ROBUST TRAJECTORY DESIGN FOR THE HERA EXPERIMENTAL PHASE USING POLYNOMIAL EXPANSIONS

I. Fodde[1,2], J. Gil-Fernández[2], J. Feng[1], M. Vasile[1], [1]Department of Mechanical and Aerospace Engineering, University of Strathclyde, 75 Montrose Street, Glasgow G1 1XJ, United Kingdom.; iosto.fodde@strath.ac.uk; [2]Systems Department, Guidance, Navigation and Control Section (TEC-SAG), European Space Agency, Keplerlaan 1, P.O. Box 299, 2200 AG Noordwijk, The Netherlands.

Keywords: Hera, Trajectory Design, Robust Optimization, Polynomial Expansion

Introduction: The Asteroid Impact and Deflection Assessment (AIDA) collaboration, consisting of NASA's DART mission and ESA's Hera mission, aims to test the capability of a kinetic impactor to deflect an asteroid [1]. At the end of September 2022, DART successfully impacted the secondary of the binary asteroid system Didymos, called Dimorphos. Hera will launch in 2024 and aims to characterize the physical properties of Didymos and Dimorphos, and investigate the consequence of the impact made by DART in more detail. The close proximity operations of Hera consist of several different phases, each one having different scientific and technical requirements. The final nominal phase is the experimental phase, where the highest resolution images of the impact crater will be taken. As during this phase the closest fly-bys of Dimorphos will take place at around 100 meters from the body, it is important to ensure the safety of the spacecraft and minimise the risk of impact. Therefore, the trajectory design of these fly-bys needs to consider the possible execution errors of the ΔV maneuvers and the uncertainties in the dynamical system, e.g. the mass of Dimorphos.

Currently, this process is subdivided between a trajectory design step and a navigation analysis step, where first a nominal trajectory is designed and afterwards the performance when uncertainties and execution errors are included is investigated [2]. This process can be inefficient and possibly result in sub-optimal designs with over-conservative margins. Therefore, various techniques have been proposed to try and combine these two steps into one [3].

Various previous works have focused on robust trajectory design for asteroid missions. In [4], a soft landing trajectory was found using a robust optimization technique where the landing area dispersion was minimised. Similarly, [5] designed a landing trajectory using a reliability assessment involving a Monte Carlo analysis. These approaches use either linearization or Monte Carlo methods to propagate the uncertainties. As the uncertainties can be quite large in this case, the linearization technique is not accurate enough. A Monte Carlo approach is more accurate, but as the inclusion of a navigation analysis in the trajectory design process requires a lot of samples to be propagated, this can become inefficient.

More efficient uncertainty propagation (UP) methods have also been investigated. In [6] an Unscented Transform was used to propagate the first two moments of normally distributed state variables and design a robust guidance policy. The authors of [7] and [8] applied polynomial algebra based UP techniques to general trajectory optimization problems. Additionally, [9] expanded these approaches to also include the navigation and control systems and used a sparse-grid interpolation approach to perform the UP.

This work applies these techniques to the problem of the design of the very-close fly-bys (VCFB) of asteroid Dimorphos for the Hera mission, combining the trajectory design and navigation assessment to produce more robust solutions. To the authors knowledge, this is the first time this technique has been applied in an asteroid scenario, where uncertainties can play a major role.

Very-Close Fly-By: The final nominal close-proximity operations phase of Hera is the experimental phase (EXP). The main goal of the EXP is the detailed characterization of the crater made by the DART impact. It consists of three hyperbolic arcs: one incoming arc, one fly-by arc, and one outgoing arc connecting again to the start of the incoming arc. During the fly-by arc, also known as the very-close fly-by (VCFB), the spacecraft will try to image the crater at a resolution of less than 10 cm/pixel with proper lighting conditions. The lighting conditions for a successful observation are required to be:

