2015 European Control Conference (ECC)
July 15-17, 2015. Linz, Austria

A Tutorial on Model Predictive Control for Spacecraft Rendezvous

Edward N. Hartley[1]

Abstract— **This tutorial paper provides a review of recent advances in the field of spacecraft rendezvous using model predictive control (MPC), an advanced optimal control strategy based on on-line constrained optimisation of control inputs based on predictions of future trajectories. Firstly, the rendezvous objectives, and the generic constrained MPC problem formulation are summarised. This is followed by a discussion of how to select the three key ingredients used in an MPC design: the prediction model, the constraints and the cost function. Since MPC implementation relies on finding the solution to constrained optimisation problems in real-time, computational aspects are also briefly examined. The paper concludes with conjecture on ways the use of MPC in this application could be further advanced.**

I. Introduction

Given two spacecraft orbiting a central body, the objective of rendezvous is for the two vehicles to reach a prescribed relative configuration in each other's proximity. Often, as in rendezvous with a space-station or a Mars Sample Return (MSR) capture scenario [1], one vehicle (which we will refer to as the "chaser") is to be actively controlled, whilst the other (which we will refer to as the "target") is passive or actively maintaining a fixed orbit. The control objective is to command forces (realised e.g., via gas thrusters), to transfer the chaser into the same orbit as the target, and then approach so that a specific point on the chaser intercepts a specific point relative to the target at a safe terminal velocity (to avoid damage). This point could be a docking port, a point reachable by a robotic arm on the target, or in the case of the MSR capture scenario, a point slightly away from the target from which the final capture is performed in a free-drift manœuvre. Strategies typically used involve visiting a sequence of pre-determined waypoints using a bank of prescribed manœuvres, such as two-boost transfers (with a limited number of mid-course corrections), closed-loop controlled straight line trajectories and position keeping [2]. The ability for the chaser spacecraft to perform rendezvous autonomously without supervision from a ground station becomes critical when round-trip communication delays become long since these impede the ability to react promptly to perturbations or critical situations [3]. More detailed overviews of historical and recent spacecraft rendezvous missions and the technologies and methods involved can be found in [2], [4], [5], whilst [6], [7] describe some of the technologies used in the European Space Agency (ESA)'s Automated Transfer Vehicle (ATV), which is representative of the state-of-the-art of industrial design.

Recently there have been studies in the use of model predictive control (MPC) as a means of closed-loop feedback control to improve performance and autonomy of spacecraft rendezvous missions. MPC is a class of control techniques based on repeated solution of a (constrained) finite-horizon optimal control problem in a receding horizon manner (e.g. [8]–[13]). The ability to explicitly handle input and state constraints is often cited as the key feature, but MPC can also be used as part of an indirect adaptive control system, since the prediction model, cost function and constraints can all be updated online to reflect changes in plant parameters, constraints or objectives. The control policy takes the form of the solution to an optimisation problem that can be solved online using generic methods, and hence there is no requirement to pose a problem that admits an analytical solution. Through creative choices of constraint sets and cost functions, system designers can achieve quite complex system behaviours and meet high-level goals in a systematic way.

Such flexibility comes at the cost of increased computational load in comparison to more basic control methods. Nevertheless, with ever advancing computational hardware, and active research into more efficient algorithms, online optimisation has become less of a barrier to application, and there has recently been significant activity in exploiting both the ability to handle constraints and time-varying systems, whilst optimising a given performance metric in the context of spacecraft rendezvous. Examples of how MPC has been employed include: accommodating limited input authority (thrust constraints) [14]–[26]; using non-quadratic cost functions to achieve particular types of behaviour, for example sparse control actions [14], [15], [17], [20], [23], [24], [27]; enforcing line-of-sight constraints [15], [16], [18], [19], [21], [22], [26]; enforcing soft-docking constraints (the approach velocity reduces in line with with distance to the target) [18], [21]; collision avoidance (with the target and obstacles) [14], [16]–[18], [21], [26]; fault-tolerance by constraining open-loop unforced trajectories to achieve passive safety [16], [28]; accommodating time-varying prediction dynamics (such as those describing relative motion in elliptical orbits) [17], [27], [29], [30]; accommodating time-varying objectives and constraints (such as docking with a tumbling or rotating target) [18], [21]; fuel-efficient station or formation keeping [31], [32] and handling interaction between attitude and translation control [17], [22].

This tutorial provides a summary of recent advances in applying MPC to the translational (position) dynamics in the final phases of spacecraft rendezvous. It should be noted that attitude control is also critical, so that the final docking or capture equipment is correctly positioned and because sensors

[1] E. N. Hartley is with the Department of Engineering, University of Cambridge, Trumpington Street. CB2 1PZ. United Kingdom. enh20@eng.cam.ac.uk

978-3-9524269-3-7 ©2015 EUCA

can be highly directional. However, unlike translation control, this can also be performed using reaction wheels, which expend only solar-generated electrical power, and therefore does not limit the lifetime of the mission. In Section II we describe the generic MPC formulation; Section III highlights a selection of applicable prediction models and compares their characteristics; Section IV examines constraints; Section V considers the choice of cost function structure and tuning; Section VI discusses computational issues; and finally, Section VII presents concluding remarks.

