}$$

and $A_{\text{tr,c}}(t)$ is the instantanous linearised realisation of the continuous time state-space relative dynamics (described in Yamanaka & Ankersen (2002)) and $B_{\text{tr,c}}(t) = m^{-1}\begin{bmatrix} \mathbf{0}_{3\times3} & I_{3\times3} \end{bmatrix}^T$ corresponds to a force input (where m is the mass of the chaser in kilograms). The notation $A(t + \delta T_s; t + T_s)$ denotes the open-loop unforced STM from time $t + \delta T_s$ to time $t + T_s$.

6.3. Attitude Prediction Model

Whilst attitude control is not considered in the MPC design for the OSTG and INTG phases (during these phases it is assumed that target-pointing is handled independently using momentum wheels), during the very short range FTTG phase, the MPC is responsible for attitude control as well as trajectory control. A small-angle quaternion-based attitude dynamics model is used. This is similar to that used by (Hegrenæs et al., 2005), however, because an elliptical orbit must be considered, this must also be implemented as an LTV model, and must account for the rate of rotation between the velocity orbital frame (VOF) and the inertial frame.

Like the cylindrical reference frame, the velocity orbital frame (VOF) has its origin placed at the target centre of mass. However, the direction x_{vof} is aligned with $\overline{V}$. The direction y_{vof} is aligned with the direction of angular velocity of the orbit, and z_{vof} completes the right-hand set (Figure 9). The alignment of x_{vof} is advantageous when a $\overline{V}$ approach is performed in an elliptical orbit as the attitude setpoint will remain approximately constant with respect to this reference frame. Using this reference frame is advantageous when performing the final straight-line approach
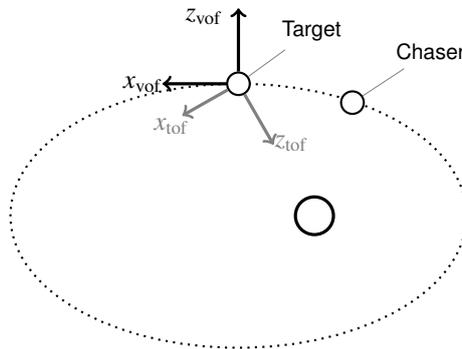

Figure 9: Velocity orbital frame (VOF)

along $\overline{V}$ because $\overline{V}$ is fixed in the co-ordinate system, even for elliptical orbits. It is also advantageous for attitude

23

pointing during this phase for the same reason. To avoid discontinuities in the attitude quaternion when approaching from "in front" of the target, a reference frame based on the VOF but rotated by π about the z_{vof} axis can be used (Bach & Paielli, 1993).

The attitude kinematics can be described using the nonlinear continuous-time model derived from Euler's moment equations

$$\frac{d}{dt}\left(\omega_{vc}^c\right)\bigg|_c = I_{\text{chs}}^{-1}\mathbf{N} - I_{\text{chs}}^{-1}\left(\left(\omega_{iv}^c + \omega_{vc}^c\right) \times I_{\text{chs}}\left(\omega_{iv}^c + \omega_{vc}^c\right)\right)$$

$$- \dot{\omega}_{iv}^c + \left(\omega_{vc}^c \times \omega_{iv}^c\right) \quad (35)$$

whilst the attitude dynamics can be described using the model

$$\begin{bmatrix} \dot{\eta} \\ \dot{\epsilon} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -\epsilon^T \omega_{vc}^c \\ \eta\omega_{vc}^c + (\epsilon \times \omega_{vc}^c) \end{bmatrix} \quad (36)$$

where ω_{vc}^c is the angular velocity between the chaser and the VOF, ω_{iv}^c is the angular velocity between the inertial frame and the VOF, $\begin{bmatrix} \eta & \epsilon^T \end{bmatrix}^T$ is a quaternion representation (e.g. Wertz (1978)) of the rotation between chaser and VOF and I_{chs} is the matrix representation of the moment of inertia of the chaser. These can be combined into a single MIMO differential equation and, as the model must be time-varying to accommodate the elliptical orbit, there is little extra effort required to re-linearise about the current attitude measurement at each sampling instant to obtain a linear local state space model

$$\delta\dot{x}_{\text{at}}(t|t_0) = A_{\text{at,c}}(t|t_0)\delta x_{\text{at}}(t|t_0) + B_{\text{at,c}}(t|t_0)\mathbf{N}(t) \quad (37)$$

where the index notation $(t|t_0)$ used here implies the prediction at time t linearised about the measurements taken at time t_0, and the deviation from the linearisation point

$$\delta x_{\text{at}}(t|t_0) = \begin{bmatrix} \omega_{vc}^c(t) \\ \eta(t) \\ \epsilon(t) \end{bmatrix} - \begin{bmatrix} \omega_{vc}^c(t_0) \\ \eta(t_0) \\ \epsilon(t_0) \end{bmatrix}. \quad (38)$$

The linearised time-varying continuous time state space matrices are defined as:

$$A_{\text{at,c}}(t|t_0) = \nabla\left(\begin{bmatrix} \frac{d}{dt}(\omega_{vc}^c(t_0))\big|_c \\ \dot{\eta}(t_0) \\ \dot{\epsilon}(t_0) \end{bmatrix}\right)\Bigg|_{\omega_{iv}^c(t),\dot{\omega}_{iv}^c(t)} \quad (39)$$

and

$$B_{\text{at,c}}(t|t_0) = \begin{bmatrix} I_{\text{chs}}^{-1} \\ \mathbf{0} \end{bmatrix}. \quad (40)$$