1. Spacecraft elevation angle ($\epsilon_{s/c}$) between 20 and 70 degrees.

2. Sun elevation angle ($\epsilon_\odot$) between 25 and 75 degrees.

3. Phase angle (ϕ) between 5 and 90 degrees.

These angles are defined in figure 1, where $\hat{\vec{n}}_c$ is the crater normal, $\vec{r}_{s/c}$ the spacecraft position with respect to the crater, and $\vec{r}_\odot$ the position of the Sun with respect to the crater. An important operational requirement is to have a certain velocity margin C,

i.e. an excess velocity above the escape velocity v_{esc}, during the full VCFB, which is defined as:

$$V_{s/c} = (1+C)\sqrt{\frac{2\mu}{r}} = (1+C)v_{esc}. \quad (1)$$

Here, the value required for C is 0.4 or above, i.e. the velocity is always 1.4 times the escape velocity. This is to ensure that if a thruster failure or missed thrust event happens, the spacecraft will escape the system on a collision free trajectory.
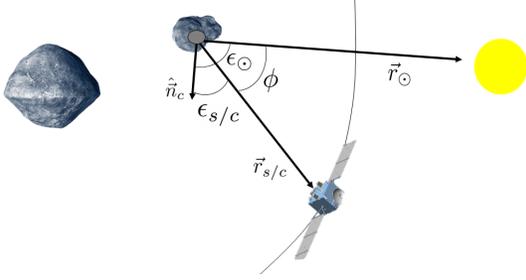
Figure 1: Definition of the various angles that influence the lighting conditions during close approach.

During the VCFB, there are three ΔV maneuvers: the first one is full open-loop (i.e. ground commanded) to get on the VCFB from the incoming arc, and the second and third one are a combined open-loop and closed-loop maneuver to decrease the perigee sequentially and correct for off-nominal conditions. The open-loop commands are calculated using two-body based hyperbolic orbits. First, a sequence of perigee distances are selected, in this case: $r_p = \{4, \ 3, \ 1.5\}$ km. From that, a set of combinations of possible hyperbolic orbits which fulfill the condition $r_i(t_f) = r_j(t_0)$ and $C_i \geq 0.4$ are determined, where t_0 and t_f are varied for each arc to find the optimal combination, where the total flight time of all three arcs is 1.5 days. From this analysis the following sequence was determined: $C = \{1.8, \ 0.8, \ 0.4\}$. An inclination is also added to the arcs, to make sure that the chance of Didymos blocking the view of Dimorphos is minimized to allow for navigation relative to Dimorphos. The resulting trajectory (propagated using the full dynamical setup, i.e. two point mass gravity models and Solar radiation pressure) is shown in figure 2.

To make sure that the observation of the crater for this nominal trajectory is successful, the various lighting angles are plotted in figure 3. During close approach all the necessary conditions are met for a successful observation of the DART crater for the nominal case.
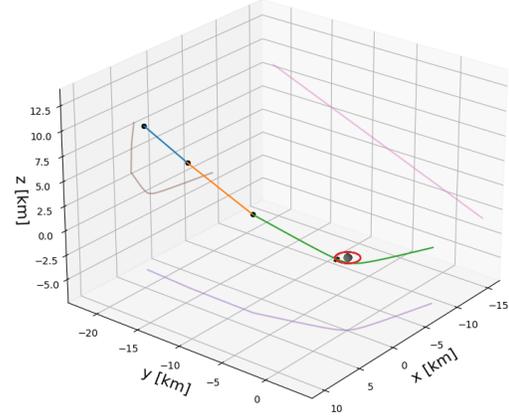
Figure 2: The nominal trajectory for the VCFB.
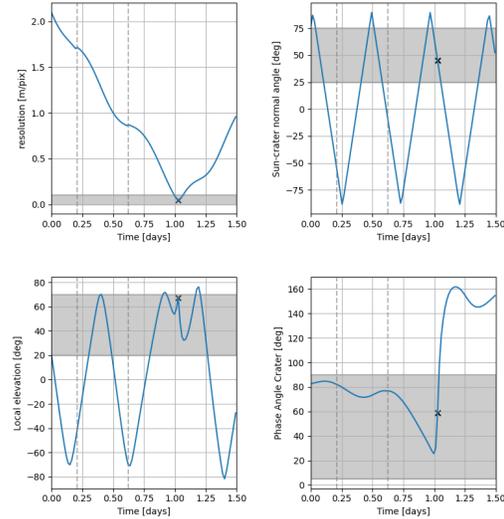
Figure 3: Evolution of lighting and observability parameters during the nominal VCFB trajectory. The vertical dashed lines are the Δ V epochs, the grey area is the region of successful observation, and the cross is the close-approach.