II. Basics of model predictive control

Consider a time-varying nonlinear discrete-time system with sampling period T_s, state $x \in \mathbb{R}^{n_x}$, and input $u \in \mathbb{R}^{n_u}$, described by the difference equation

$$x(k+1) = f(x(k), u(k), k). \tag{1}$$

Assume that an estimate $\hat{x}(k)$ of the state is available. Let N be a prediction horizon over which an optimisation should be performed, and let $\ell(x(k), u(k), k)\colon \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \mapsto \mathbb{R}$ be the cost of being at state $x(k)$ and applying input $u(k)$ at time step k. If the prediction horizon is allowed to vary online, then let $N_{\max}$ be an upper bound on the prediction horizon. Let $\mathbb{X}_u(k) \subseteq \mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$ and $\mathbb{T}(k) \subseteq \mathbb{R}^{n_x}$ be (time-varying) constraint sets. At each time step k, the archetypal predictive controller solves the optimisation:

$$\min_{x_i, u_i, N} \quad J = J_N(x_N, k+N) + \textstyle\sum_{i=0}^{N-1} \ell(x_i, u_i, k+i) \tag{2a}$$

$$\begin{aligned}
\text{s.t.} \quad & x_0 = \hat{x}(k) && && \text{(2b)}\\
& x_{i+1} = f(x_i, u_i, k+i) && \forall i \in \{0, \dots, N-1\} && \text{(2c)}\\
& (x_i, u_i) \in \mathbb{X}_u(k+i) && \forall i \in \{0, \dots, N-1\} && \text{(2d)}\\
& x_N \in \mathbb{T}(k+N) && && \text{(2e)}\\
& 1 \le N \le N_{\max}. && && \text{(2f)}
\end{aligned}$$

The control law $u(k) = \kappa(\hat{x}(k)) \triangleq u_0$ is applied to the plant, and the procedure is repeated at the next sampling instant. The "standard" case of fixed prediction horizon can be achieved by solving for a fixed $N = N_{\max}$. The sampling period T_s must be chosen as a compromise between the control bandwidth, the length of the predictions horizons required, and the number of decision variables. If the computation time is more than $\approx 10\%$ of the sampling period then it is useful to introduce a unit delay and an open-loop predictor, i.e., $u(k) = \kappa(f(\hat{x}(k-1), u(k-1), k-1))$ so that $u(k)$ is computed using measurements from time $k-1$.

The state constraints, as written in (2d) are hard constraints. If it is not possible to satisfy them, the optimisation problem is infeasible and the control action is undefined in the absence of additional supervisory logic. Constraints can be "softened" to ensure feasibility of the optimisation problem by introducing additional "slack" variables measuring violation of the constraints in the optimisation, and heavily penalising this in the cost function. Violation of the original constraints becomes feasible, but the optimiser has an incentive to avoid this. Exact penalty functions [33] can be imposed on these slack variables to avoid unnecessary constraint violation.

III. Prediction model

To plan over future trajectories, a representative model is needed to make predictions. Both the chaser and the target are orbiting the central body, and their behaviour can be modelled using Newton's laws in an inertial reference frame. In principle this model could be applied directly to form a nonlinear MPC problem. Alternatively, Gauss's equations can be used to model the dynamics in terms of Keplerian orbital elements. Whilst conceptually simple, MPC with a nonlinear model is computationally demanding, and it is desirable if possible to find linear time-invariant, or linear time-varying approximations of the spacecraft motion.

Since the quantity of interest to be controlled is usually the *relative* position of the target and chaser, it is more commonplace to consider a relative reference frame centred on the target. When the target is in a circular orbit, the relative dynamics of the chaser with respect to the target can be expressed in a cartesian, local vertical, local horizontal (LVLH) reference frame centred on the target with one axis (z_{tgt}) pointing towards the focus of the orbit, one aligned with the angular velocity vector (y_{tgt}) and the third (x_{tgt}) completing a right-handed set. The relative behaviour is approximated locally by the linearised Hill equations, which can be discretised to obtain the Clohessy-Wiltshire (HCW) equations [2], [34]. Zero-order hold (ZOH) can be appropriate for short sampling periods, but an impulsive discretisation may be more appropriate for longer manœuvres. Denote the discretised dynamics, expressed as a linear time-invariant state space model

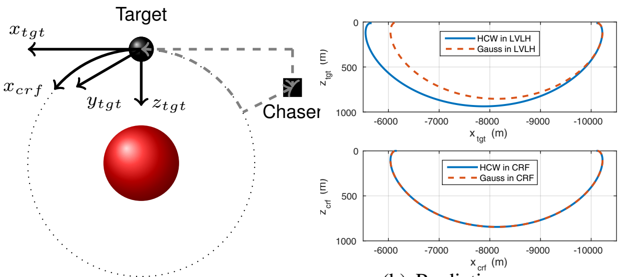
$$f(x(k), u(k), k) = Ax(k) + Bu(k) \tag{3}$$

whose state comprises of the three relative position vectors and their first derivatives with respect to time. The accuracy of the linearisation for large in-track separations can be improved by transforming measurements expressed in the local LVLH frame into a cylindrical coordinate system [35]. Figure 1(a) shows the relation between the measurements in the cartesian and cylindrical (CRF) reference frame. To emphasise the effect, Figure 1(b) shows and a comparison of the prediction in response to an impulse in the in-track direction from an equilibrium point ≈ 10 km from the target in cylindrical vs cartesian coordinates in an MSR-circular orbit [17] using the HCW equations compared with integrating the nonlinear Gauss' equations.

