Multi-Objective Robust Trajectory Optimization of Multi Asteroid Fly-By Under Epistemic Uncertainty *

Simão da Graça Marto[1][0000−0001−7530−3194] and Massimiliano Vasile[2][0000−0001−8302−6465]

[1] University of Strathclyde, 75 Montrose Street, G1 1XJ, Glasgow, United Kingdom
simao.da-graca-oliveira-marto@strath.ac.uk
[2] massimiliano.vasile@strath.ac.uk

Abstract. Methods are proposed and compared to generate robust optimal trajectories subject to epistemic uncertainty, meaning uncertainties that derive from a lack of knowledge on system's and launcher's parameters. This type of uncertainty is typical of the early stage of the design process when multiple options need to be evaluated and only a partial knowledge of each of them is available. The uncertainty is modelled using probability boxes (p-boxes) and lower expectation. The p-box is a family of distributions that is known to contain the real probability distribution, and the lower expectation is the minimum expectation that can be obtained with distributions within that family. We test multiple methods for efficiently estimating this quantity. These lower expectations are optimised using a Multi-Objective solver MACS (Multi Agent Collaborative Search), and with surrogate models to speed-up the optimization. Furthermore, novel dimensionality reduction methods are employed, based on control mapping, as well as a method, threshold mapping, that improves the quality of the optimization by focusing the search on target sets that produce non-trivial values of the lower expectation.

Keywords: Robust Optimization · Lower Expectation · Epistemic Uncertainty · Multi-Objective Optimization.

1 Introduction

This paper considers the problem of finding optimal trajectories subject to epistemic uncertainty in system parameters and initial conditions. In particular the paper considers the case in which the uncertainty is time dependent and an optimal control law is sought that is robust versus multiple realisations of a set of uncertain model parameters.

* This research has been developed with the partial support of the CNES grant R-S17/BS-0005-034 "Robust Optimization of Low-Thrust Interplanetary Trajectories" and the H2020 MCSA ITN UTOPIAE grant agreement number 722734.

This type of problem is characteristic of the preliminary design of small class low-cost space missions, where the uncertainty on the performance of the spacecraft is expected to be large, especially in the early stage of the system definition. When this is the case the design of a single optimal trajectory without accounting for uncertainty could lead to solutions with a high probability of failure. A direct assessment of the robustness of the solution via Monte Carlo simulation is, however, expensive, in particular in the case of epistemic uncertainty when a single distribution is unknown.

Thus in this paper we propose a method that uses Bernstein polynomials to build a representation of a family of probability distributions and compute the lower expectation on the realisation of a set of events. The trajectory is then optimised with respect to this expectation. The use of the lower expectation as cost function provides solutions that are robust against the worst case realisation of the uncertainty in the trajectory model. The paper is particularly concerned with the development of an efficient approach to the calculation of the lower expectation and optimisation of the trajectory.

Previous work on the subject was presented by Olympio and Izzo [13] using Taylor Algebra and more recently by Gerco et al. [7] using a set-oriented approach based on generalised polynomial algebra (GPA). In both cases, polynomial expansions were used to relate the final state with a set of uncertain variables. The work in [13] assumes uncertainty only in initial/final states, and each control law is optimal for each realisation of the uncertain variables. Thus it obtains a family of control laws under the assumption that uncertainty is aleatory. The work in [7], on the other hand, obtains a single robust feed-forward solution, and considers the effect of epistemic uncertainty on in the initial state.

Stochastic approximation methods were proposed by Olympio [12], which uses Robbins-Monro to enforce chance constraints, and Kiefer-Wolfowitz for stochastic minimisation. The paper applies these methods to the optimisation of a control law that is required to be robust to temporary single-engine failures, where the trajectory is re-calculated following the engine recovery, thus being a feed-back control law as well. Also in this case the uncertainty is modelled as aleatory.

In [14], Ozaki et al. proposed a method based on differential dynamic programming with an unscented transform for uncertainty propagation. A feed-back control law was introduced to optimises the expected value of a cost function that combines Δv with final distance to target. This formulation assumes that the uncertain is Gaussian and remains Gaussian when propagated forward in time, thus it can always be described by the first two statistical moments.

Following a different approach, Wachi [18] proposes a method that changes targets in the event of a mission failure. A value is attributed to each target, and its expectation is optimised using dynamic programming with a Markov decision process formulation.

In most of these methods, even though the problem is not deterministic, the assumption is still to have a complete knowledge of the probability density function associated to each uncertain variable. However, many researchers [11] have

argued that modelling uncertainty due to lack of knowledge probabilistically is inadequate. For example, if different experts suggest different probability distributions, we could use a Bayesian approach and attribute a probability to each distribution. However, this could result in severely underestimating the probability of some negative event happening, leading to a much worse design than expected.

In the work of Di Carlo et al. [5], an epistemic uncertainty formulation was proposed, using Dempster-Shaffer theory of evidence. The belief in meeting a propellant mass threshold is optimised subject to a constraint in the belief in reaching the desired final state. Quantifying epistemic uncertainty using lower expectation and Bernstein polynomials was first proposed by Vasile and Tardioli [17] using a linear optimisation approach. However, the size of the linear optimisation problem was growing exponentially with the number of dimensions. Thus, in this paper, we revisit the approach proposed in [17] so that the size of the optimisation problem grows linearly with the number of dimensions. We also combine the use of Bernstein polynomials with the idea, proposed in [5], to use a surrogate that directly maps the decision variables into the value of the lower expectation. This mapping completely bypasses the computation of the lower expectation during the search for an optimal trajectory.

We consider a simple case of a low-thrust multi-asterod fly-by tour with epistemic uncertainty in initial conditions, thrust modulus and specific impulse and also obtain results for a previously introduced asteroid rendezvous problem [9] with the same type of uncertainty.

2 Problem Formulation

In this paper we consider general problems of the following form:

$$\begin{aligned} &\min_{\mathbf{y}\in Y, \boldsymbol{\nu}\in\mathcal{N}} [-\underline{E}(h_1 < \nu_1), -\underline{E}(h_2 < \nu_2), \ldots, -\underline{E}(h_m < \nu_m), \nu_1, \nu_2, \ldots, \nu_m]^T \\ &s.t. \\ &\dot{\mathbf{x}} = \mathbf{b}(\mathbf{x}) + \mathbf{f}(\mathbf{x}, \mathbf{y}, \boldsymbol{\xi}) \end{aligned} \tag{1}$$

where $\mathbf{x}$ is the state vector, $\mathbf{y} \in Y$ is a transcription for the control variable, $\boldsymbol{\xi} \in \Xi$ is a transcription for the uncertainty, the scalar functions $h_{(\cdot)} = h_{(\cdot)}(\mathbf{y}, \boldsymbol{\xi})$ represent quantities of interest, and $\boldsymbol{\nu} = [\nu_1, \nu_2, \ldots \nu_m] \in \mathcal{N}$ is a vector of thresholds on these quantities. In this work, these quantities of interest can be the propellant mass, distance to target, or relative speed to target. The lower expectation $\underline{\mathrm{E}}$ quantifies the uncertainty, and its definition and calculation are in Section 3.

In the following, the motion will be defined in non-singular equinoctial parameters, thus we have that $\mathbf{x} = [a, P_1, P_2, Q_1, Q_2, L, m]^T$, where L is the true longitude.

In this paper we consider low thrust trajectories consisting of an ejection by a conventional launcher, followed by a number of alternating coast and thrust arcs with ion propulsion. The ejection is characterized by the departure time t_D, and the magnitude v_∞, azimuth γ and declination δ of the hyperbolic excess

velocity relative to the Earth in a heliocentric reference frame. The i^{th} coast arc is characterized by its length in longitude $\Delta L_{OFF,i}$, i.e., the difference between the longitude at the end of the arc and at the beginning. The i^{th} thrust arc is characterized by its length $\Delta L_{ON,i}$, and by the azimuth α_i and declination β_i that the spacecraft engine is pointing towards. Variables that are required to calculate a trajectory, but which are not part of the control vector, are the engine thrust at $r = 1$AU, T, and the specific impulse I_{sp}. We also optimize the times of arrival at each target $t_{T,i}$.