The closed-loop maneuvers are used to bring the true trajectory to the nominal trajectory. A simple linear targeting approach is used, which calculates the maneuver using the linearized dynamics around the nominal trajectory given by Φ as follows:

$$\begin{pmatrix} \delta\vec{r}_f \\ \delta\vec{v}_f \end{pmatrix} = \Phi \begin{pmatrix} \delta\vec{r}_0 \\ \delta\vec{v}_0 + \delta\Delta\vec{v}_0 \end{pmatrix} \qquad (2)$$

$$= \begin{pmatrix} \Phi_{1,1} & \Phi_{1,2} \\ \Phi_{2,1} & \Phi_{2,2} \end{pmatrix} \begin{pmatrix} \delta\vec{r}_0 \\ \delta\vec{v}_0 + \delta\Delta\vec{v}_0 \end{pmatrix}, \qquad (3)$$

where $\delta\vec{r}$ and $\delta\vec{v}$ are the changes in position and velocity with respect to the nominal values, and $\delta\Delta\vec{v}_0$ is the closed-loop maneuver. Eq. (3) can then be inverted to obtain $\delta\Delta\vec{v}_0$:

$$\delta\Delta\vec{v}_0 = -\Phi_{1,2}^{-1} \begin{pmatrix} \Phi_{1,1} & \Phi_{1,2} \end{pmatrix} \begin{pmatrix} \delta\vec{r}_0 \\ \delta\vec{v}_0 \end{pmatrix} \qquad (4)$$

This linear targeting controller requires an estimate of the state at the time of ΔV execution. This estimate comes from the navigation system of Hera which uses images of Dimorphos to measure its relative state. Observables like the centre of brightness are extracted from these images and fed into a navigation filter to obtain an estimate of the position and velocity together with their covariance, also known as the estimation knowledge. In this work, an analytical expression for the evolution of the knowledge covariance is given instead of a full measurement and filter simulation. This is to reduce the complexity of the implementation and improve the numerical efficiency when implemented in a trajectory optimization context. The analytical model is given as follows:

$$\vec{\sigma}_{\tilde{r}} = \begin{cases} \vec{\sigma}_{0,\tilde{r}} + \vec{\epsilon}_{\Delta V} \cdot (t - t_{\Delta V}) & t < t_1 \\ (\vec{\sigma}_{0,\tilde{r}} + \vec{\epsilon}_{\Delta V} \cdot (t - t_{\Delta V})) \cdot e^{-(t-t_1)/\tau} & t_1 \leq t < t_2 \\ \vec{\sigma}_{ss,\tilde{r}} & t_2 \leq t \end{cases}$$
$$\qquad (5)$$

$$\vec{\sigma}_{\tilde{v}} = \begin{cases} \vec{\sigma}_{0,\tilde{v}} + \vec{\epsilon}_{\Delta V} & t < t_1 \\ (\vec{\sigma}_{0,\tilde{v}} + \vec{\epsilon}_{\Delta V}) \cdot e^{-(t-t_1)/\tau} & t_1 \leq t < t_3 \\ \vec{\sigma}_{ss,\tilde{v}} = \vec{\sigma}_{ss,\tilde{r}}/\tau & t_3 \leq t \end{cases}, \quad (6)$$

where σ are the standard deviations, σ_0 the standard deviations at the start of the current arc, $\vec{\epsilon}_{\Delta V} = 2.5\% \cdot \Delta\vec{v}$ the expected error in the maneuver, $t_{\Delta V}$ is the time of the previous maneuver, τ the characteristic time of the optical navigation equal to a sixth of the orbital period of Dimorphos (T), and $t_1 = T/10$. The navigation reaches a steady-state error after a certain amount of time, which is calculated as follows:

$$\sigma_{ss,\tilde{r}} = \sqrt{\left(\frac{R}{5}\right)^2 + \sigma_{ephem}^2}, \qquad (7)$$

where R is the radius of Dimorphos and σ_{ephem} is the error in the ephemeris of Dimorphos, estimated to be 10 meters at the time of the VCFB. The time of steady-state is calculated using the following equation:

$$(\vec{\sigma}_{0,v} + \vec{\epsilon}_{\Delta V}) \cdot e^{-(t_{2/3}-t_1)/\tau} = \sigma_{ss,r/v}. \qquad (8)$$