Alternatively to expressing forces, accelerations, or impulsive ΔV directly in the LVLH frame, it is possible to employ a (time-varying as a function of attitude) mapping matrix $M(k)$ to map the thrust directions in a reference frame mounted on the chaser body to the LVLH frame, allowing individual thrust commands to be optimised:

$$f(x(k), u(k), k) = Ax(k) + BM(k)u(k). \tag{4}$$

When the orbital eccentricity $e > 0$, the HCW equations become increasingly inaccurate over longer periods. Either the controller must be designed to be accordingly robust to the inevitable plant-model mismatch [19] or more accurate



(a) Definition

(b) Prediction accuracy

Fig. 1. Cylindrical coordinate system

prediction models should be employed. One such model is the Yamanaka-Ankersen (YA) state transition matrix (STM) [36], which is a solution of the continuous-time Tschauner-Hempel equations. This propagates the current state (defined in the same way as for the HCW equations) in the cartesian or cylindrical reference frame over the chosen period of time, and is a function of the true anomaly of the target at the start and end of the prediction period. If the target is in an ideal Keplerian orbit, this is a function of time, obtained by propagating the mean anomaly using the mean anomaly rate, then recovering the true anomaly by solving Kepler's equation [34] iteratively or by application of a trigonometric expansion known as *L'equation du centre* [37]. This is independent of the chaser control inputs, so is only solved at the point of posing the optimisation problem, not at each iteration of solution. The YA-STM does not accommodate a ZOH input discretisation, but an impulsive input is modelled by considering the input as an additive perturbation to the velocity components. The prediction dynamics are therefore of the linear time varying (LTV) form:

$$f(x(k), u(k), k) = A(k)x(k) + B(k)u(k). \tag{5}$$

Note that with this model, the prediction matrices $A(k)$, $B(k)$ will *vary throughout the prediction horizon* in (2), and not simply correspond to re-linearisation at each sampling instant.

The HCW and YA equations assume that the distance between the chaser and the target is small compared to the distance between the target and the centre of the gravity field, and break down for large out-of-plane, or radial separations. Gim and Alfriend (GA) [38] propose a STM based on propagation in terms of non-singular Keplerian orbital elements. A linearised transformation between cartesian coordinates and the orbital elements is applied to give an STM that still applies in cartesian reference frame. The STM of [38] also includes the J_2 effect caused by a non-uniform gravitational field. An alternative to the GA STM is to consider the *relative* non-singular Keplerian orbital elements between the chaser and target as the state vector, and transform the cartesian state measurement/estimate into this coordinate system using a standard nonlinear transformation [34]. The modified state vector can be propagated using Gauss's Variational Equations (GVEs) [39]. The (inverse) linearised GA transformation matrix is used to transform constraints and objectives from the cartesian frame. In this approach, inputs can be assumed to be impulsive velocity changes expressed in a second cartesian reference frame centred on the chaser. Figure 2 compares (uncontrolled) GVE and HCW predictive capability over 1 orbit in an MSR-circular scenario, from a non-equilibrium initial separation of approximately 300 km in-track and 75 km radially, translated back to the cartesian LVLH frame. The (more complex) GVE model gives better predictive capability than the HCW model using the cylindrical transformation, without which the HCW is poor. The trade between model complexity, required control update period, and prediction accuracy over the expected manœuvre duration means different models are suitable for different phases of rendezvous, as demonstrated in [17].

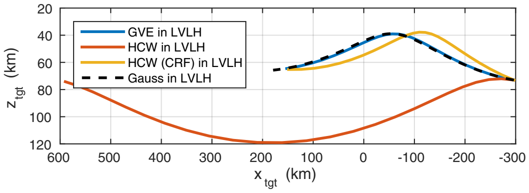


Fig. 2. Comparison of predictive accuracy of GVE vs HCW at long range

IV. Applicable Constraints

The most obvious constraints are input constraints defined in terms of the maximum thrust available. If the "input" to the model $f(x, u)$ is three forces, accelerations or impulsive ΔVs, i.e. $u = [u_x, u_y, u_z]^T$, then the following constraints may be appropriate

$$(u_x^2 + u_y^2 + u_z^2)^2 \le u_{\max}^2; \text{ or} \tag{6a}$$

$$-u_{p\max} \le u_p \le u_{p\max}, \quad p \in \{x, y, z\}. \tag{6b}$$

The first constrains the net thrust, whilst the second bounds each direction independently. If u is partitioned into 3 positive and 3 negative thrusts (which makes $1-$norm costs simple to implement) then these can be considered as:

$$0 \le u_{p+} \le u_{p\max}, \;\; 0 \le u_{p-} \le u_{p\max}, \;\; p \in \{x, y, z\}. \tag{7}$$

If an array of thrusters is mounted on the body of the spacecraft, a constraint on the individual thrusters would minimise conservatism, but the mapping between these and the force delivered in the LVLH frame varies with attitude.

A second commonly imposed constraint is a visibility cone that limits the direction of approach of the target [15], [16], [18], [19], [21], [22]. A projection of this onto the $x-z$ plane is shown in Figure 3 for an in-track approach with a cone half-angle of γ. The 3-D constraint for an in-track approach "from behind" as shown in the figure can be expressed as:

$$z_{tgt} + x_{tgt}\tan\gamma \le 0 \qquad -z_{tgt} + x_{tgt}\tan\gamma \le 0, \tag{8a}$$

$$y_{tgt} + x_{tgt}\tan\gamma \le 0 \qquad -y_{tgt} + x_{tgt}\tan\gamma \le 0. \tag{8b}$$

For different approach directions, the inequalities can be generalised by shifting and rotating the cone [18], [19], [21].