For more details on the transcription approach, including accuracy and computational cost please refer to [9], [20], [4] and [3].

We consider the thrust and specific impulse to be varying in an epistemically uncertain way, so these quantities are modelled as functions parameterized by epistemically uncertain parameters, $T = T(L; T_1, \ldots, T_{n_T})$ and $I_{sp} = I_{sp}(L; I_{sp,1}, \ldots, I_{sp,n_T})$. The values T_i and $I_{sp,i}$ are the values of the thrust and specific impulse at equispaced points in the trajectory, and the value of thrust and specific impulse for a particular true longitude L is obtained via linear interpolation. We also consider v_∞ as an epistemically uncertain variable, so that the uncertain vector is $\boldsymbol{\xi} = [v_\infty, T_1, \ldots, T_{n_T}, I_{sp,1} \ldots I_{sp,n_I}]$.

3 Lower Expectation

Before we go into the solution of problem (1) we introduce a procedure to compute the lower expectations. This is a critical point of our methodology that requires careful consideration because the computation of the lower expectation is an NP-hard global optimisation problem, in the general case.

We write the indicator function I of a generic quantity of interest h as:

$$I(\mathbf{y}, \boldsymbol{\xi}, \nu_h) = h(\mathbf{y}, \boldsymbol{\xi}) < \nu_h. \tag{2}$$

For clarity, throughout this section we will drop the dependency of I on $\mathbf{y}$ and ν_h.

With epistemic uncertainty, the probability distribution followed by the uncertain variables $\boldsymbol{\xi}$ is unknown. Multiple conflicting sources of information may suggest different distributions. So, combining these sources of information, a family of distributions or p-box $\mathcal{P}$ is defined so as to contain the real distribution. Thus, we quantify the lower expectation $\underline{\mathrm{E}}$ as the minimum expectation obtainable with distributions $p \in \mathcal{P}$:

$$\underline{\mathrm{E}}(I; p) = \min_{p \in \mathcal{P}} \mathrm{E}(I; p), \tag{3}$$

where $\mathrm{E}(I; p)$ is the expectation function:

$$\mathrm{E}(I; p) = \int_{\Xi} I(\boldsymbol{\xi}) p(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi}, \tag{4}$$

and Ξ is the space of the uncertain variables $\boldsymbol{\xi}$.

The formula in Equation (3) requires estimating expectation, and then finding its minimum, but first a family of distributions must be defined. For this purpose we follow [17] and use Bernstein polynomials. We could have a family composed of multi-variate Bernstein polynomials, which would include distributions of dependent variables,

$$\mathcal{P}_m = \left\{ \begin{array}{l} p_m(\boldsymbol{\xi}, \mathbf{c}) = \sum\limits_{\mathbf{j} \in \mathcal{J}} c_{\mathbf{j}} B_{\mathbf{j}} \left(\tau(\boldsymbol{\xi})\right) \\ \forall \mathbf{c} > 0 : \sum\limits_{\mathbf{j} \in \mathcal{J}} c_{\mathbf{j}} = 1 \end{array} \right\}, \tag{5}$$

or a family of distributions of independent variables where each variable's distribution is a uni-variate Bernstein distribution

$$\mathcal{P}_u = \left\{ \begin{array}{l} p_u(\boldsymbol{\xi}; \mathbf{c}) = \prod\limits_{k=1}^{n_\xi} \sum\limits_{j=0}^{q_k} c_j^{(k)} b_{j;q_k}(\tau_k(\xi_k)) \\ \forall \mathbf{c} > 0 : \sum\limits_{j} c_j^{(k)} = 1 \ \forall k \end{array} \right\}. \tag{6}$$

In our previous work [9], we showed that optimizing with either of these families is equivalent to using

$$\mathcal{P}_j = \left\{ B_{\mathbf{j},\mathbf{q}} \left(\tau(\boldsymbol{\xi})\right) \ \forall \mathbf{j} \in \mathcal{J} \subset \mathbb{N}^{n_\xi} \right\}. \tag{7}$$

where B is a multivariate Bernstein basis function (Equation (8)), scaled to be a valid probability distribution (i.e. to integrate to one over the uncertainty space Ξ), and τ is a function that maps the uncertainty space to the unit hyper-cube $[0,1]^{n_\xi}$, n_ξ being the number of uncertain variables.

$$B_{\mathbf{j},\mathbf{q}} \left(\mathbf{x}\right) = \prod_{k=1}^{n_\xi} (q_k + 1) \binom{q_k}{j_k} x^{j_k} (1-x)^{q_k - j_k}, \tag{8}$$

where the integer vector $\mathbf{q}$ indicates the order of the polynomial for each stochastic variable, and $\mathbf{j}$ is the multi-index that selects a specific function within the family. The problem of minimizing the expectation becomes one of finding the the multi-variate Bernstein basis function that minimizes the problem, and we only need to compute expectations for those distributions.

3.1 Minimizing the Expectation

Obtaining the lower expectation requires solving the optimization in Eq. (3). Since the family of distributions is $\mathcal{P} = \mathcal{P}_j$, this can be written as

$$\underline{\mathrm{E}}(I) = \min_{\mathbf{j} \in \mathcal{J}} \int_{\Xi} I(\boldsymbol{\xi}) B_{\mathbf{j},\mathbf{q}} \left(\tau(\boldsymbol{\xi})\right) \, \mathrm{d}\boldsymbol{\xi}, \tag{9}$$

where we wish to find the multi-index $\mathbf{j}$ that corresponds to the function $B_{\mathbf{j},\mathbf{q}}$ that minimizes the expectation. This is an optimization problem over integer variables.

We consider a local search method, a pattern search, as in our previous work [9], except here we consider more variants of this algorithm, and compare them. In this iterative method, if at iteration k we have the multi-index $\mathbf{j}^k$, we form a neighbourhood N_k around point $\mathbf{j}^k$ and choose the multi-index $\mathbf{j}^{k+1} \in N_k$ that minimizes $\mathrm{E}(I; B_{\mathbf{j}^{k+1}})$. This neighbourhood is composed of multi-indexes where only one index is different from $\mathbf{j}^k$, hence the term pattern search. We consider two different neighbourhoods, N_k^B and N_k^S. The former considers every multi-index that only differs from $\mathbf{j}^k$ by one index, Eq. (10). The latter only contains multi-indexes where the differing index differs by 1, Eq. (11).

$$N_k^B(\mathbf{j}^k) = \left\{\mathbf{j} \in \mathcal{J} \setminus \{\mathbf{j}^k\} : \exists m, \forall i \neq m, \mathbf{j}_i = \mathbf{j}_i^k\right\} \tag{10}$$

$$N_k^S(\mathbf{j}^k) = \left\{\mathbf{j} \in N_k^B : \exists m, \mathbf{j}_m = \mathbf{j}_m^k \pm 1\right\} \tag{11}$$

These methods always find a local optimum, but have no guarantee of finding the global optimum. The solution returned by these methods depend on the initial guess $\mathbf{j}^0$. In [9], we proposed a greedy initialization

$$\forall k,\ \mathbf{j}_k^0 = \underset{i}{\operatorname{argmin}} \int_{\Xi} I(\boldsymbol{\xi}) b_{i,q_k}(\xi_k) \mathrm{d}\boldsymbol{\xi}. \tag{12}$$

Let $\mathbf{j}_k^{f_1}$ be the outcome of our pattern search algorithm from this starting point. Again following [9], we also investigate the option of re-starting the search from the symmetric point $\mathbf{j}^{f_1+1}$:

$$j_k^{f_1+1} = q_k - j_k^{f_1} \forall k. \tag{13}$$

The solution obtained from this starting point is termed $\mathbf{j}^{f_2}$, and naturally we use the optimal of the two $j^{\omega_2} = \operatorname{argmin}_{\mathbf{j} \in \{\mathbf{j}^{f_1}, \mathbf{j}^{f_2}\}} \mathrm{E}(I; \mathbf{j})$.