The instantaneous continuous time dynamics are then discretised using the same δT_s in every T_s pulse discretisation used for the trajectory model to obtain a discrete time model

$$\delta x_{\text{at}}(t + T_s|t_0) = A_{\text{at}}(t|t_0)\delta x_{\text{at}}(t|t_0) + B_{\text{at}}(t|t_0)u_{at}(t) + f_{\text{at}}(t|t_0). \quad (41)$$

The constant vector term $f_{\text{at}}(t|t_0)$ is calculated as:

$$f_{\text{at}}(t|t_0) = \begin{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t}(\omega_{vc}^c(t_0))\big|_c \\ \dot{\eta}(t_0) \\ \dot{\epsilon}(t_0) \end{bmatrix} T_s. \tag{42}$$

The chaser inertia matrix I_{chs} is not diagonal. This, in combination with the time-varying VOF, makes the calculation of the linearised $A_{at,c}$ matrix less elegant than the case where the inertia matrix is diagonal and the orbit is always circular. To calculate the Jacobian (39), the MATLAB Symbolic Toolbox is used to symbolically differentiate the non-linear model, and to cast the result as an Embedded MATLAB compliant function.

6.4. State Disturbance Estimation

In order to achieve offset-free control, the difference between the predicted state and the estimated state from navigation is integrated to estimate unmeasured disturbances:

$$\begin{bmatrix} d_{\text{tr}}(t) \\ d_{\text{at}}(t) \end{bmatrix} = \begin{bmatrix} d_{\text{tr}}(t - T_s) \\ d_{\text{at}}(t - T_s) \end{bmatrix} + W \left(\begin{bmatrix} x_{\text{tr}}(t|t) \\ x_{\text{at}}(t|t) \end{bmatrix} - \begin{bmatrix} x_{\text{tr}}(t|t - 2T_s) \\ x_{\text{at}}(t|t - 2T_s) \end{bmatrix} \right). \tag{43}$$

The comparison over two time steps is for implementational convenience. The matrix W is chosen to be (denoting $\mathbf{1}_p$ as a column vector of p ones):

$$W = \text{diag} \left(\begin{bmatrix} 0.01 \times \mathbf{1}_6 \\ 0.02 \times \mathbf{1}_7 \end{bmatrix} \right). \tag{44}$$

6.5. Cost Function

Because accuracy is the primary objective during the final translation phase, a much shorter sampling period is used. This shorter sampling period means that the control is more susceptible to navigation noise. In order to balance control accuracy against navigation uncertainty, a quadratic cost function is used (defining $j = t + kT_s$):

$$J = \sum_{k=1}^{N-1} \left((x(j|t) - r(j))^T Q(x(j|t) - r(j)) + \Delta u^T(j|t) R \Delta u(j|t) \right). \tag{45}$$

Again, the prediction model assumes a unit delay on inputs, and prediction matrices of structure (6). The change in control input Δu is included in the cost function rather than the absolute control input u. Initially this looks to not make a huge amount of physical sense, but it should be noted that the priority during this phase is accuracy, not explicit minimisation of propellant consumption. The penalty on Δu allows a position reference setpoint to be tracked without requiring it to be an unforced equilibrium (e.g. Maciejowski (1998)). In combination with the inclusion of an estimated state disturbance in the model, this allows offset free tracking in the presence of constant disturbances as long as sufficient control authority is available.

Because the prediction models for the elliptical scenario are LTV, an infinite horizon LQR terminal cost (Rawlings & Muske, 1993; Chmielewski & Manousiouthakis, 1996) to impose stability is not directly applicable, so for the purposes of this design no terminal cost is imposed. Stability is obtained by tuning Q and R, and verified experimentally.

6.6. Input Constraints

Due to the larger number of decision variables, and the higher sampling frequency, it is necessary to be frugal with the number of constraints imposed. These are limited to positivity and saturation (with maximum force F_{max}) constraints on each thruster:

$$\begin{bmatrix} I_8 \\ -I_8 \end{bmatrix} u(j) \leq \begin{bmatrix} F_{\text{max}} \mathbf{1}_8 \\ \mathbf{0}_8 \end{bmatrix}. \tag{46}$$

No terminal constraint is imposed, as it was found that a terminal equality constraint caused high gains and excessive fuel consumption.

6.7. Tuning

Control accuracy is crucial during the FTTG phase, therefore T_s must be much shorter than in the other phases. A minimum practical sampling period of $T_s = 1$ s is determined by that of the thruster subsystem. However, this would also mean that the pulse-length δT_s would need to be 1 s, reducing to a zero-order hold. Experimentation showed that $T_s = 3$ s with $\delta T_s = 1$ s yielded propellant consumption competitive with HARVD. Given the shorter sampling period, the length of the (fixed) prediction horizon N is constrained from above by computational availability. However, it was found that too short a prediction horizon made tuning to avoid underdamping in the closed-loop system very difficult. As a compromise, a value of $N = 15$ has been chosen.

7. Collision-Avoidance Manœuvre (CAM)

The objective of the collision avoidance manœuvre is to move the chaser away from the target in event of a contingency. The chaser must reach a distance of 500 m in no more than 3 orbits. The CAM must actively increase the extent of out-of-plane motion to improve passive safety. The CAM operates over similar ranges to INTG and should generate trajectories with similar characteristics, however the requirements are sufficiently different that it must be implemented as a separate phase.