Collision avoidance constraints have also been proposed. The obstacle to avoid could be a part of the target itself, or an external object such as debris. The convex hull of the

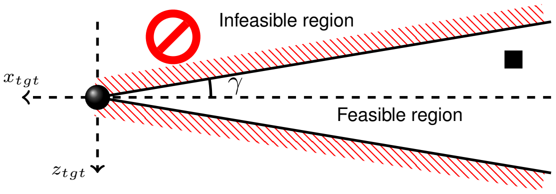


Fig. 3. Visibility cone

space occupied by the object is a compact set defined in the relative reference frame by the linear inequalities $H_o x \leq h_0$. For the chaser to remain *outside* of this set is a non-convex constraint, and imposing $H_o x(k) \geq h_o$ would be infeasible. If $dim(h_o) = n_h$, a workaround is to introduce an n_h dimensional vector $b(k)$ of binary variables, a sufficiently large scalar M, and impose the constraint:

$$H_o x(k) \geq h_o + (b(k) - 1)M \qquad \sum_{q=1}^{n_h} b_q(k) \geq 1. \tag{9}$$

This implies that $b_q(k) = 0$ allows row q of the inequality to be relaxed, but at least one row of $H_o x(k) \geq h_o$ must be active. The binary constraint implies a mixed-integer program, and M must be large enough to relax the constraint but small enough to avoid ill conditioning [15], [40]. A convex alternative is to use a time-varying halfspace constraint chosen to rotate at a pre-determined rate based on the anticipated trajectory and current state. Slightly different implementations of this approach are applied in [21] for obstacle avoidance, and [17] to avoid collision with the target. In [41] collision avoidance is also a feature but using a different approach based on analytical optimal solutions to trajectory segments. An innovative application of constraints in [16] involves not only constraining predicted trajectories to not collide with the target, but also extrapolating an open-loop prediction over a pre-chosen time period from every sampling instant and constrain these passive trajectories to also avoid collision. Thus, the constrained MPC generates trajectories that are passively safe with respect to total thrust loss. In [28], passive safety is considered in a probabilistic setting, whilst [16] considers also the possibility of active abort using subsets of available thrust directions. In a similar vein, [42] proposes an approach to guarantee feasibility of a reactive safety mode in case of changes in state constraints (e.g., due to detection of new obstacles). The purpose of the reactive safety mode is to hold the system state in a constraint-admissible invariant set to buy time for higher level decision processes. Constraint satisfaction between sampling instants is also guaranteed in [42].

Positively invariant terminal constraints $\mathbb{T}$ (2e) are a tool often used to achieve theoretical guarantees of closed-loop stability and recursive feasibility of MPC control laws [8]. For tracking control [43], [44] parameterises these in terms of the setpoint, and [45] in terms of piecewise-constant constraint bounds. When a variable prediction horizon [15] is employed, a terminal constraint is used to achieve finite-time “completion” of a manœuvre, and does not necessarily have to be invariant. It defines the “end point” of the manœuvre, and the cost function (see Section V) trades completion time against fuel usage. Constraints can also limit the approach velocity, either through a simple bound, or a “soft docking” constraint, which limits the magnitude approach velocity as a function of distance from the desired manœuvre end point [18], [21].

Modelling error, disturbances, and sensor noise mean that the predictions and the true trajectories will not exactly coincide. When there are state constraints, this can lead to infeasibility. Two complementary approaches tackle this. The first is to simply “soften” the constraints and accept a degree of constraint violation. The second is to systematically tighten constraints based on the bounds of the disturbance. Conservatism can be reduced by considering feedback in the predictions when determining the constraint tightening policy [15], [46]. Since the disturbance bounds may not be known *a priori*, in [19], a recursive estimation algorithm with a forgetting factor to accommodate time-varying behaviour is employed to estimate the corresponding mean and covariance matrices for a Gaussian distribution, which is then used to tighten nominal constraints online to achieve a specified probability of violation of the original constraints.

Another method to ensure robust constraint satisfaction is a tube-MPC [47] approach, which can be interpreted as separating the control policy into a nominal “guidance” term with tightened constraints and an explicit feedback “tracking” component which maintains the state in an admissible invariant tube around the predicted nominal trajectory [30], [48]. In tube approaches the feedback term is often a static policy that is designed *a priori*, but the guidance term is periodically re-computed in a receding horizon manner.

V. Cost function structure and tuning

Let x_s and u_s denote a state and input setpoint value. Letting notation $\|y\|_Z^2 \triangleq y^T Z y$, the classical quadratic cost function used in MPC uses the stage and terminal costs

$$\ell(x, u, k) = \|x - x_s\|_Q^2 + \|u - u_s\|_R^2, \quad J_N(x) = \|x - x_s\|_P^2 \tag{10}$$

where $Q \geq 0$, $R > 0$ and $P \geq 0$ are appropriately sized matrices. Assuming horizon N is fixed, if P is chosen to solve the appropriate Riccati equation, and there are no active constraints, then this coincides with the classical infinite-horizon linear quadratic regulator (LQR), giving a smooth closed-loop transient response and has desirable intrinsic robustness properties. In [19] Q is chosen as time-varying $Q(k)$ (with $P = 0$), to encode a prescribed arrival time.