We compare the results of these methods with multiple restarts from random initial points, and with Matlab's® genetic algorithm, using its default settings, i.e. a uniformly random initial population (50 if $n_\xi \leq 50$, 200 otherwise), the scattered crossover function with 0.8 crossover fraction, and a Gaussian mutation function. Because the lower expectation is always greater than or equal to zero, whenever any of these algorithms find a zero, the search is interrupted.

3.2 Estimating the Expectation

Given the difficulty of calculating the integral in Eq. (4) analytically, a quasi-Monte Carlo (qMC) approach is employed, following our previous work [9]. This is identical to a Monte Carlo approach, but instead of selecting the samples randomly, a deterministic, low-discrepancy sequence $\boldsymbol{\xi}^{(1)}, \ldots, \boldsymbol{\xi}^{(N)}$ is used. We use the Halton sequence [8] for this purpose, transformed so that it follows whichever distribution $B_{\mathbf{j}}$ we want to estimate the expectation with.

The estimate of the expectation using qMC is:

$$\hat{\mathrm{E}}_P(I;p) = \frac{1}{N}\sum_{i=1}^{N} I(\xi_i)\ , \quad \xi_i \sim p\ , \tag{14}$$

where the number of samples we use is $N = 5000$.

At each step of the pattern search algorithm, the multi-index $\mathbf{j}$ specifies the current estimate of which distribution $B_{\mathbf{j}}$ corresponds to the lower expectation. Our local search method consists in evaluating the expectation while varying one of the indices in $\mathbf{j}$ at a time. To evaluate the points in N_k^B, for instance, requires simulating the trajectory $N \cdot q \cdot n_\xi$ times.

There is overlap between the Bernstein basis distributions, which allows making use of some of the samples obtained to estimate $\mathrm{E}(I; B_{\mathbf{j}^k})$ in the estimates of $\mathrm{E}(I; B_{\mathbf{j}^{k+1}})$ for $\mathbf{j}^{k+1} \in N_k(\mathbf{j}^k)$. We now introduce a method based on rejection sampling, which we term qMC+RS to distinguish from the baseline method qMC. We choose samples to reuse based on the acceptance probability given as

$$p_a(\mathbf{x}) = \min\left(\frac{B_{\mathbf{j}^{k+1},\mathbf{q}}(\mathbf{x})}{B_{\mathbf{j}^k,\mathbf{q}}(\mathbf{x})}, 1\right). \tag{15}$$

The resulting samples will follow the distribution

$$p_r(\mathbf{x}) = \min\left(B_{\mathbf{j}^{k+1},\mathbf{q}}(\mathbf{x}), B_{\mathbf{j}^k,\mathbf{q}}(\mathbf{x})\right). \tag{16}$$

The function $p_r(\mathbf{x})$ is not normalized in the above equation, and its integral represents the fraction of re-used samples $\eta_r = \int_{[0,1]^{n_\xi}} p_r(\mathbf{x})\,\mathrm{d}\mathbf{x}$. In order to obtain samples following distribution $B_{\mathbf{j}^{k+1}}$, we sample a complementary distribution, given as

$$p_c(\mathbf{x}) = \max\left(0, B_{\mathbf{j}^{k+1},\mathbf{q}}(\mathbf{x}) - B_{\mathbf{j}^k,\mathbf{q}}(\mathbf{x})\right), \tag{17}$$

where again $p_c(\mathbf{x})$ is not normalized in the above equation, and its integral represents the fraction of new samples $1 - \eta_r = \int_{[0,1]^{n_\xi}} p_c(\mathbf{x})\,\mathrm{d}\mathbf{x}$. Note that $p_r(\mathbf{x}) + p_c(\mathbf{x}) = B_{\mathbf{j}^{k+1},\mathbf{q}}(\mathbf{x})$. Reused samples and newly obtained samples are combined and used to estimate the expectation. When the best element in N_k is selected, it is sampled with exclusively new samples, i.e. using qMC only, to avoid the accumulation of loss of accuracy, caused by applying rejection sampling with a non-smooth p_a [19].

Only one index differs between $\mathbf{j}^k$ and $\mathbf{j}^{k+1}$. Let i be this index such that $\mathbf{j}_i^k \neq \mathbf{j}_i^{k+1}$. In Table 1, the value of η_r is shown for all possible pairs $\mathbf{j}_i^k, \mathbf{j}_i^{k+1}$.

To keep their low-discrepancy properties, the samples are accepted deterministically. Let $(\mathbf{x})_k,\ k = 1 \ldots N$ be our original samples following $B_{\mathbf{j}^k,\mathbf{q}}$. We go through the individual $(\mathbf{x})_k$ in ascending order of $(\mathbf{x}_i)_k$, keeping a subtotal of the values of $p_a((\mathbf{x})_k)$. Whenever the subtotal exceeds 1, it is decremented and the sample is accepted. Otherwise it is rejected.

Table 1: Fraction of reusable points η_r, when $\mathbf{j}^k$ and $\mathbf{j}^{k+1}$ only differ for the i-th index, given for each pair $\mathbf{j}_i^k, \mathbf{j}_i^{k+1}$.

		$\mathbf{j}_i^{k+1}$				
		0	1	2	3	4
$\mathbf{j}_i^k$	0	-	0.590	0.331	0.164	0.062
	1	0.590	-	0.654	0.375	0.164
	2	0.331	0.654	-	0.654	0.331
	3	0.164	0.375	0.654	-	0.590
	4	0.062	0.164	0.331	0.590	-

4 Multi-Objective Optimization

Due to the high number of dimensions in the control law $\mathbf{y} \in \mathbf{Y}$, we follow [9] in using a control mapping strategy to reduce the number of dimensions of the search space. As such, we have a control map $\mathbf{y}(\mathbf{z})$, where the proxy control variable $\mathbf{z} \in \mathbf{Z}$ has fewer dimensions than $\mathbf{y}$, as described in Section 4.1.

As previously mentioned, the target sets are represented by a threshold ν on a metric, and thus the lower expectations are written as functions of $\boldsymbol{\nu}$, as well as $\mathbf{y}$, $\underline{\mathrm{E}} = \underline{\mathrm{E}}(\mathbf{y}(\mathbf{z}), \boldsymbol{\nu})$. To avoid the computational burden of estimating the lower expectation and computing the control map, surrogate models $\tilde{\underline{\mathrm{E}}}$ are obtained that sidestep both of these computations, by approximating $\tilde{\underline{\mathrm{E}}}(\mathbf{z}, \nu) \approx \underline{\mathrm{E}}(\mathbf{y}(\mathbf{z}), \nu)$, similarly to [5]. We use Krigging models implemented in the DACE toolbox [10]. The optimization problem (1) is thus written as

$$\min_{\mathbf{z}, \boldsymbol{\nu}} \left[-\tilde{\underline{\mathrm{E}}}(\mathbf{z}, \boldsymbol{\nu}), \boldsymbol{\nu} \right]. \tag{18}$$

We introduce a threshold mapping technique, wherein a proxy threshold value $\boldsymbol{\nu}^*$ is mapped to $\boldsymbol{\nu}$, in order to focus the search on more useful regions of the search space. This process is explained in Section 4.2. When using threshold mapping, the optimization problem becomes

$$\min_{\mathbf{z}, \boldsymbol{\nu}^*} \left[-\tilde{\underline{\mathrm{E}}}(\mathbf{z}, \boldsymbol{\nu}^*), \tilde{\boldsymbol{\nu}}(\mathbf{z}, \boldsymbol{\nu}^*) \right]. \tag{19}$$

These surrogate models are first trained with 100 initial training points. After this, we solve problem (18) or (19) using MACS [15], with 10 agents and 10 elements in the archive. After MACS runs, the points in its estimate of the Pareto front are evaluated exactly, and these exact values are added to the surrogate model's training set. These steps are repeated 10 times, such that in total we evaluate $100 + 10 \times 10$ points.