Uncertainty Propagation: For the design of the VCFB trajectory, there are several sources of uncertainties and errors that need to be taken into account. First, the initial state of the spacecraft at the start of the VCFB will be uncertain due to execution errors from previous maneuvers to get to that point. Second, the commanded ΔV will not be exactly the same as the executed ΔV due to imperfect pointing and thruster performance. Finally, the knowledge of the state required for the linear targeting controller is not perfect, making the executed ΔV different from the desired one corresponding to the true state of the spacecraft. These uncertainties are summarised in table 1

Source	Magnitude (1-σ)
Initial position dispersion	700 m
Initial velocity dispersion	10 mm/s
Initial position knowledge	100 m
Initial velocity knowledge	0.5 mm/s
ΔV magnitude	0.33 %
ΔV angle	1.0°

Table 1: Uncertainties considered for the VCFB

To analyse the effects of these uncertainties, they need to be propagated through the system. Consider an initial value problem defined as follows:

$$\begin{cases} \dot{\vec{x}} = \vec{f}(\vec{x}(t), \vec{\beta}, t) \\ \vec{x}(t_0) = \vec{x}_0 \end{cases} \qquad (9)$$

where t is the time, $\vec{x}$ is the state vector, and $\vec{\beta}$ is a vector containing the model parameters (e.g. the ΔV parameters). Consider a set of N realisation from the uncertainties: $[\vec{x}_{0,1}, \vec{\beta}_1, ..., \vec{x}_{0,N}, \vec{\beta}_N]$. Each sample is propagated through Eq.(9) until time t_f, which results in a set of trajectories $\vec{x}_i(t_f) = \phi_i(\vec{x}_{0,i}, \vec{\beta}_i, t_f)$. The set representing all

possible trajectories at time t from the realisations of the uncertainty vector $\vec{\xi} = [\vec{x}_0, \vec{\beta}]$ is defined as:

$$\Omega_t(\vec{\xi}) = \{\vec{x}(t) = \phi(\vec{\xi}, t) \mid \vec{\xi} \in \Omega_{\vec{\xi}}\}. \qquad (10)$$

To understand the effect of the uncertainties and propagate them efficiently, an analytical expression of this set needs to be obtained. If $\vec{x}_t$ is continuous in $\vec{\xi}$ and the set is compact, $\Omega_t(\vec{\xi})$ can be approximated using a polynomial function:

$$\tilde{\Omega}_t(\vec{\xi}) = P_{n,d}(\vec{\xi}) = \sum_{i=0}^{\mathcal{N}} c_i(t)\alpha_i(\vec{\xi}), \qquad (11)$$

where $\alpha_i(\vec{\xi})$ are a set of multivariate polynomial basis functions, $c_i(t)$ are the corresponding coefficients, and $\mathcal{N} = \binom{n+d}{d}$ is the number of terms of the polynomial, where n is the degree of the polynomial and d is the number of variables. A polynomial approximation is used as it is efficient to evaluate and can be made more accurate by increasing its degree n.

Chebyshev polynomials are often used as basis functions for approximation purposes as they have several attractive numerical properties [10]. These polynomials have been previously used in an astrodynamics setting as well in [11] and [12]. This work follows a similar approach as [11] and [9], and uses a Chebyshev polynomial basis together with a Smolyak sparse grid sampling approach to obtain the polynomial from Eq. (11), which is hereafter called the non-intrusive Chebyshev Interpolation (NCI) method. As a Chebyshev basis has a finite support and the uncertainties in table 1 are assumed normally distributed, i.e. infinite support, the range of the uncertainties is bounded to $\pm 4 - \sigma$ to allow for the use of Chebyshev polynomials.

The Smolyak sparse grid was developed in [13], and selects a set of points based on the extrema of Chebyshev polynomials. An important aspect is that they do not suffer the curse of dimensionality, as the number of points grow polynomially with the dimension of the problem instead of exponentially. A more in depth explanation of this method for uncertainty propagation is given in [11].