The core MPC concept centres on explicitly optimising finitely-parameterised trajectories online, and there is no specific need, even in the absence of constraints, for a simple analytical solution to exist. This gives more flexibility in the choice of cost function than is practical for off-line control policy synthesis. As a pertinent example, to more directly encode the fuel consumption, which is directly proportional to the force delivered, a 1-norm cost function can be used:

$$\ell(x, u, k) = \|Q(x - x_s)\|_1 + \|R(u - u_s)\|_1 \tag{11a}$$

$$J_N(x) = \|P(x - x_s)\|_1. \tag{11b}$$

This particular class of cost function leads to sparser control actions, which can be preferable when thrust delivery is not continuously variable. It can be tuned to give dead-beat (minimum time) or idle (do nothing) control [49], but can also be non-robust to uncertainties and sensor noise since small perturbations in state can lead to a large perturbation in control action. In [27] a "zone-based" $1-$norm cost was used to improve robustness to uncertainties. The cost function is designed to be zero if the state is inside a hyper-cube $-b \leq x \leq b$ containing the setpoint, and a 1-norm penalty placed on the deviation s from this set:

$$\begin{aligned} &\ell(x,u) = \|Qs\|_1 + \|R(u-u_s)\|_1 \qquad (12)\\ \text{s.t.}\quad &x - x_s \leq b + s,\;\; x_s - x \leq b + s,\;\; s \geq 0. \end{aligned}$$

An alternative approach to sparsify the control action is the ℓ_{asso} cost function:

$$\ell(x,u,k) = \|x-x_s\|_Q^2 + \|u-u_s\|_R^2 + \|R_\lambda u\|_1 \qquad (13a)$$

$$J_N(x) = \|x-x_s\|_P^2. \qquad (13b)$$

This blends the quadratic and 1-norm cost, weighted by matrix $R_\lambda \geq 0$ in an attempt to inherit the robustness of the former with the sparse action of the latter. In [23] the costs (10), (11), (12), and (13) were analysed for the terminal phase of a circular MSR capture scenario, and (13) was shown to robustify a terminal-phase rendezvous trajectory tracking control law to the effects of the "minimum impulse bit" (MIB), a discontinuity in the thrust command envelope around zero.

In [22], [26] a different regularisation term is used, this time to *smooth* the response. Letting $\Delta u(k) = u(k) - u(k-1)$,

$$\ell(x,u,k) = \|x-x_s\|_Q^2 + \|u-u_s\|_R^2 + \|\Delta u\|_{R_\Delta}^2. \qquad (14)$$

The penalty on Δu (weighted by matrix $R_\Delta \geq 0$) limits the attitude manœuvres when a single thruster must be re-directed. In [22], [26], the setpoint (x_s, u_s) is virtualised as a decision variable in the optimisation and constrained to be an equilibrium pair. An additional cost term (also termed "offset cost function" in [44]) penalises deviation of this pair from the "true" setpoint, in what is described as a "reference governor" approach.

When a variable horizon is used, a terminal constraint is a compulsory part of the design, and the state error penalty is removed. Instead the stage cost includes a constant term which when summed represents a penalty on the number of time steps to reach the terminal constraint. The cost of being inside the terminal set is zero, e.g.,

$$\ell(x,u,k) = 1 + \|Ru\|_1, \quad J_N(x) = 0 \qquad \text{(linear)} \quad (15a)$$

$$\ell(x,u,k) = 1 + \|u\|_R^2, \quad J_N(x) = 0. \quad \text{(quadratic)} \quad (15b)$$

This type of cost function trades completion time against fuel usage, and can be used to enforce finite-time completion.

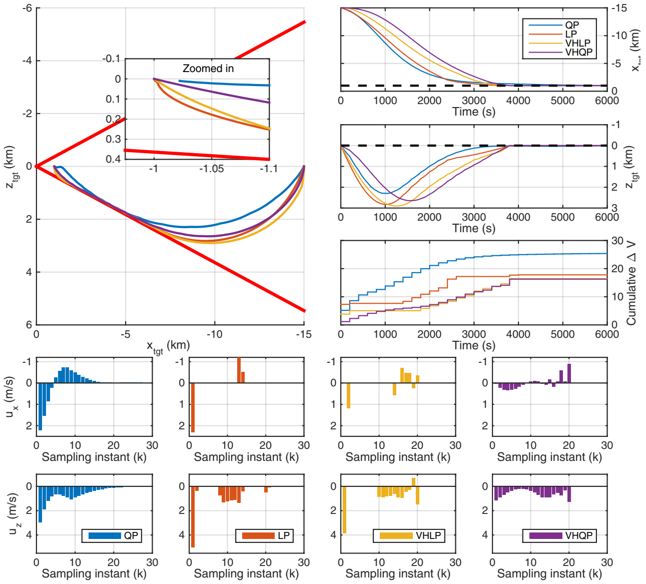


Fig. 4. Rendezvous trajectories obtained using MPC with different cost functions

Different cost functions are appropriate for different mission phases and different mechanical configurations. For example, [17] uses (15a) at longer-range where fuel optimality is the key priority, and (10) at terminal-range where robust tracking accuracy is most important. In [17], [23] multiple thrusters that can be used without re-orientation are considered and sparse control actions are preferred, whilst [22], [26] considers a scenario where a single thruster must be re-oriented and therefore the thrust direction change must be limited to avoid over-exertion of the attitude control system. Tuning the weights in any of these cost functions is an important part of the design. This can be done "by hand" based on intuition, or by gridding a limited number of parameters and analysing simulations (the approach in [17]), or using global optimisation routines to tune the cost weights to minimise a high-level heuristic functions evaluated over closed-loop simulations (as in [23]). If a good linear control law is already known and the requirement is simply to "upgrade" it with constraint handling, controller matching or reverse-engineering can be applied [50]–[52].