As for INTG, the Yamanaka-Ankersen STM (Yamanaka & Ankersen, 2002) is used for the prediction model. The CAM demands rapid response, so unlike other phases, a delay of sampling period T_s is not included in the prediction model. Instead it is assumed that the impulsive thruster actuation will happen instantaneously (and in any case, for successful collision avoidance the first move of the manœuvre must indeed happen as soon as possible). Computation time is therefore at a premium. In addition, CAM ought not to rely on frequent information, as this may be faulty. If target-pointing is lost, relative navigation will degrade, being based on propagation of last known measurements. As such, it is desirable for the CAM to consist of two thruster burns at either end of the trajectory rather than a more complicated trajectory. This is imposed in the constraints, and has the added computational advantage that passive safety is retained without the need for a free-drift trajectory to be predicted at every time step. The CAM controller therefore employs just a single LP optimisation to generate a trajectory that is then implemented open-loop.

The flag indicating "success" of the CAM is triggered by the signal from navigation (even if possibly estimated) indicating a separation of greater than 500 m has been reached. At this point, INTG is re-enabled, either holding the chaser on a periodic trajectory a safe distance from the target, or re-commencing the impulsive approach. (Thus, technically the second burn of the CAM is performed by INTG rather than CAM.)

7.1. Cost Function

The state vector for CAM is the same as for INTG and the cost function is as follows (letting $j = t + kT_s$):

$$J = \sum_{k=0}^{N} \|E_c\,(x(j|t) - r(j))\,\|_1 + w_u\|u(j|t)\|. \tag{47}$$

The matrix E_c selects only the out-of-plane component of the position

$$E_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{48}$$

and the reference signal $r(j)$ is defined as

$$r(j|t) = \begin{bmatrix} 0 & r_e \cos v_{\text{tgt}}(j) & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{49}$$

where r_e is the desired peak amplitude of the out-of-plane component of the trajectory.

7.2. Terminal Constraint

The terminal constraint for CAM imposes that the in-track separation should be greater than a minimum value instantaneously, and at 1/4, 1/2 and 3/4 orbits into the future, and should be nominally periodic. In addition, an imposition is placed on the out-of-plane motion such that it is between $0.9r_e \cos v_{\text{tgt}}(t+NT_s)$ and $1.1r_e \cos v_{\text{tgt}}(t+NT_s)$. This, in combination with two possible approach sides leads to four possible terminal constraints (depending on the sign of $\cos v_{\text{tgt}}(t)$ and the direction of approach). The correct one to use can be determined from the current navigation signals. For brevity these will not be written in full.

7.3. Running Constraints

Input constraints are (defining u_{lim} as for INTG):

$$-u_{\text{lim}} \leq u_i(j|t) \leq u_{\text{lim}}, \qquad\qquad i \in \{1, 2, 3\}, \quad j \in \{t, t + NT_s\}$$

$$u_i(j|t) = 0, \qquad\qquad i \in \{1, 2, 3\}, \quad \text{otherwise.}$$

In addition at each time step, safety-sphere avoidance constraints are imposed as a simplified form of those used for INTG:

$$\begin{bmatrix} -\text{sgn}(x_{\text{crf}}(t)) & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x(j|t) \leq -R_{\mathcal{S}}(t). \tag{50}$$

This means that CAM is not permitted to cross above or below the target, to further reduce the collision risk. Constraint tightening is then performed on these constraints in the same manner as for INTG.

8. Controller Implementation

The MPC controllers are implemented in Simulink using MPCTOOL — an extended version of the MATLAB MPC Toolbox (Bemporad et al., 2009), also developed within the ORCSAT project. This allows the controllers to be used directly with the simplified rendezvous performance simulator (SRPS) and the functional engineering simulator (FES) as well as for C-code to be generated through auto-coding tools. The MPCTOOL extends the standard MPC Toolbox to include a number of features demanded by the ORCSAT scenario. In particular it provides MPC for linear time-varying (LTV) systems based upon LP as well as QP optimisation, to suit the LTV dynamics and 1-norm fuel costs present in this problem.

8.1. Prediction Model, Cost and Constraints Definition

The LTV prediction models, cost matrices and constraint definitions are constructed in Embedded MATLAB (EML). The use of EML enables the controller design to be carried out within a Simulink environment, and to be automatically converted to platform independent C-code using Real-Time Workshop, permitting the controllers to be implemented on any sufficiently capable supported embedded microcontroller, in line with the envisaged provisioning workflow (Colmenarejo-Matellano et al., 2008).

8.2. Numerical Solvers

In addition to the Dantzig-Wolfe QP solver on which the MPC Toolbox is based, a revised dual-simplex LP solver (Bertsimas & Tsitsiklis, 1997) has been implemented in Embedded MATLAB to solve the 1-norm based constrained optimisation problem associated with the OSTG and INTG mission phases. The LP solver uses steepest-edge pricing to minimise the number of iterations required, an η-file approach (Chvátal, 1983) to reduce the number of mathematical operations required each time the basis is changed, and Harris' two-pass ratio test (Harris, 1973) in order to be robust to the ill-conditioned problem data stemming from the combination of integrating prediction models and long prediction horizon.

8.3. Variable Horizon Implementation

Rather than implementing a bank of $N_{\max}$ individual fixed horizon MPC controllers with horizons 1 to $N_{\max}$, MPCTOOL allows the prediction horizon to be specified through an input signal (up to a pre-configured maximum value configured in the Simulink "block mask", which determines the static memory allocation for the prediction matrices). Therefore, a Simulink *For loop iterator* block can be used to iterate through horizon lengths.