4.1 Control Mapping For Dimensionality Reduction

Here the control mapping strategy in [9] is briefly explained in "Deterministic Control Map", as well as two novel control maps in "Max-Min Control Map" and "Min-Max Control Map".

Deterministic Control Map In this control map, proposed in [9], the search space is restricted to control laws that are optimal for a deterministic setting, targeting positions and velocities in the vicinity of the targets. The only difference is that here we use multiple shooting, and thus the optimization variable includes the states at the beginning of each arc, i.e. at the nodes, $\mathbf{x}_i^{ON}$ for ON nodes and $\mathbf{x}_i^{OFF}$ for OFF nodes. The states at a node that are obtained by propagating from the previous node are written as $\left(\mathbf{x}_i^{ON}\right)^+$ and $\left(\mathbf{x}_i^{OFF}\right)^+$.

Defining the array of node states,

$$X = \left[\mathbf{x}_0, \mathbf{x}_1^{ON}, \mathbf{x}_1^{OFF}, \ldots, \mathbf{x}_{n_{LT}}^{ON}, \mathbf{x}_{n_{LT}}^{OFF}\right] \tag{20}$$

Thus the control mapping $\mathbf{y} = \mathbf{y}^*(\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v)$ is defined as the vector $\mathbf{y}$ that, along with X, solves the optimal control problem

$$\begin{aligned}
&\mathbf{y}^*(\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v), X^* = \underset{\mathbf{y}, X}{\operatorname{argmin}}\, m_p(\mathbf{y}, \boldsymbol{\xi}) \\
s.t.\quad &\mathbf{r}(X, \mathbf{y}, \boldsymbol{\xi}, (\mathbf{t_T})_r) = \mathbf{r}_r((\mathbf{t_T})_r) + (\mathbf{D}_r)_r \\
&\mathbf{v}(X, \mathbf{y}, \boldsymbol{\xi}, (\mathbf{t_T})_r) = \mathbf{v}_r((\mathbf{t_T})_r) + (\mathbf{D}_v)_r \text{ if target is not fly-by only.} \\
&\mathbf{x}_s^{ON} = \left(\mathbf{x}_s^{ON}\right)^+ \\
&\mathbf{x}_s^{OFF} = \left(\mathbf{x}_s^{OFF}\right)^+ \quad s = 1, \ldots, n_{LT} \\
&\mathbf{x}_0 = (\mathbf{x}_0)^+
\end{aligned} \tag{21}$$

where the r-th target position $\mathbf{r}_r$ and velocity $\mathbf{v}_r$ are calculated using Keplerian propagation for the target's fly-by time $(\mathbf{t_T})_r$, and the spacecraft position at flyby $\mathbf{r}(X, \mathbf{y}, \boldsymbol{\xi}, (\mathbf{t_T})_r)$ and $\mathbf{v}(X, \mathbf{y}, \boldsymbol{\xi}, (\mathbf{t_T})_r)$ are calculated for $(\mathbf{t_T})_r$, by propagating from the immediately previous node in X. The state at departure $(\mathbf{x}_0)^+$ is defined by the departure conditions v_∞, γ, δ, the last two being part of $\mathbf{y}$. The vectors $(\mathbf{D}_r)_r$ and $(\mathbf{D}_v)_r$ are the displacements in position and velocity for the r-th target. If a target is fly-by only, as in the test case in Section 5, the constraint on velocity is not applied and $(\mathbf{D}_v)_r$ is not a part of $\mathbf{z}$. If, on the other hand, the target is a rendezvous target, the constraints are in fact applied by equating the equinoctial elements, instead of position and velocity.

This restricts the search to the set $Y^* \subset Y$, given as

$$Y^* = \left\{y \in Y : y = y^*(\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v)\ \forall \xi \in \Xi, \mathbf{D}_r \in R_r, \mathbf{D}_v \in R_v\right\}, \tag{22}$$

where R_r and R_v are 3D box sets.

We also consider a further restricted search space $Y^- \subset Y^* \subset Y$, where we fix the displacements to zero

$$Y^- = \left\{y \in Y : y = y^*(\boldsymbol{\xi}, \mathbf{0}, \mathbf{0})\ \forall \xi \in \Xi\right\}. \tag{23}$$

For control maps Y^* and Y^-, we have $\mathbf{z} = [\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v]$ and $\mathbf{z} = \boldsymbol{\xi}$, respectively.

Max-Min Control Map In order to reduce the search space even further, and remove the dependency of the computational time on the number of uncertain

variables n_ξ, we test a new control map, where we restrict the search only to the control laws that correspond to a worst case scenario, that is, to the uncertain vector $\boldsymbol{\xi}$ for which the optimal control law is worse, such that the proxy control variable is given as $\mathbf{z} = [\mathbf{D_r}, \mathbf{D_v}]$. We define it as

$$\begin{aligned} \mathbf{y}^B(\mathbf{D}_r, \mathbf{D}_v) &= \mathbf{y}^*(\boldsymbol{\xi}_M, \mathbf{D}_r, \mathbf{D}_v) \\ s.t.\ \boldsymbol{\xi}_M &= \underset{\boldsymbol{\xi}}{\operatorname{argmax}}\, m_p(\mathbf{y}^*(\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v), \boldsymbol{\xi}). \end{aligned} \tag{24}$$

The value of $\boldsymbol{\xi}_M$ is defined through a max-min problem, a bilevel problem where only the lower level, the solution of $\mathbf{y}^*$, is subject to constraints.

Similarly to the work in [16], we use a surrogate for the inner problem output, i.e. we obtain $\tilde{m}_p(\boldsymbol{\xi}) \approx m_p(\mathbf{y}^*(\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v), \boldsymbol{\xi})$. However, for the specific application of solving a max-min problem, we use the fact that the inner problem objective is the same value as the outer problem, thus we solve

$$\boldsymbol{\xi}_M = \underset{\boldsymbol{\xi}}{\operatorname{argmax}}\, \tilde{m}_p(\boldsymbol{\xi}). \tag{25}$$

We use a Krigging model, implemented in the DACE toolbox [10], for the surrogate $\tilde{m}_p$. We train this model with 15 training points. Equation 25 is solved on this surrogate model using MP-AIDEA [2]. This solution is then verified by computing it without the surrogate model and the model is refined with this value. This process of optimizing and refining is iterated 5 times.

This control map restricts the search space to Y^B, defined as

$$Y^B = \left\{ \mathbf{y} \in Y : \mathbf{y} = \mathbf{y}^B(\mathbf{D}_r, \mathbf{D}_v)\ \forall \mathbf{D}_r \in R_r, \mathbf{D}_v \in R_v \right\}, \tag{26}$$

where $Y^B \subset Y^* \subset Y$, and R_r and R_v are box sets. Because this control map does not depend on $\boldsymbol{\xi}$, its size is decoupled from the number of uncertain variables n_ξ. This makes the surrogate modelling and MO optimization process more scalable.