Figure 4 shows a simulation of MPC transferring a chaser from 15 km to 1 km from a target in an MSR circular orbit scenario, assuming a $20°$ visibility cone constraint (softened using an exact penalty), $u_{\max} = 5\text{ m/s}$ with $T_s = 200\text{ s}$ and a prediction horizon $N = 30$ with a quadratic cost (QP), a 1-norm cost (LP), and $N_{\max} = 30$ for their variable horizon counterparts (VHLP) and (VHQP). The VH examples use $\mathbb{T} \triangleq x_s$, and are tuned so that convergence happens in approximately half an orbit. Prediction and simulation use the HCW equations with impulsive ΔV discretisation. As expected the quadratic costs given a smoother response away from the constraints, whilst the $1-$norm costs give a more "bang-off-bang" input trajectory, with corrections to enforce the cone constraint. The fixed-horizon quadratic cost could be tuned to use less fuel (cumulative ΔV), but whilst initial response is fast, the final convergence is asymptotic and reaching the setpoint becomes very slow (see zoom box).

VI. Computational issues

For fixed horizon MPC, if the inequality constraints are convex and linear, the prediction model is linear and a $1-$norm cost function is used, then the optimisation problem is a linear program (LP). A (convex) quadratic cost leads to a quadratic program (QP). Additional convex quadratic constraints (e.g. (6a)), leads to a quadratically constrained quadratic program (QCQP), which can be embedded in a second-order cone program (SOCP). If the problem is time varying, the problem needs to be re-formed at each time step. Conventionally, these problems are solved using either active set (AS) or interior point (IP) methods. For embedded control, with limited computational resources, it is helpful to use tailored software that exploits the structure of the problem. Examples include CVXGEN [53] and FORCES [54] which are online code-generators to generate custom structure-exploiting IP solvers. ECOS [55] is a library-free ANSI-C tool to solve SOCPs, and in [56] automatic code generation is used to create custom IP SOCP solvers. Recently other classes of optimisation methods have been investigated, such as projected gradient methods [57] and the alternating direction of multiplier method (ADMM) [58]. Custom code generators also exist for first order methods, for example the FiOrdOs toolbox [59]. Compared to IP and AS, these involve a larger number of simpler iterations. Useful iteration bounds have also been found [57], but convergence is sensitive to conditioning, and it is worthwhile testing a selection of different solvers for a given application.

Explicit MPC [60] has been applied to time-invariant spacecraft rendezvous problems in [21], [25]. Here, multi-parametric programming is applied to compute the control law *off-line* as a piecewise-affine function. The online task is then a point location problem followed by evaluation of a local affine control law. However, the complexity becomes intractable with growing problem sizes. Another approach is to customise the computation hardware. In [24], [29], MPC is implemented in a Field Programmable Gate Array (FPGA) and applied to different phases of rendezvous in circular and elliptical orbits. This approach parallelises parts of the algorithms to reduce computation latency between measurement and control application, whilst maintaining relatively low clock rates required for robustness to effects such as solar radiation.

Variable horizons are implemented by enumerating a sequence of optimisation problems with fixed horizon N and taking the feasible solution for which the minimum cost is achieved [17], [29] or by using mixed-integer program ming (MIP) as for non-convex collision avoidance constraints with binary variables and a "big-M". MIPs are NP-complete, but one systematic and often tractable approach is to use a "branch-and-bound" method.

VII. Concluding Remarks

Recent investigations have shown overlap between the requirements of spacecraft rendezvous and the capabilities of MPC. MPC has already been tested in space by the PRISMA project [20], the interior-point solvers of [56] have already been validated for a landing scenario on a NASA test rocket, and the European Space Agency's ORCSAT project [17] investigated applicability of MPC to the MSR capture scenario. Nevertheless, there is scope for further development. For longer manœuvres, which should ideally comprise of short thrusts interspersed with long periods of free drift, performance might also be limited by the fixed-period sampling nature. Event triggered MPC [61] could be an applicable tool. Also, recent modelling developments (e.g., [62], [63]) could be applied to simplify handling of elliptical orbits. Many of the studies cited in this tutorial assume good quality state estimates with idealised uncertainty models and rigid-body models of the spacecraft. Analysis of the cross-interaction between MPC, navigation uncertainty and state estimators, and flexible modes of the vehicles will be critical to it becoming a main-stream rendezvous technology. Moreover, efficient verification, validation and clearance methods must also be investigated, and on-going algorithmic developments are likely to contribute to this task.

References

[1] P. Régnier, C. Koeck, X. Sembely, B. Frapard, M.-C. Parkinson, and R. Slade, "Rendez-vous GNC and system analyses for the Mars Sample Return mission," in *56th Int. Astronautical Congress of the Int. Astronautical Federation, the Int. Academy of Astronautics, and the Int. Inst. of Space Law*, Fukuoka, Japan, Oct 17–21 2005.

[2] W. Fehse, *Introduction to Automated Rendezvous and Docking of Spacecraft*. Cambridge University Press, 2003.

[3] D. Geller, "Orbital rendezvous: When is autonomy required?" *J. Guidance, Control, and Dynamics*, vol. 30, no. 4, pp. 974–981, 2007.

[4] D. Woffinden and D. Geller, "Navigating the road to autonomous orbital rendezvous," *J. Spacecraft and Rockets*, vol. 44, no. 4, pp. 898–909, 2007.

[5] Y. Luo, J. Zhang, and G. Tang, "Survey of orbital dynamics and control of space rendezvous," *Chinese J. Aeronautics*, vol. 27, no. 1, pp. 1–11, 2014.

[6] E. D. Pasquale, "ATV Jules Verne: a Step by Step Approach for In- Orbit Demonstration of New Rendezvous Technologies," in *Proc. SpaceOps Conference*, Stockholm, 2012.

[7] M. Ganet-Schoeller, J. Bourdon, and G. Gelly, "Non-linear and robust stability analysis for atv rendezvous control," in *Proc. AIAA Guidance, Navigation, and Control Conf.*, Chicago, Illinois, Aug 10–13 2009.

[8] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, June 2000.