8.4. Parameter Reconfiguration

Each of the prediction models is parameterised in terms of the Keplerian elements of the target orbit. Because the orbit of the target will be unknown until it is detected, this information must be obtained from navigation data. Each of the MPC controllers implemented obtains this data from a time-varying input parameter vector. For the OSTG and
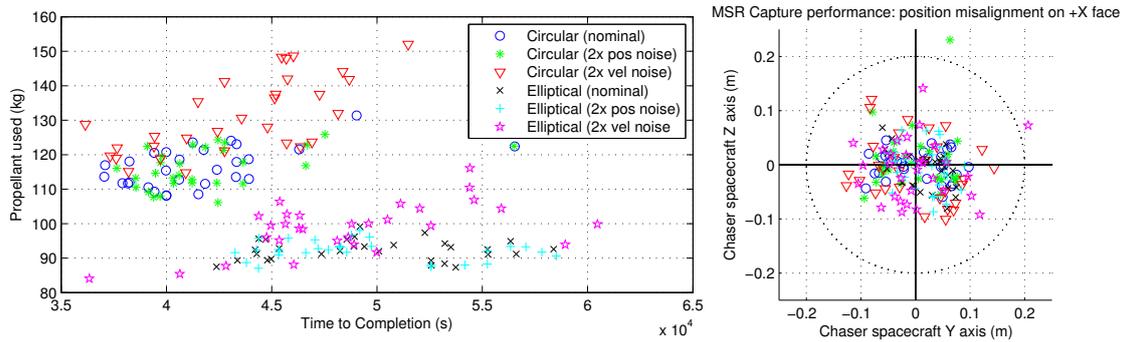
Figure 10: Sensitivity to navigation uncertainty

FTTG phases, these parameters are updated with the latest navigation data each time the controller is evaluated. For the INTG phase, apart from the true anomaly, which is updated at each sampling instant, these parameters are updated at the beginning of each transfer from one hold point to the next only.

8.5. ΔV Impulse Realisation

The prediction models for the INTG and OSTG phases model the control inputs as impulsive ΔV in the CRF and COF respectively. These are realised as finite duration, finite amplitude accelerations. In order to approximate the impulse as closely as possible, the maximum allowable acceleration is realised for as short a period as possible in multiples of 1 s. If the total ΔV cannot be realised as an exact multiple of the maximum acceleration over 1 s time frames, the acceleration during the final second is reduced so that the exact commanded ΔV is realised.

8.6. Computational Profiling

Since the ultimate objective is for the MPC controllers to be implemented on embedded hardware, it is important to profile the computational load. This has been done using the Simulink profiling tool, and by stand-alone evaluation of the Embedded MATLAB implementation of the prediction model builder and either the linear or quadratic solver code (as appropriate for the mission phase). The estimated time on a desktop computer is obtained by profiling all components of each MPC module other than the EML model builder and QP/LP solver on a desktop computer (2.8 GHz Intel Mac Pro, using the gcc 4.0.1 C compiler with -O3 optimisations) using the Simulink profiling tool to get average values. Worst case times are then obtained from stand-alone evaluation of the model builder and QP/LP solvers compiled using `emlc` and run from the MATLAB command line using representative data logged from a closed-loop simulation. Use of BLAS libraries is disabled when profiling using Simulink and code compiled using `emlc`, in order that the code be representative of that which would be generated for an Embedded Target. A rough scaling is then performed to estimate how the controller might perform on an AT697E (LEONFT) microcontroller running at 80 MHz and a PowerPC 750FX v2.3 microcontroller running at 798 MHz. The scaling is performed based upon running the Netlib Whetstone Benchmark (Painter, 1998) on the desktop computer, and using the Whetstone results provided by Thales Alenia Space Italia for the embedded processors.

29

The results from this profiling are summarised in Table 4, which shows the amount of time allowed to perform calculation, the time taken on desktop hardware, estimated times on the two embedded microprocessors and the estimated fraction of the allowed time that would have been consumed by the computation. (As a sidenote, the OSTG phase is allowed 300 s rather than the full 600 s sampling period due to the way the Simulink block has been implemented.)

Module	Quota	Desktop T	LEON2 T (Scaled)	Frac.	PowerPC T (Scaled)	Frac.
OSTG	300 s	7.8575 s	410.9571 s	136.99%	23.7401 s	7.91%
INTG	300 s	1.9028 s	99.5188 s	33.17%	5.7490 s	1.92%
FTTG	3 s	0.0885 s	4.6287 s	154.29%	0.2674 s	8.91%
CAM	5 s	0.1249 s	6.5324 s	130.6%	0.3774 s	7.55%

Table 4: Scaled computation times on various microprocessors

It can be seen that the LEON2 is unlikely to be suitable for implementation of this controller, but the PowerPC microprocessor appears to be more than capable.

9. Simulation Results in the Simplified Rendezvous Performance Simulator

Extensive closed loop simulations have been performed using the SRPS, varying initial target conditions, and initial chaser-target relative conditions. Figure 10 presents the propellant consumption, completion time and capture accuracy from the nominal elliptical and circular scenario initial conditions with 30 different random number seeds for navigation uncertainty as well as equivalent data with position and velocity uncertainty doubled. As expected, doubling the velocity noise increases the spread of results, with a bias towards longer time to completion and greater propellant consumption and slightly degraded capture accuracy for circular and elliptical scenarios. Propellant consumption and completion time are less sensitive to position error, however final capture accuracy is still affected. All nominal scenarios complete correctly. However, one circular case with doubled position noise, and one elliptical case with doubled velocity noise slightly violate the desired tolerances.

Simulations with varying initial chaser-target relative conditions and initial target conditions are summarised in Figure 11. These correspond to each of the extreme points of the initial conditions summarised in Table 5 and Table 6 respectively.