Min-Max Control Map An additional control map is also proposed which defines $\mathbf{z}$ the same way as Y^B, but is formulated as a min-max problem where the constraint is applied on both the minimization and maximization subproblems:

$$\begin{aligned} \mathbf{y}^M(\mathbf{D}_r, \mathbf{D}_v) = &\underset{\mathbf{y}\in\mathbf{Y}}{\operatorname{argmin}} \max_{\boldsymbol{\xi}\in\Xi} m_p(\mathbf{y}, \boldsymbol{\xi}) \\ &s.t. \\ &max_{\boldsymbol{\xi}\in\Xi}\ C(\mathbf{y}, \boldsymbol{\xi}) \leq \epsilon \end{aligned} \tag{27}$$

where the constraint $C(\mathbf{y}, \boldsymbol{\xi})$ is defined as

$$C(\mathbf{y}, \boldsymbol{\xi}) = \sum_r \|\mathbf{r}(X, \mathbf{y}, \boldsymbol{\xi}, (\mathbf{t_T})_r) - \mathbf{r}_r((\mathbf{t_T})_r)\|^2, \tag{28}$$

for flyby constraints, and as

$$C(\mathbf{y}, \boldsymbol{\xi}) = \sum_r \sum_j \left(\mathbf{x}(\mathbf{y}, \boldsymbol{\xi}, (\mathbf{t_T})_r)_j - \mathbf{x}_{rj}((\mathbf{t_T})_r)\right)^2, \tag{29}$$

for the rendezvous constraints, where $\mathbf{x}$ is the equinoctial elements, and the semi-major axis a is in astronomical units.

We solve Eq. (27) using *MacMinMax* [6], a Min-Max solver with strict constraints. Because the constraints cannot be met for all $\boldsymbol{\xi}$, the constraint threshold $\epsilon \in \mathbb{R}$ is found by *MacMinMax* via its iterative constraint relaxation process. This algorithm works by solving outer, inner, and constraint subproblems, for more information consult [6]. Given the simplicity of the outer and constraint subproblems, these are run using MATLAB's *fmincon-sqp* ® with step and constraint tolerances set to 10^{-10} and 10^{-5}, respectively, and a maximum of 5000 function evaluations. The inner subproblem is solved using MP-AIDEA [2].

This control map defines a reduced space,

$$Y^M = \left\{ \mathbf{y} \in Y : \mathbf{y} = \mathbf{y}^M(\mathbf{D}_r, \mathbf{D}_v) \; \forall \mathbf{D}_r \in R_r, \mathbf{D}_v \in R_v \right\}, \tag{30}$$

which does not necessarily satisfy $Y^M \subset Y^*$, unlike Y^B, since there is no expectation that the resulting control satisfies the constraints for any specific values of $\boldsymbol{\xi}, \mathbf{D}_r, \mathbf{D}_v$.

4.2 Threshold Mapping

By defining the search space as an hyper-rectangle over the thresholds, we include a lot of "uninteresting" points where the lower expectation is either zero or one. These points are always dominated, as shown in Fig. 1.

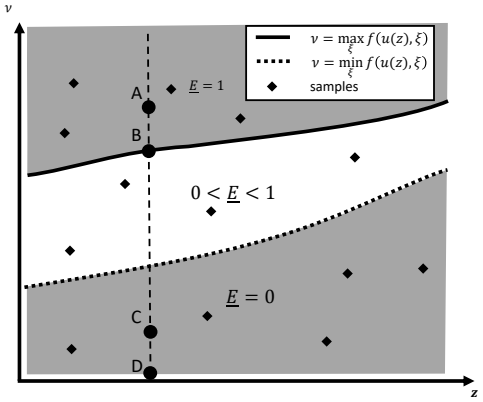

Fig. 1: Abstract diagram representing the search space. When using a hyper-rectangular sample region, as we do, many of the samples fall into the regions where $\underline{\mathrm{E}} = 0$ or $\underline{\mathrm{E}} = 1$, which are shaded in this diagram. Points in these regions are always dominated. For example, point A is dominated by point B, and C by D, since they have the same value of $\underline{\mathrm{E}}$ but with larger ν.

We investigate a threshold mapping strategy intended to focus the search efforts on the "interesting" region. We introduce a proxy threshold variable ν^*,

which maps to the thresholds ν, in such a way that

$$0 < \underline{\mathrm{E}}(\mathbf{y}(\mathbf{z}), \nu(\nu^*, \mathbf{y}(\mathbf{z}))) < 1 \ . \tag{31}$$

Let $h(\mathbf{y}(\mathbf{z}), \xi)$ be our quantity of interest involved in the definition of the lower expectation, as explained in Section 2. Let also $\underline{h}(y(z)) = \min_{\boldsymbol{\xi} \in \Xi} h(\mathbf{y}(\mathbf{z}), \xi)$ and $\overline{h}(\mathbf{y}(\mathbf{z})) = \max_{\boldsymbol{\xi} \in \Xi} h(\mathbf{y}(\mathbf{z}), \xi)$. Our requirement in Eq. (31) implies $\underline{h} < \nu < \overline{h}$, which can be trivially satisfied with the following mapping

$$\nu(\nu^*, \mathbf{y}(\mathbf{z})) = \underline{h} + \left(\overline{h} - \underline{h}\right) \nu^* \ , \quad \nu^* \in [0, 1] \ . \tag{32}$$

This is applied to each threshold of each quantity of interest involved in the optimization.

The surrogate model is trained on $\mathbf{z}$ and ν^*, so that $\tilde{\underline{\mathrm{E}}}(\mathbf{z}, \nu^*) \approx \underline{\mathrm{E}}(\mathbf{z}, \nu(\nu^*, \mathbf{z}))$. Therefore the search space for MACS is $Z \times [0, 1]^{n_s}$, where n_s is the number of metrics. Since we want to minimize ν and not ν^*, this option requires additional surrogate models for each threshold, $\tilde{\nu}(\nu^*, z) \approx \nu(\nu^*, z)$, to model the mapping.

5 Asteroid Tour Test Case

The proposed approach will be applied to a multi fly-by mission to a sequence of four near earth asteroids (NEA), based on the tour proposed in [4], visiting 2006 UJ47, 2007 UV, 2005 YN176, and Ockeghem (Ock).

We consider a case with 10 uncertain variables. These are the hyperbolic exhaust velocity at departure from Earth v_∞, the Thrust at 5 points in the trajectory T_i for $i = 1, \ldots, 5$ and the specific impulse at 4 points in the trajectory $I_{sp,i}$ for $i = 1, \ldots, 4$. Their nominal values, $\boldsymbol{\xi}_0$, are presented in the Table 2. The value of thrust or specific impulse at each point in the trajectory is obtained by linear interpolation.

Table 2: Nominal values for the uncertain variables.

	v_∞ [Km/s]	$I_{sp,i}$ [s]	T_i [N]
$\boldsymbol{\xi}_0$	5.846	3000	.15

We use as an uncertainty space the hyper-rectangle $\Xi = [\boldsymbol{\xi}^L, \boldsymbol{\xi}^U]$, where $\boldsymbol{\xi}^L = 0.9\boldsymbol{\xi}_0$ and $\boldsymbol{\xi}^U = 1.1\boldsymbol{\xi}_0$, and $\boldsymbol{\xi}_0$ is the nominal value of the uncertain parameters, as defined in Table 2. For the control law, as defined in Section 1, we use $n_{LT} = 20$ thrust and coast arcs.

The optimal trajectory for this nominal case is displayed in Figure 2.

The multi-objective problem for this test case consists in maximizing the lower expectation that the mass and the distance to each target are below the respective threshold, and the minimization of these thresholds:

$$\min_{\mathbf{z}, \boldsymbol{\nu}} \begin{bmatrix} -\underline{\mathrm{E}}(m_p(\mathbf{y}(\mathbf{z}), \boldsymbol{\xi}), < \nu_m) \ , \ \nu_m \\ -\underline{\mathrm{E}}(\Delta r_i(\mathbf{y}(\mathbf{z}), \boldsymbol{\xi}) < \nu_i) \ , \ \nu_i \text{ for each target } i \end{bmatrix}, \tag{33}$$

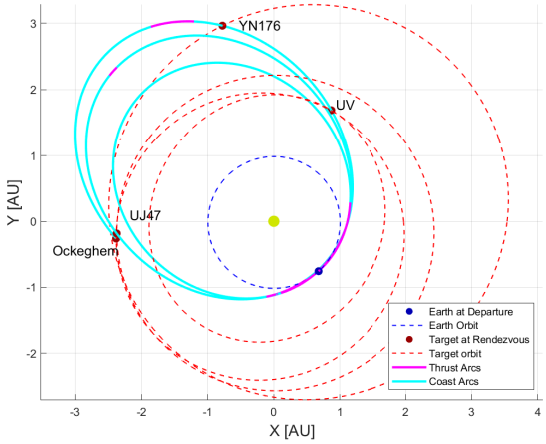

Fig. 2: Plot of a trajectory that visits flies by the asteroids in the asteroid tour. Obtained for the nominal case.

where Δr_i is the distance to target i, and ν_i is the threshold on this quantity.