[9] J. M. Maciejowski, *Predictive Control with Constraints*. Pearson Education, 2002.

[10] E. F. Camacho and C. Bordons, *Model predictive control*. London: Springer-Verlag, 2004.

[11] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Publishing, 2009.

[12] F. Borrelli, A. Bemporad, and M. Morari, "Predictive control for linear and hybrid systems," http://www.mpc.berkeley.edu/mpc-course-material/MPC_Book.pdf, Mar 2014.

[13] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 12 2014.

[14] A. Richards, J. How, T. Schouwenaars, and E. Feron, "Plume avoidance maneuver planning using mixed integer linear programming," in *AIAA Guidance, Navigation, and Control Conf. and Exhibit*, Montreal, Canada, Aug 6–9 2001.

[15] A. Richards and J. How, "Performance evaluation of rendezvous using model predictive control," in *AIAA Guidance, Navigation and Control Conf. and Exhibit*, Austin, TX, Aug 11–14 2003.

[16] L. Breger and J. P. How, "Safe trajectories for autonomous rendezvous of spacecraft," *J. Guidance, Control, and Dynamics*, vol. 31, no. 5, pp. 1478–1489, 2008.

[17] E. N. Hartley, P. A. Trodden, A. G. Richards, and J. M. Maciejowski, “Model predictive control system design and implementation for spacecraft rendezvous,” *Control Eng. Pract.*, vol. 20, no. 7, pp. 695–713, 2012.

[18] H. Park, S. Di Cairano, and I. Kolmanovsky, “Model predictive control for spacecraft rendezvous and docking with a rotating/tumbling platform and for debris avoidance,” in *Proc. American Control Conf.*, San Francisco, CA, Jun. 29 – Jul. 1 2011, pp. 1922–1927.

[19] F. Gavilan, R. Vazquez, and E. F. Camacho, “Chance-constrained model predictive control for spacecraft rendezvous with disturbance estimation,” *Control Eng. Pract.*, vol. 20, no. 2, pp. 111–122, 2012.

[20] P. Bodin, R. Noteborn, R. Larsson, and C. Chasset, “System test results from the GNC experiments on the PRISMA in-orbit test bed,” *Acta Astronautica*, vol. 68, no. 7–8, pp. 862–872, April 2011.

[21] S. Di-Cairano, H. Park, and I. Kolmanovsky, “Model predictive control approach for guidance of spacecraft rendezvous and proximity maneuvering,” *Int J. Robust Nonlin. Control*, vol. 22, no. 12, pp. 1398–1427, 2012.

[22] A. Weiss, I. Kolmanovsky, M. Baldwin, and R. S. Erwin, “Model predictive control of three dimensional spacecraft relative motion,” in *Proc. American Control Conf.*, Montreal, Canada, June 27–29 2012, pp. 173–178.

[23] E. N. Hartley, M. Gallieri, and J. M. Maciejowski, “Terminal spacecraft rendezvous and capture using LASSO MPC,” *Int. J. Control*, vol. 86, no. 11, pp. 2104–2113, 2013.

[24] E. N. Hartley and J. M. Maciejowski, “Graphical FPGA design for a predictive controller with application to spacecraft rendezvous,” in *Proc. Conf. Decision and Control*, Florence, Italy, Dec. 10–13 2013, pp. 1971–1976.

[25] M. Leomanni, E. Rogers, and S. B. Gabriel, “Explicit model predictive control approach for low-thrust spacecraft proximity operations,” *J. Guidance, Control, and Dynamics*, vol. 37, no. 6, pp. 1780–1790, 2014.

[26] A. Weiss, M. Baldwin, R. Erwin, and I. Kolmanovsky, “Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies,” *IEEE Trans. Control Syst. Tech.*, vol. (In Press), 2015.

[27] R. Larsson, S. Berge, P. Bodin, and U. Jönsson, “Fuel efficient relative orbit control strategies for formation flying and rendezvous within PRISMA,” in *Proc. 29th AAS Guidance and Control Conf.*, 2006.

[28] M. Holzinger, J. DiMatteo, J. Schwartz, and M. Milam, “Passively safe receding horizon control for satellite proximity operations,” in *Proc. 47th IEEE Conf. Decision and Control*, Cancun, Mexico, Dec 2008, pp. 3433–3440.

[29] E. N. Hartley and J. M. Maciejowski, “Field programmable gate array based predictive control system for spacecraft rendezvous in elliptical orbits,” *Optim. Control Appl. Meth.*, vol. (Article in press), 2014.

[30] G. Deaconu, C. Louembet, and A. Théron, “Minimizing the effects of navigation uncertainties on the spacecraft rendezvous precision,” *J. Guidance, Control, and Dynamics*, vol. 37, no. 2, pp. 695–700, 2014.

[31] M. Tillerson, G. Inalhan, and J. P. How, “Co-ordination and control of distributed spacecraft systems using convex optimization techniques,” *Int. J. Robust Nonlin. Control*, vol. 12, no. 20–3, pp. 207–242, 2002.

[32] P. R. A. Gilz and C. Louembet, “Predictive control algorithm for spacecraft rendezvous hovering phases,” <hal-01078506>, 2014.

[33] E. C. Kerrigan and J. M. Maciejowski, “Soft constraints and exact penalty functions in model predictive control,” in *Proc. UKACC Int. Conf. (Control 2000)*, Cambridge, UK, Sep. 2000.

[34] M. J. Sidi, *Spacecraft dynamics and control: A practical engineering approach*. Cambridge University Press, 1997.

[35] M. H. Kaplan, *Modern spacecraft dynamics & control*. Wiley, 1976.