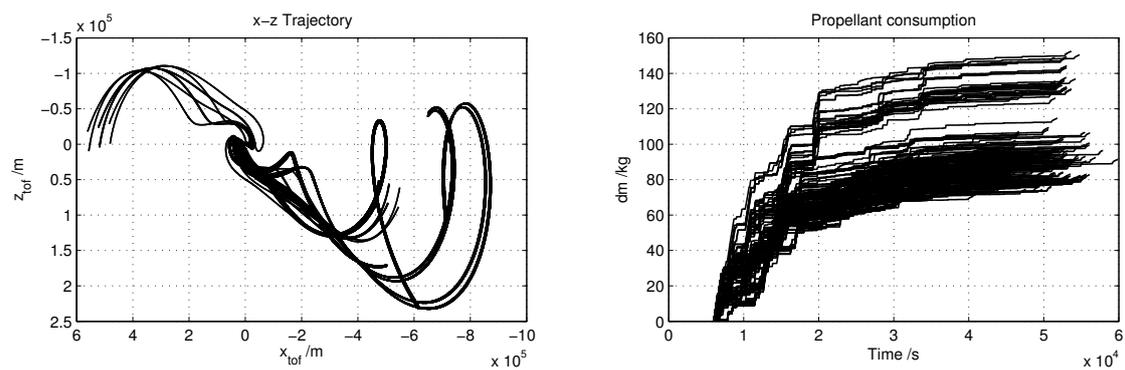|	Chaser relative elements				
	Δa	Δe	Δi	$\Delta \Omega$	$\Delta \omega$
Min	−50 km	−0.003	−0.3°	−0.3°	−0.3°
Max	50 km	0.003	0.3°	0.3°	0.3°

Table 5: Varying relative initial conditions of chaser wrt. target (nominal target position)
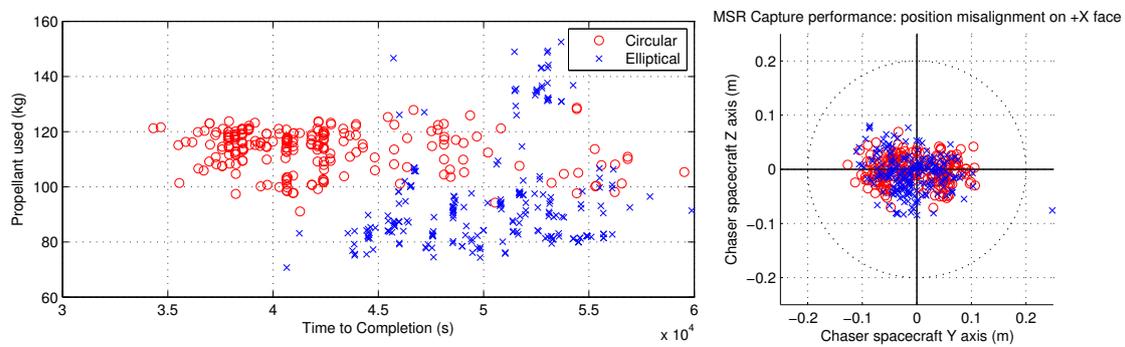
(a) Propellant consumption and capture accuracy
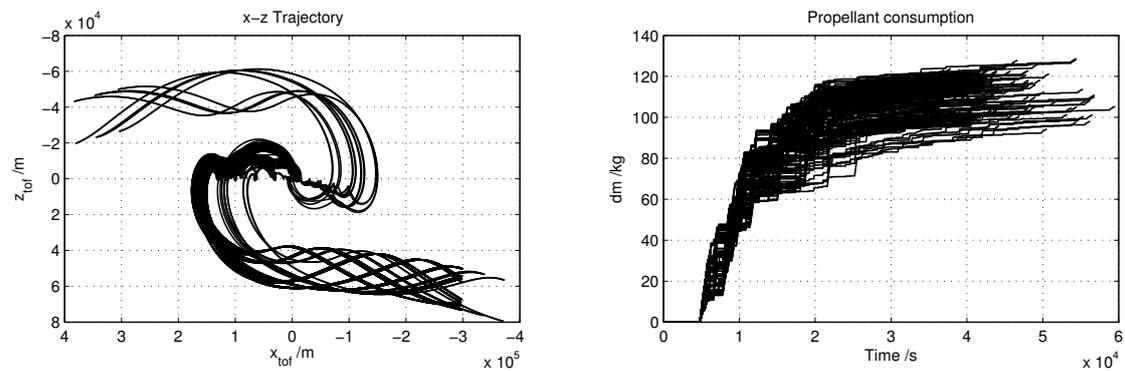
(b) Trajectories and propellant consumption (circular scenarios)

(c) Trajectories and propellant consumption (elliptical scenarios)

Figure 11: Sensitivity to initial conditions

31

			Target deviation from nominal			
	Δa_{tgt}	Δe_{tgt}	Δi_{tgt}	$\Delta \Omega_{\text{tgt}}$	$\Delta \omega_{\text{tgt}}$	$\Delta \nu_{\text{tgt}}$
Min	−10 km	-0.003	−0.05°	−0.1°	−0.1°	0°
						(+intervals of 60°)
Max	10 km	0.003	0.05°	0.1°	0.1°	320°

Table 6: Varying target position with respect to nominal conditions (nominal target-chaser separation)

It should be noted that the SRPS does not provide attitude control during OSTG and INTG phases, so the attitude pointing at the beginning of the FTTG phase may be significantly different from the desired setpoint. Despite the FTTG controller needing to correct this large error to obtain the correct alignment for capture, only a single case violates capture tolerances.

10. Controller Integration and Results

The MPC control subsystem presented in this paper has been developed using a simplified simulator of the non-linear chaser and target orbital dynamics, provided by GMV (Strippoli & Barrena, 2010). Subsequently, the design was successfully integrated by Thales Alenia Space Italia into the detailed functional engineering simulator (FES) (Le Peuvédic et al., 2008a) upon which HARVD is based, with complete models of actuator and sensor dynamics, and realistic navigation filters. This allows comparison of the performance using MPC with that achieved by the HARVD control system (Kerambrun et al., 2008) in the same simulator. It has been shown that the MPC controller is able to perform the rendezvous successfully with signals obtained from realistic sensor models and navigation filters.