Given the propagated samples, the coefficients of the polynomial can be obtained by inverting the following system:

$$HC = Y, \qquad (12)$$

where:

$$H = \begin{bmatrix} T_{i_1}(\vec{\xi}_1) & \dots & T_{i_s}(\vec{\xi}_1) \\ \vdots & \ddots & \vdots \\ T_{i_1}(\vec{\xi}_s) & \dots & T_{i_s}(\vec{\xi}_s) \end{bmatrix}, C = \begin{bmatrix} c_{i_1} \\ \vdots \\ c_{i_s} \end{bmatrix}, Y = \begin{bmatrix} y_1 \\ \vdots \\ y_s \end{bmatrix}$$

$$(13)$$

where $s = \mathcal{N} = \binom{n+d}{d}$, $\vec{\xi}_1, \dots, \vec{\xi}_s$ are the Smolyak sparse grid points, and Y the vector containing all the corresponding propagated samples $y_i = \phi_i(\vec{\xi}_i, t)$.

Concluding, instead of propagating a large number of samples using a Monte Carlo like approach to obtain the distribution at a later point in time, a polynomial expansion of the dynamics is first constructed using a small number of points. The desired samples for the trajectory design and navigation analysis process can then be propagated by evaluating the polynomial in Eq. (11). These two steps combined are much more efficient compared to just propagating all desired samples numerically in a Monte Carlo like fashion [14] [9].

The polynomial expansion in a Chebyshev basis represents all possible trajectories originating from the uncertainties, and does not assume any specific probability distribution. The propagation of Normally distributed variables in general requires integrating the following types of equation:

$$1/\sqrt{\pi^d |\Sigma|} \int_{-\infty}^{\infty} e^{-\frac{1}{2}(\vec{x}-\vec{\mu})^T \Sigma^{-1}(\vec{x}-\vec{\mu})} f(\vec{x}) d\vec{x}, \qquad (14)$$

where $\vec{\mu}$ are the means and Σ the covariance matrix. $f(\vec{x})$ can be various functions, e.g. $f(\vec{x}) = \vec{x}$ for the mean and $f(\vec{x}) = (\vec{x} - \vec{\mu})^T(\vec{x} - \vec{\mu})$ for the covariance. These integrals can be solved numerically using Gauss-Hermite quadrature and a change of variables ($\vec{x} = \sqrt{2}L\vec{y} + \vec{\mu}$, where $\Sigma = LL^T$ and L is determined using Cholesky decomposition), as follows [10]:

$$1/\sqrt{\pi^d} \int_{-\infty}^{\infty} e^{-\vec{y}^T \vec{y}} f(\sqrt{2}L\vec{y} + \vec{\mu}) d\vec{y} \qquad (15)$$

$$\approx \sum_{i=0}^{N} \frac{w_i}{\sqrt{\pi^d}} f(\sqrt{2}L\vec{\zeta}_i + \vec{\mu}), \qquad (16)$$

where w_i are the Gauss-Hermite weights and ζ_i the roots of the Hermite polynomial. The accuracy of the integration can be tuned by increasing the number of quadrature points. In case of propagating a set of Gaussian variables, e.g. the state knowledge, from time t_k to t_{k+1}, a large number of

quadrature samples would need to be numerically integrated. However, as a polynomial approximation is used, instead of a numerical integration only a polynomial evaluation is needed for each quadrature point, significantly reducing the computation time. For example, the mean of the state at time t_{k+1} can be calculated as follows:

$$\mu_{\vec{x}_{k+1}} \approx \sum_{i=0}^{N} \frac{w_i}{\sqrt{\pi^d}} \tilde{\Omega}_{t_{k+1}} (\sqrt{2}L\vec{\zeta_i} + \vec{\mu}). \quad (17)$$

Normally, the multivariate quadrature points are constructed as a Cartesian products of univariate ones. This method suffers from the curse of dimensionality in d, thus here a similar approach to the NCI method is used where a sparse grid is constructed instead.

Robust Trajectory Optimization: To design a trajectory that is robust against the uncertainties presented before, a trajectory optimization scheme needs to be used that implements the information on the distribution of these uncertainties. This means that the optimal control problem that is often used in the deterministic case needs to be adapted to objective and constraint functions that are dependent on the distribution of the variables. In this work, the specific methods described in [8] and [9] are adapted and applied to the case of the VCFB of Hera.