Because these targets are all fly-by only, in the control maps of Section 4.1, the constraints on velocity are not applied, and the displacements in velocity are not part of the proxy control variables $\mathbf{z}$.

6 Results

We test our methods on the Asteroid tour test case from Section 5, and on the Apophis rendezvous which was presented in a previous paper [9].

6.1 Control Map and Threshold Map

In Fig 3 some multi-objective metrics are shown as a function of the iteration number. We compare methods with and without the threshold map in Section 4.2, and with the four different control maps explained in Section 4.1. The metrics used are the generational distance (GD), the Hausdorff distance (Hausdorff), the modified inverse generational distance (IGD+) [1], and the number of points that are not dominated by the reference front (N_P). The reference front for these metrics is the combination of all non-dominated samples from all methods. These results show that our proposed control map Y^M is an improvement on previous maps. This improvement is not simply due to a reduction in the number of dimensions, since the dimensionality of Y^- ($n_\xi = 10$) is lower than Y^M (12), yet Y^M has better performance. Furthermore, the threshold map consistently improves the results obtained using the Y^B and Y^M control maps.

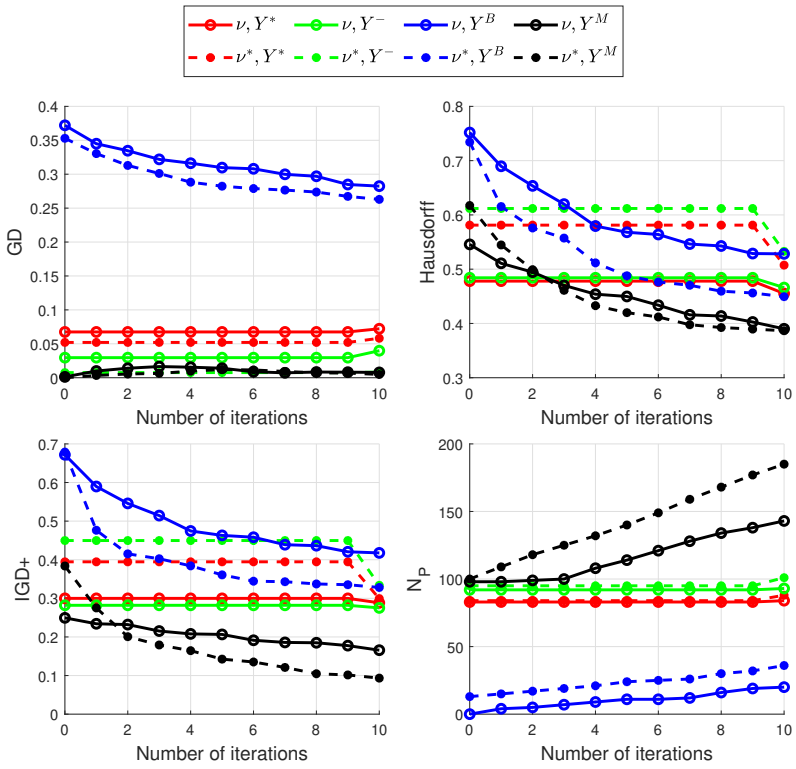

Fig. 3: Plots of 4 MO metrics for the proposed methods to solve the asteroid tour test case. The color of the lines indicates which control map was used, see Section 4.1. Solid lines with circles indicate threshold mapping is not used (ν), while dashed lines with asterisks indicate that it is used (ν^*).

The errors on the surrogates are presented in Fig. 4, which shows that the error on the $\underline{\text{E}}$ surrogate is low (mostly below 5%) for Y^M with threshold mapping, which has the best MO metrics. The opposite is seen for the threshold mapping surrogate model, where its error is among the highest at 17%.

Some of the solutions to the asteroid tour test case, obtained using the Min-Max control map Y^M are shown in Table 3, and Figure 7 shows the trajectory corresponding to the first solution in that table.

The same tests were performed for the Apophis rendezvous test case, presented in [9]. Figure 5 shows the MO metrics, and Fig. 6 plots the surrogate errors. For this test case, in terms of MO metrics the threshold map is still consistently advantageous, but the newly proposed control maps Y^M and Y^B are not. Figure 6 show that the RMSE is kept below 5% for the Y^- and Y_* control maps, except for the $\underline{\text{E}}$ surrogate with Y^*, which is close to 20%. Sometimes the best methods, in terms of MO metrics, do not correspond to the lowest surrogate

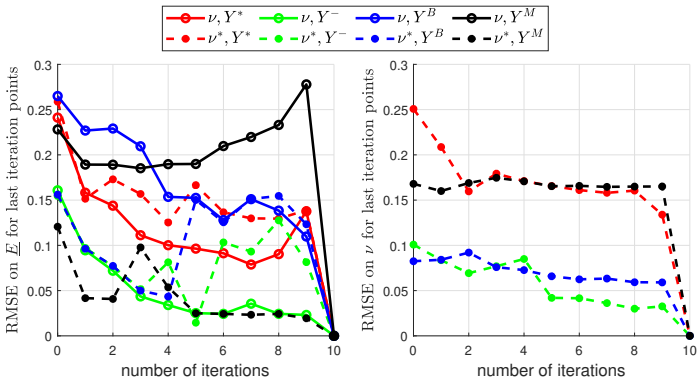

Fig. 4: Root mean square errors (RMSE) for the surrogates of the lower expectation (left) and threshold map (right), for the asteroid tour test case. The RMSE is evaluated for the points that are obtained at the last iteration, and the mean is taken across all metrics. For the threshold map, the error is normalized by dividing by the difference between the maximum and minimum values. The line and marker styles follow the same pattern as in Fig. 3.

RMSE. One explanation is that a better method will obtain a wider variety of points, which are harder to model accurately.

6.2 Lower Expectation

The lower expectation algorithms presented in Section 3.1 are tested on all 200 evaluated points obtained using the Min-Max control map Y^M and threshold mapping for both test cases in Table 4. In this table, the error rate is the percentage of pairs $\mathbf{z}, \boldsymbol{\nu}$ for which each method does not obtain the minimum value of E within $\pm 10^{-6}$, for each metric. This minimum value of E is estimated by taking the best estimate of $\underline{\mathrm{E}}$ from all algorithms being tested. The average number of evaluations of $\mathrm{E}(\mathrm{I}; \mathbf{j})$, N_{E}, is used to measure the efficiency of each algorithm.

These results show the pattern search method with restart and N_k^B is the most accurate. The higher accuracy and lower computational cost associated with this method compared to the Genetic algorithm and the random restarts method validates it, and justifies its use throughout this work.

6.3 Expectation and Sampling Methods

We compare estimating the expectation $E\left(I(\mathbf{y}, \boldsymbol{\nu}, \boldsymbol{\xi}), B_{\mathbf{j}}^{k+1}\right)$ with simple quasi-Monte Carlo (qMC) and with the proposed efficient sampling method (qMC+RS) where we use rejection sampling on a neighbouring sampling on the previous trial point $\mathbf{j}^k$, as in Section 3.2. We measure the accuracy and efficiency of these methods by comparing with qMC with 10^6 samples, and do this for 1000 randomly selected tuples $\mathbf{y}, \boldsymbol{\nu}, k$. The results are in Table 5.