A comparison of the trajectories and propellant consumption obtained using the MPC controllers and the baseline HARVD control subsystem is presented in Figure 12 (the "Ground GO" signal is held high so that holding points are briefly visited but not held). It can be seen that for both circular and elliptical orbits, the MPC implementation makes a substantial saving in propellant consumption. In the elliptical scenario a time saving of approximately half an orbit is obtained, whilst in the circular scenario the MPC takes approximately one orbit longer to complete the rendezvous.

A noticeable difference between the baseline HARVD trajectories and the MPC trajectories, visible in Figure 12, is that the MPC control system does more firing early in the rendezvous, and the OSTG phase over-shoots the target by a smaller amount.

Another significant difference, which provides a large contribution to the saving in propellant consumption, is that the OSTG phase of the MPC control system is designed to bring the chaser closer to the target, meaning that less time is spent, and a shorter distance is travelled during the INTG phase, with consequent savings in propellant. This can be observed in Figure 13 which shows the apportionment of fuel consumption between phases for the MPC design and the baseline HARVD. According to Figure 13, MPC provides a significant advantage during the OSTG phase of the elliptical scenario, and a modest advantage during the INTG phase of the elliptical scenario. For the circular scenario,

32

(a) Circular scenario ($e = 0$) trajectory

(b) Elliptical scenario ($e = 0.20441$) trajectory

(c) Circular scenario ($e = 0$) fuel usage

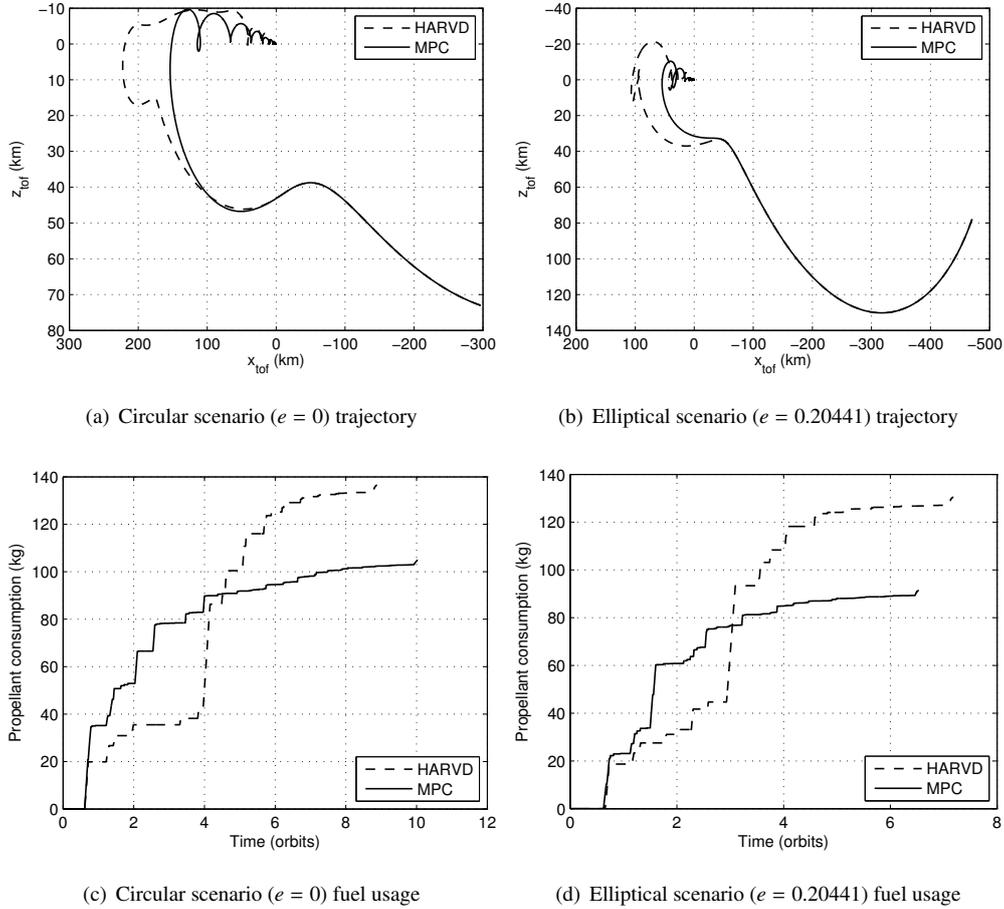(d) Elliptical scenario ($e = 0.20441$) fuel usage

Figure 12: MPC trajectory comparison with HARVD

whilst superficially it appears that little propellant saving is made during the OSTG phase, a significant saving is made overall due to the significant reduction in control effort required during the INTG phase.

A point result of capture performance of the integrated simulation is shown in Figure 14 for circular and elliptical scenarios. The requirement that the target centre of mass is within 20 cm of the centre of the capture basket is adequately met.