[36] K. Yamanaka and F. Ankersen, “New state transition matrix for relative motion on an arbitrary elliptical orbit,” *J. Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 60–66, 2002.

[37] F. Tisserand, *Traité de Mécanique Celeste*. Paris: Gauthier-Villars et Fils, Imprimeurs-Libraires, 1889, vol. 1.

[38] D. Gim and K. T. Alfriend, “State transition matrix of relative motion for the perturbed noncircular reference orbit,” *J. Guidance, Control, and Dynamics*, vol. 26, no. 6, pp. 956–971, 2003.

[39] L. Breger and J. P. How, “J2-modified GVE-based MPC for formation flying spacecraft,” in *Proc. AIAA Guidance, Navigation, and Control Conf.*, vol. 1, San Francisco, CA, August 15–18 2005, pp. 158–169.

[40] A. Bemporad and M. Morari, “Control of systems integrating logic, dynamics and constraints,” *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.

[41] L. Sauter and P. Palmer, “Analytic model predictive controller for collision-free relative motion reconfiguration,” *J. Guidance, Control, and Dynamics*, vol. 35, no. 4, pp. 1069–1079, 2012.

[42] J. M. Carson III, B. Acikmese, R. M. Murray, and D. G. MacMartin, “A robust model predictive control algorithm augmented with a reactive safety mode,” *Automatica*, vol. 49, no. 5, pp. 1251–1260, 2013.

[43] D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho, “MPC for tracking piecewise constant references for constrained linear systems,” *Automatica*, vol. 44, no. 9, pp. 2382–2387, 2008.

[44] A. Ferramosca, D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho, “MPC for tracking with optimal closed-loop performance,” *Automatica*, vol. 45, no. 8, pp. 1975–1978, 2009.

[45] E. N. Hartley and J. M. Maciejowski, “Reconfigurable predictive control for redundantly actuated systems with parameterised input constraints,” *Systems & Control Letters*, vol. 66, pp. 8–15, 4 2014.

[46] A. Richards and J. P. How, “Robust variable horizon model predictive control for vehicle maneuvering,” *Int. J. Robust Nonlin. Control*, vol. 16, no. 7, pp. 333–351, 2006.

[47] D. Q. Mayne, M. M. Seron, and S. V. Rakovic, “Robust model predictive control of constrained linear systems with bounded disturbances,” *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[48] B. Acikmese, J. M. Carson, and D. S. Bayard, “A robust model predictive control algorithm for incrementally conic uncertain/nonlinear systems,” *Int. J. Robust Nonlin. Control*, vol. 21, no. 5, pp. 563–590, 2011.

[49] C. V. Rao and J. B. Rawlings, “Linear programming and model predictive control,” *J. Process Control*, vol. 10, no. 2–3, pp. 283–289, 2000.

[50] S. Di Cairano and A. Bemporad, “Model predictive control tuning by controller matching,” *IEEE Trans. Autom. Control*, vol. 55, no. 1, pp. 185–190, 2010.

[51] E. N. Hartley and J. M. Maciejowski, “Designing output-feedback predictive controllers by reverse engineering existing LTI controllers,” *IEEE Trans. Autom. Control*, vol. 58, no. 11, pp. 2934–2939, 2013.

[52] Q. N. Tran, L. Özkan, and A. C. P. M. Backx, “Generalized predictive control tuning by controller matching,” *J. Process Control*, vol. 25, pp. 1–18, 2015.

[53] J. Mattingley and S. Boyd, “CVXGEN: A code generator for embedded convex optimization,” *Optimization and Engineering*, vol. 13, no. 1, pp. 1–27, 2012.

[54] A. Domahidi, A. Zgraggen, M. N. Zeilinger, and C. N. Jones, “Efficient interior point methods for multistage problems arising in receding horizon control,” in *Proc. IEEE Conf. Decision and Control*, Maui, HI, USA, Dec 2012, pp. 668–674.

[55] A. Domahidi, E. Chu, and S. Boyd, “ECOS: An SOCP solver for embedded systems,” in *Proc. European Control Conf.*, Zurich, Jul. 17–19 2013, pp. 3071–3076.

[56] D. Dueri, J. Zhang, and B. Acikmese, “Automated custom code generation for embedded real-time second order cone programming,” in *Preprints of the 19th IFAC World Congress*, Cape Town, South Africa, 2014, pp. 1605–1612.

[57] S. Richter, C. N. Jones, and M. Morari, “Computational complexity certification for real-time MPC with input constrained based on the fast gradient method,” *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1391–1403, 2012.

[58] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[59] F. Ullmann, “A Matlab toolbox for C-code generation for first order methods,” Master’s thesis, ETH Zurich, 2011.

[60] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, “The explicit linear quadratic regulator for constrained systems,” *Automatica*, vol. 38, pp. 3–20, 2002.

[61] D. Lehmann, E. Henriksson, and K. H. Johansson, “Event-triggered model predictive control of discrete-time linear systems subject to disturbances,” in *Proc. European Control Conf.*, Zurich, Switzerland, July 17–19 2013, pp. 1156–1161.

[62] R. E. Sherrill, A. J. Sinclair, S. C. Sinha, and T. A. Lovell, “Time-varying transformations for Hill–Clohessy–Wiltshire solutions in elliptic orbits,” *Celestial Mechanics and Dynamical Astronomy*, vol. 119, no. 1, pp. 55–73, 2014.

[63] A. J. Sinclair, R. E. Sherrill, and T. A. Lovell, “Geometric interpretation of the Tschauner–Hempel solutions for satellite relative motion,” *Advances in Space Research*, vol. (In Press), 2015.