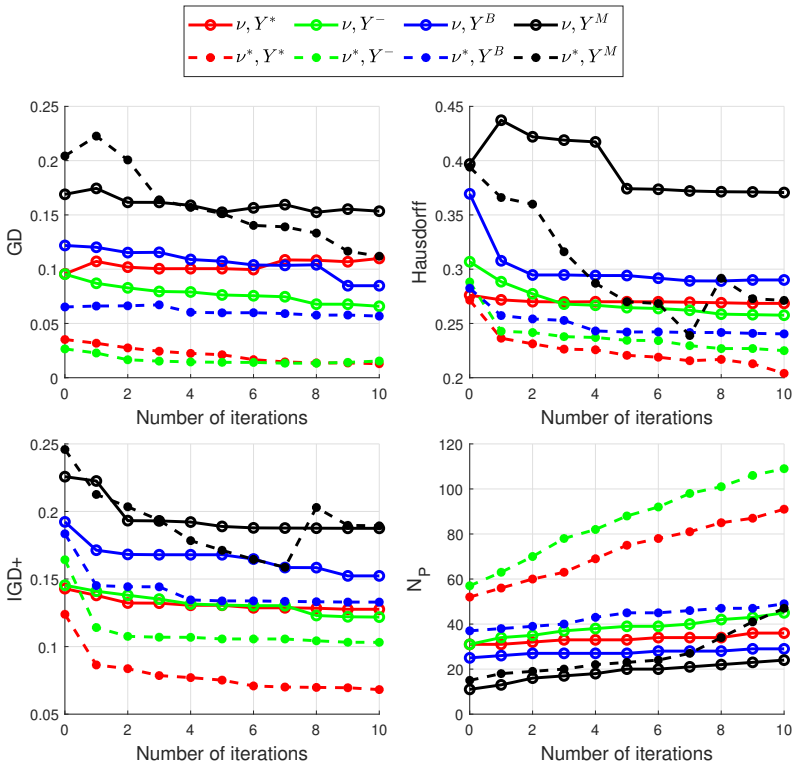

Fig. 5: The 4 MO metrics, as in Fig. 3, for the Apophis rendezvous test case.

The fraction of reused samples η_r was 29%, but the reduction in computational time was only about 18% due to overhead in selecting the reused samples and obtaining the new samples. The accuracy is worsened by only 3.6%. Given that the RMSE of this method is inversely proportional to the square root of the number of samples [8], the number of samples would have to be increased 7.3% to counter this reduction in accuracy, which would still make this method with rejection sampling more efficient.

6.4 Execution Times

Here we show the execution times for a Windows®10 machine with Matlab® 2018b, and Intel®Core™ i7-8700 CPU @ 3.2GHz, and 8GB of RAM. The threshold map took on average 0.5 seconds. The lower expectation took on average 11.5 seconds for the Apophis rendezvous case and about 60 seconds for the asteroid tour test case. The lower expectations were obtained using qMC, without the rejection sampling approach proposed in Section 3.2, since due to time constraints, these scripts were run before this method could be finalized and tested. Each iteration of running MACS took on average 213 seconds. As for the control maps,

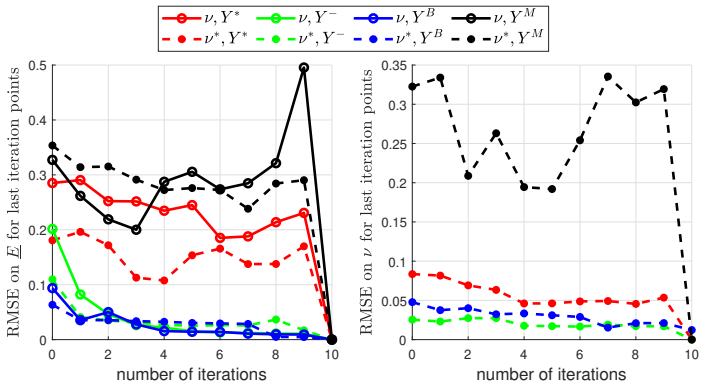

Fig. 6: Root mean square errors (RMSE) for the surrogates of the lower expectation (left) and threshold map (right), for the Apophis Rendezvous test case.

- Y^* and Y^+ take approximately 35 seconds for the asteroid tour case and 8 seconds for the Apophis rendezvous case,
- Y^B takes approximately 205 seconds for the asteroid tour case and 55 seconds for the Apophis rendezvous case,
- Y^M takes approximately 128 seconds for the asteroid tour case and 75 seconds for the Apophis rendezvous case.

The total time for the asteroid tour case with control map Y^M, evaluating 100 initial points plus 10 per each of 10 iterations, is approximately 8 hours.

7 Conclusions

This paper has presented methods to obtain robust solutions for the optimal control problem of finding optimal spacecraft trajectories under low thrust, although it could be applied to a variety of problems. This was tested on a previously presented problem consisting of a rendezvous with an asteroid, as well as an asteroid fly-by tour proposed by other authors. In both of these test cases we considered an engine and launcher with unknown parameters, which evolve over time according to epistemically uncertain parameters.

The uncertainty was modelled using lower expectation, a method which allows taking into account epistemic uncertainty. Multiple methods to calculate it without exponential complexity on the number of uncertain variables were developed and compared, and showed good accuracy and efficiency. Furthermore, a more efficient approach is proposed to the estimate of the expectation during the calculation of the lower expectation.

The newly proposed control methods have considerably higher computational times, however, they use algorithms with many hyper-parameters, that we believe can be further fine-tuned. Furthermore, in the asteroid tour test case, our

Table 3: A selection of 10 solutions out of the 188 returned by our method. Solutions were selected using the same archival algorithm employed by MACS [15]. If x is one of our targets, $\underline{\mathrm{E}}^{(x)}$ is used as short-hand for $\underline{\mathrm{E}}(\Delta_{r_x} < \nu_{\Delta_{r_x}})$ and ν_x is short-hand for $\nu_{\Delta r_x}$ and is in AU. For the propellant mass m_p, $\underline{\mathrm{E}}^{(m_p)} = \underline{\mathrm{E}}(m_p < \nu_m)$, and its threshold ν_m is in Kg.

$\underline{\mathrm{E}}^{(m_p)}$	$\underline{\mathrm{E}}^{\mathrm{UJ47}}$	$\underline{\mathrm{E}}^{\mathrm{UV}}$	$\underline{\mathrm{E}}^{\mathrm{YN176}}$	$\underline{\mathrm{E}}^{\mathrm{Ock}}$	ν_m [Kg]	ν_{UJ47}	ν_{UV}	ν_{YN176}	ν_{Ock}
0.044	0.239	0.232	0.548	0.032	129.742	0.212	0.306	0.327	0.104
0.000	0.333	0.727	0.162	0.657	74.023	0.265	0.396	0.255	0.522
1.000	1.000	0.000	0.000	1.000	89.345	0.463	0.012	0.040	0.700
1.000	0.000	1.000	1.000	1.000	190.913	0.016	0.519	0.491	0.810
1.000	1.000	1.000	1.000	0.000	79.550	0.484	0.539	0.536	0.024
0.000	1.000	0.000	1.000	0.993	259.146	0.488	0.004	0.534	0.648
1.000	0.000	0.000	0.000	0.000	87.848	0.005	0.041	0.062	0.013
0.000	1.000	1.000	0.000	0.000	189.049	0.478	0.507	0.009	0.063
1.000	0.000	0.000	1.000	1.000	92.752	0.039	0.018	0.555	0.694
1.000	1.000	1.000	0.000	1.000	62.334	0.437	0.577	0.052	0.746

Table 4: Comparison of local search variants proposed in Section 3.1 for both test cases, with 10 uncertain variables. In the starting point row, "G" denotes using only the greedy starting point in Eq. 12, "G+S" includes the re-starting procedure and "R5" refers to using 5 random initial starting points and taking the best outcome. The neighbourhoods are defined in Eqs. (10) and (11).