11. Conclusions

The work performed has shown that MPC is a capable technology for providing trajectory guidance and control for spacecraft rendezvous and capture. By using LTV prediction models, the design is capable of operation with circular and elliptical target orbits. Simulations using a detailed Functional Engineering Simulator of the plant including sensors, actuators and navigation filters, have shown that the methods presented in this paper generate trajectories that achieve non-trivial savings in propellant consumption with respect to the baseline HARVD control implementation,
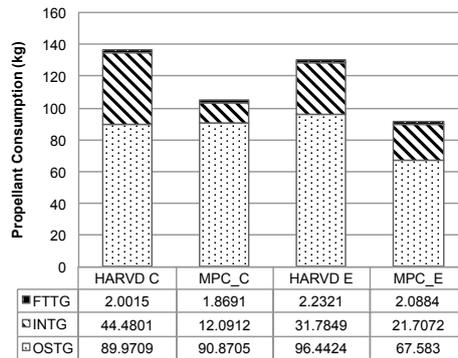
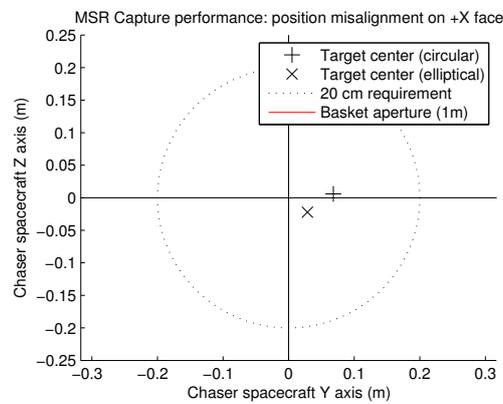Figure 13: Phase-by-phase propellant usage comparison

Figure 14: MPC capture accuracy (typical result in FES)

whilst maintaining comparable time to completion and capture accuracy.

The MPC control system has been designed to be easily implemented on embedded hardware through the use of automatic C-code generation from Embedded MATLAB, and computational profiling results indicate that the computational demands are not unrealistic.

Acknowledgements

References

Almeida, F. A. (2008). Waypoint navigation using constrained infinite horizon model predictive control. In *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*. Honolulu, Hawaii.

Bach, R., & Paielli, R. (1993). Linearization of attitude-control error dynamics. *IEEE Transactions on Automatic Control*, *38*, 1521–1525.

Battin, R. H. (1999). *An Introduction to the Mathematics and Methods of Astrodynamics, Revised Edition*. AIAA Education Series. American Institute of Aeronautics and Astronautics.

Beaty, D., Grady, M., May, L., & Gardini, B. (2008). Preliminary planning for an international Mars Sample Return mission. Report of the International Mars Architecture for the Return of Samples (iMARS) Working Group.

Bemporad, A., Borrelli, F., & Morari, M. (2002a). Model predictive control based on linear programming - the explicit solution. *IEEE Transactions on Automatic Control*, *47*, 1974–1985.

Bemporad, A., & Morari, M. (1999). Control of systems integrating logic, dynamics and constraints. *Automatica*, *35*, 407–427.

Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E. N. (2002b). The explicit linear quadratic regulator for constrained systems. *Automatica*, *38*, 3–20.

Bemporad, A., Morari, M., & Ricker, N. L. (2009). *Model Predictive Control Toolbox 3.1.1 User's Guide*. The Mathworks (2009th ed.).

Bertsimas, D., & Tsitsiklis, J. N. (1997). *Introduction to Linear Optimization*. Athena Scientific.

Bodin, P., Noteborn, R., Larsson, R., & Chasset, C. (2011). System test results from the GNC experiments on the PRISMA in-orbit test bed. *Acta Astronautica*, *68*, 862–872.

Breger, L., & How, J. P. (2007). Gauss's variational equation-based dynamics and control for formation flying spacecraft. *Journal of Guidance, Control, and Dynamics*, *30*, 437–448.

Breger, L., & How, J. P. (2008). Safe trajectories for autonomous rendezvous of spacecraft. *Journal of Guidance Control and Dynamics*, *31*, 1478–1489.

Breger, L. S. (2002). *Model predictive control for formation flying spacecraft*. Master's thesis Massachusetts Institute of Technology.

Breger, L. S., & How, J. P. (2006). Safe trajectories for autonomous rendezvous of spacecraft. In *AIAA Guidance, Navigation and Control Conference and Exhibit*. Keystone, Colorado.

Camacho, E. F., & Bordons, C. (2004). *Model predictive control*. London: Springer-Verlag.

Carter, T. E. (1998). State transition matrices for terminal rendezvous studies: brief survey and new example. *Journal of Guidance Control and Dynamics*, *21*, 148–155.

Chmielewski, D., & Manousiouthakis, V. (1996). On constrained infinite-time linear quadratic optimal control. *Systems & Control Letters*, *29*, 121–129.

Chvátal, V. (1983). *Linear Programming*. Series in mathematical sciences. W. H. Freeman.

Clohessy, W. H., & Wiltshire, R. S. (1960). Terminal guidance system for satellite rendezvous. *Journal of the Aerospace Sciences*, *27*, 653–658.

Colmenarejo-Matellano, P., Barrena-Perez, V., Modrego-Contreras, D., Le Peuvédic, C., & Guiotto, A. (2008). Integrated development, verification and validation approach for space systems using autocoding techniques. In *Proceedings of DASIA 2008 (Data Systems in Aerospace)* (pp. 189–193). Palma de Majorca, Spain.

Fehse, W. (2003). *Introduction to Automated Rendezvous and Docking of Spacecraft*. Cambridge University Press.

Ganet, M., Quinquis, I., Bourdon, J., & Delpy, P. (2002). ATV GNC during rendezvous with ISS. In *Proceedings of the 5th Cranfield DCSSS Conference*. King's College Cambridge.

Gim, D., & Alfriend, K. T. (2003). State transition matrix of relative motion for the perturbed noncircular reference orbit. *Journal of Guidance, Control, and Dynamics*, *26*, 956–971.

Harris, P. M. J. (1973). Pivot selection methods of the devex LP code. *Mathematical Programming*, *5*, 1–28.