The steps of the method implemented here are graphically described in figure 4. In this section, these steps are described in more detail.

1. The initial dispersion in position and velocity from table 1 is used as an input for the NCI method to generate a polynomial mapping from the initial VCFB point in time t_0 to the time of the first ΔV maneuver $t_{\Delta V_1}$, see eq. (11).

2. A set of observations are randomly generated at time t_0. The evolution of the knowledge covariance from t_0 to $t_{\Delta V_1}$ for these observations is simulated using Eqs. (5) and (6).

3. Using Eq. 3, for each observation the closed-loop ΔV is calculated using the propagated samples of **2**. The total control is then calculated as follows:

$$\Delta \vec{V} = \Delta \vec{V}_{OL} + \delta \Delta \vec{v} + \epsilon_{\Delta \vec{V}}, \quad (18)$$

where $\Delta \vec{V}_{OL}$ is the nominal pre-computed open-loop ΔV and $\epsilon_{\Delta \vec{V}}$ is the stochastic execution error of the maneuver. The distributions in states from the sample observations, together with all the calculated $\Delta \vec{V}$ are used to define the new bounds at $t_{\Delta V_1}$, which is then propagated to $t_{\Delta V_2}$ using the

NCI method.

4. The observations and their distributions at $t_{\Delta V_1}$ are then propagated to $t_{\Delta V_2}$ using the polynomial based Hermite-Gauss quadrature of eq. (17). This does not represent the state knowledge distribution, but the actual dispersion of states, as the calculated $\Delta \vec{V}$ from the mean is applied to all realisations within the knowledge distribution.

5. All dispersion distributions at time $t_{\Delta V_2}$ are combined into one distribution by resampling (grey samples in figure 4) from all the different distributions and calculating the new mean and covariance.

6. The navigation process is again simulated using Eqs. (5) and (6) up until $t_{\Delta V_2}$ to obtain the knowledge distribution at that point in time.

7. As in step **3**, the control is again calculated using both the pre-determined open-loop and autonomous closed-loop ΔV, and applied to the full region of the knowledge distribution.

8. As in step **4**, the distributions are propagated using the quadrature method up until the time of close-approach $t_{C/A}$.

9. At close approach the various objective and constraint functions are evaluated, which are then given to the optimization algorithm to calculate the next set of decision variables. From this point, if the optimization has not converged, the process will start again using the new decision variables.

The robust optimization problem considered here is formulated as follows:

$$\min_{\vec{u}} \quad |\text{diag}(\Sigma_{C/A})|, \quad (19)$$

$$\text{s.t.} \quad \vec{x}_{k+1} = \tilde{\Omega}_{t_{k+1}}(\vec{\xi}_k), k = 0, 1, 2 \quad (20)$$