Starting point		G		G + S		R5	Genetic
Neighbourhood		N_k^S	N_k^B	N_k^S	N_k^B	N_k^B	-
Error rate	Asteroid Tour	4.8%	2.1%	3.6%	**0.5%**	0.8%	1.3%
	Apophis	7.0%	3.2%	5.0%	**0.5%**	2.7%	1.3%
N_{E}	Asteroid Tour	320.7	486.8	590.1	1050	15054	2553
	Apophis	118.9	187.3	192.6	384.2	1214	10049

Table 5: Comparison of two methods of estimating the expectation, described in Section 3.2.

	RMSE	η_r	Time [s]
Basic qMC	9.64×10^{-04}	-	45.46
qMC + RS	9.99×10^{-04}	0.2922	37.31

old control maps were not allowing any improvement in the MO metrics at all, in light of which the newly proposed control maps are certainly advantageous.

The threshold map, with its relatively low computational cost and its significant advantage it brings to the MO metrics, is also a promising approach.

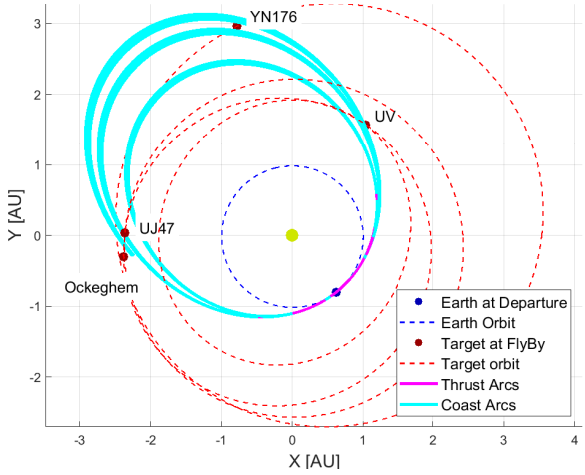

Fig. 7: Trajectories for the control law corresponding to the fifth solution in Table 3 and for 200 different samples of $\boldsymbol{\xi}$, so that the thickness represents the uncertainty in position. The plot is seen perpendicularly to the ecliptic plane.

References

1. Bezerra, L.C.T., López-Ibáñez, M., Stützle, T.: An empirical assessment of the properties of inverted generational distance on multi- and many-objective optimization. In: Trautmann, H., Rudolph, G., Klamroth, K., Schütze, O., Wiecek, M., Jin, Y., Grimme, C. (eds.) Evolutionary Multi-Criterion Optimization, vol. 10173, pp. 31–45. Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-54157-0˙3, http://link.springer.com/10.1007/978-3-319-54157-0˙3, series Title: Lecture Notes in Computer Science
2. Carlo, M.D., Vasile, M., Minisci, E.: Multi-population inflationary differential evolution algorithm with adaptive local restart. In: IEEE Congress on Evolutionary Computation (CEC) (2015)
3. Di Carlo, M., da Graça Marto, S., Vasile, M.: Extended analytical formulae for the perturbed keplerian motion under low-thrust acceleration and orbital perturbations (in press)
4. Di Carlo, M., Vasile, M., Dunlop, J.: Low-thrust tour of the main belt asteroids. Advances in Space Research **62**(8), 2026 – 2045 (2018). https://doi.org/https://doi.org/10.1016/j.asr.2017.12.033, http://www.sciencedirect.com/science/article/pii/S0273117717309213, past, Present and Future of Small Body Science and Exploration
5. Di Carlo, M., Vasile, M., Greco, C., Epenoy, R.: Robust optimisation of low-thrust interplanetary transfers using evidence theory. In: Topputo, F., Sinclair, A., Wilkins, M., Zanetti, R. (eds.) 29th AAS/AIAA Space Flight Mechanics Meeting. vol. 168, pp. 339–358 (2 2019)
6. Filippi, G., Vasile, M.: A memetic approach to the solution of constrained min-max problems. In: 2019 IEEE Congress on Evolutionary Computation (CEC). pp. 506–513. IEEE. https://doi.org/10.1109/CEC.2019.8790124, https://ieeexplore.ieee.org/document/8790124/

7. Greco, C., Di Carlo, M., Vasile, M., Epenoy, R.: An intrusive polynomial algebra multiple shooting approach to the solution of optimal control problems (10 2018), https://www.iac2018.org/, 69th International Astronautical Congress, IAC 2018 ; Conference date: 01-10-2018 Through 05-10-2018
8. Halton, J.H.: Quasi-probability: Why quasi-monte-carlo methods are statistically valid and how their errors can be estimated statistically **11**(3). https://doi.org/10.1515/1569396054495130, https://www.degruyter.com/view/j/mcma.2005.11.issue-3/1569396054495130/1569396054495130.xml
9. Marto, S.G., Vasile, M., Epenoy, R.: Multi-Objective Robust Trajectory Optimisation Under Epistemic Uncertainty and Imprecision. 70th International Astronautical Congress p. 13 (2019)
10. Nielsen, H.B., Lophaven, S.N., Søndergaard, J.: DACE - a matlab kriging toolbox (2002), http://www2.imm.dtu.dk/pubdb/views/publication˙details.php?id=1460
11. Oberkampf, W., Helton, J.: Investigation of Evidence Theory for Engineering Applications. In: 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference. American Institute of Aeronautics and Astronautics (Jun 2012). https://doi.org/10.2514/6.2002-1569, https://arc.aiaa.org/doi/abs/10.2514/6.2002-1569
12. Olympio, J.T.: Designing robust low-thrust interplanetary trajectories subject to one temporary engine failure. In: 20th AAS/AIAA Space Flight Mechanics Meeting (2010)
13. Olympio, J.T., Izzo, D.: Trajectory Optimisation under Uncertainties. Tech. rep. (Oct 2012),
14. Ozaki, N., Campagnola, S., Yam, C.H., Funase, R.: Differential dynamic programming approach for robust-optimal low-thrust trajectory design considering uncertainty. In: 25th International Symposium on Space Flight Dynamics, Munich, Germany. pp. 19–23 (2015)
15. Ricciardi, L.A., Vasile, M.: Improved archiving and search strategies for multi agent collaborative search. In: Minisci, E., Vasile, M., Periaux, J., Gauger, N.R., Giannakoglou, K.C., Quagliarella, D. (eds.) Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences, pp. 435–455. Computational Methods in Applied Sciences, Springer International Publishing. https://doi.org/10.1007/978-3-319-89988-6˙26
16. Sinha, A., Lu, Z., Deb, K., Malo, P.: Bilevel optimization based on iterative approximation of multiple mappings **26**(2), 151–185. https://doi.org/10.1007/s10732-019-09426-9, https://doi.org/10.1007/s10732-019-09426-9
17. Vasile, M., Tardioli, C.: On the Use of Positive Polynomials for the Estimation of Upper and Lower Expectations in Orbital Dynamics. Stardust Final Conference pp. 99–107 (2018). https://doi.org/10.1007/978-3-319-69956-1˙6, https://link.springer.com/chapter/10.1007/978-3-319-69956-1˙6
18. Wachi, A.: Low-Thrust Trajectory Design to Improve Overall Mission Success Probability Incorporating Target Changes in Case of Engine Failures. In: International Symposium on Space Flight Dynamics (2017)
19. Wang, X.: Improving the rejection sampling method in quasi-monte carlo methods **114**(2), 231–246. https://doi.org/10.1016/S0377-0427(99)00194-6, http://www.sciencedirect.com/science/article/pii/S0377042799001946
20. Zuiani, F., Vasile, M.: Extended analytical formulas for the perturbed keplerian motion under a constant control acceleration **121**(3), 275–300. https://doi.org/10.1007/s10569-014-9600-5