Hegrenæs, O., Gravdahl, J. T., & Tondel, P. (2005). Spacecraft attitude control using explicit model predictive control. *Automatica*, *41*, 2107–2114.

Hohmann, W. (1925). Die Erreichbarkeit der Himmelskörper. *Oldenbourg, München*, .

Inalhan, G., Tillerson, M., & How, J. P. (2002). Relative dynamics and control of spacecraft formations in eccentric orbits. *Journal of Guidance Control and Dynamics*, *25*, 48–59.

Kaplan, M. H. (1976). *Modern spacecraft dynamics & control*. Wiley.

Kerambrun, S., Despré, N., Frapard, B., Hyounet, P., Polle, B., Ganet, M., Silva, N., Cropp, A., & Philippe, C. (2008). Autonomous rendezvous system: The HARVD solution. In *Proceedings of the 7th International ESA Conference on Guidance, Navigation & Control Systems*. Tralee, Ireland.

Larsson, R., Berge, S., Bodin, P., & Jönsson, U. (2006). Fuel efficient relative orbit control strategies for formation flying and rendezvous within PRISMA. In *Proceedings of the 29th Annual AAS Guidance and Control Conference*.

Le Peuvédic, C., Colmenarejo, P., & Guiotto, A. (2008a). Autonomous rendezvous control system: a high fidelity functional engineering simulator for GNC/AMM/FDIR validation. In *Proceedings of the 7th International ESA Conference on Guidance, Navigation & Control Systems*.

Le Peuvédic, C., Colmenarejo, P., & Guiotto, A. (2008b). *Integrated multi-range RDV control system – autonomous RDV GNC test facility – HARVD control system trade-off analysis and baseline solution*. Technical Report GMV-HARVD-TN06 GMV.

Maciejowski, J. M. (1998). The implicit daisy-chaining property of constrained predictive control. *Applied Mathematics and Computer Science*, *8*, 101–117.

Maciejowski, J. M. (2002). *Predictive Control with Constraints*. Pearson Education.

Manikonda, V., Arambel, P. O., Gopinathan, M., Mehra, R. K., & Hadaegh, F. Y. (1999). A model predictive control-based approach for spacecraft formation keeping and attitude control. In *Proceedings of the American Control Conference* (pp. 4258–4262). San Diego, CA volume 6.

Melton, R. G. (2000). Time-explicit representation of relative motion between elliptical orbits. *Journal of Guidance, Control, and Dynamics*, *23*, 604–610.

Mura, F. (2007). *Mars Sample Return: ENG-02 Mission Architecture Definition*. Technical Report European Space Agency.

Painter, R. (1998). C converted Whetstone double precision benchmark. http://www.netlib.org/benchmark/whetstone.c.

Prussing, J. E. (1991). Simple proof of the global optimality of the Hohmann Transfer. *Journal of Guidance, Control, and Dynamics*, *15*, 1037–1038.

Qin, S. J., & Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, *11*, 733–764.

Rawlings, J. B., & Mayne, D. Q. (2009). *Model predictive control: Theory and design*. Nob Hill Publishing.

Rawlings, J. B., & Muske, K. R. (1993). The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, *38*, 1512–1516.

Régnier, P., Koeck, C., Sembely, X., Frapard, B., Parkinson, M.-C., & Slade, R. (2005). Rendez-vous GNC and system analyses for the Mars Sample Return mission. In *56th International Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law*. Fukuoka, Japan.

Richards, A., & How, J. (2003a). Performance evaluation of rendezvous using model predictive control. In *AIAA Guidance, Navigation and Control Conference and Exhibit*. Austin, Texas.

Richards, A., & How, J. P. (2006). Robust variable horizon model predictive control for vehicle maneuvering. *International Journal of Robust and Nonlinear Control*, *16*, 333–351.

Richards, A. G., & How, J. P. (2003b). Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility. In *Proceedings of the 2003 American Control Conference* (pp. 4034–4040). Denver, Colorado volume 5.

Schaub, H., Vadali, S. R., Junkins, J. L., & Alfriend, K. T. (2000). Spacecraft formation flying control using mean orbital elements. *Journal of the Astronautical Sciences*, *48*, 69–87.

Shim, D. H., Kim, H. J., & Sastry, S. (2003). Decentralized nonlinear model predictive control of multiple flying robots. In *Proceedings of the 42nd IEEE Conference on Decision and Control* (pp. 3621–3626). Maui, Hawaii USA volume 4.

Sidi, M. J. (1997). *Spacecraft dynamics and control: A practical engineering approach*. Cambridge University Press.

Strippoli, L., & Barrena, V. (2010). *Simplified RVD performance simulator user's manual*. GMV (1st ed.).

Tillerson, M., Inalhan, G., & How, J. P. (2002). Co-ordination and control of distributed spacecraft systems using convex optimization techniques.

International Journal of Robust and Nonlinear Control, *12*, 207–242.

Tisserand, F. (1889). *Traité de Mécanique Celeste* volume 1. Paris: Gauthier-Villars et Fils, Imprimeurs-Libraires.

Tschauner, J. (1967). Elliptical orbit rendezvous. *AIAA Journal*, *5*, 1110–1113.

Wertz, J. R. (Ed.) (1978). *Spacecraft Attitude Determination and Control*. Kluwer Academic Publishers.

Wood, M., & Chen, W. H. (2008). Model predictive control of low Earth orbiting satellites using magnetic actuation. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, *222*, 619–631.

Yamanaka, K., & Ankersen, F. (2002). New state transition matrix for relative motion on an arbitrary elliptical orbit. *Journal of Guidance Control and Dynamics*, *25*, 60–66.