$$\mathbb{E}[\vec{x}_{C/A}] \in X_{nom} \quad (21)$$

$$\Pr(C \geq 0.4) > 99.7\% \quad (22)$$

$$\text{PoI} < 0.1\% \quad (23)$$

$$\mathbb{E}[r_f] > 10\text{km} \quad (24)$$

With this setup, the main goal is to desensitize the trajectory against uncertainties, i.e. to make the covariance during close-approach small while still being close to the nominal trajectory. The objective's goal, (19), is thus to minimize the norm of the covariance diagonal at close-approach, assuming these are the main terms contributing to the overall dispersion. The decision variables are the open-loop $\Delta \vec{V}$, the initial state $\vec{x}_0$, and the times of the two maneuvers: $t_{\Delta V_1}$ and $t_{\Delta V_2}$. Constraint (21) tries to ensure that the mean of the close-approach distribution remains close to the nominal state (within X_{nom}, which is a sphere cen-
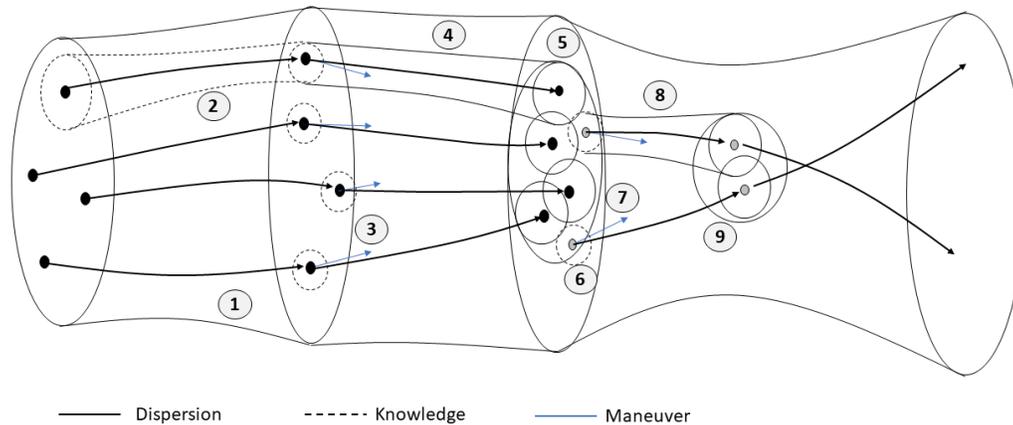
Figure 4: Diagram explaining the steps required for the transcription of the problem.

tered around the nominal trajectory with a radius of 500 meters). Constraints (22) and (23) make sure that the trajectory flown is still safe by ensuring a 3σ probability of having a velocity margin above the constraints and minimizing the probability of impact. Finally, constraint (24) makes sure that the spacecraft is far away enough from the system at the end of the third arc to allow it to safely return to a safe point far from the asteroids. The problem is solved by the WORHP solver from the Pagmo optimization package [15], using the nominal trajectory as an initial guess.

Conclusion: This research designs a robust trajectory for the very-close fly-by of the Hera mission during its final experimental phase. A novel method is used which combines the nominal trajectory design with the navigation assessment using a non-intrusive uncertainty propagation technique. This method is shown to be sufficiently efficient to be able to be used inside an optimization problem. This problem is thus able to solve for objective functions and constraints that are a function of the probability distribution of the variables. The found trajectory is less sensitive to the uncertainties and fulfills all safety constraints, without needing to perform additional navigation analyses.

References: [1] P. Michel, et al. (2022) *The Planetary Science Journal* 3(7):160 ISSN 2632-3338 `doi`. [2] C. Bottiglieri, et al. (2021) in *AAS/AIAA Astrodynamics Specialist Conference*. [3] C. Greco, et al. (2020) in *71th International Astronautical Congress (IAC 2020)*. [4] H. Hu, et al. (2016) *Aerospace Science and Technology* 48:178 ISSN 1270-9638 `doi`. [5] Y. Ren, et al. (2015) *Journal of Guidance, Control, and Dynamics* 38(9):1810 ISSN 15333884 `doi`. [6] K. Oguri, et al. (2021) *https://doiorg/102514/1G005426* 44(7):1295 ISSN 15333884 `doi`. [7] P. Di Lizia, et al. (2014) *Acta Astronautica* 93:217 ISSN 00945765 `doi`. [8] C. Greco, et al. (2020) *Acta Astronautica* 170:224 ISSN 00945765 `doi`. [9] C. Greco, et al. (2022) *Journal of Guidance, Control, and Dynamics* 45(6) `doi`. [10] W. H. Press, et al. (2007) *Numerical Recipes 3rd Edition: The Art of Scientific Computing* Cambridge University Press, USA 3rd edn. ISBN 0521880688. [11] A. Riccardi, et al. (2016) in *Advances in the Astronautical Sciences* vol. 156 707–722 Univelt Inc., Vail ISBN 9780877036296 ISSN 00653438. [12] I. Fodde, et al. (2021) in *8th International Conference on Astrodynamics Tools and Techniques - Virtual*. [link]. [13] S. A. Smolyak (1963) *Doklady Akademii Nauk SSSR* 148(5):1042. [14] C. Tardioli, et al. (2016) in *Advances in the Astronautical Sciences* vol. 156 3979–3992 ISBN 9780877036296. [15] F. Biscani, et al. (2020) *Journal of Open Source Software* 5(53):2338 `doi`.