# IAC-08 - D1.3
# PRELIMINARY SPACE MISSION DESIGN UNDER UNCERTAINTY

Mr. Nicolas Croisard
University of Glasgow, Glasgow, United Kingdom, n.croisard@aero.gla.ac.uk

Dr. Massimiliano Vasile
University of Glasgow, Glasgow, United Kingdom, mvasile@aero.gla.ac.uk
Mr. Stephen Kemble
Astrium Ltd., Hertfordshire, United Kingdom, stephen.kemble@astrium.eads.net
Dr. Gianmarco Radice
University of Glasgow, Glasgow, United Kingdom, gradice@aero.gla.ac.uk

ABSTRACT

This paper proposes a way to model uncertainties and to introduce them explicitly in the design process of a preliminary space mission. Traditionally, a system margin approach is used in order to take them into account. In this paper, Evidence Theory is proposed to crystallise the inherent uncertainties. The design process is then formulated as an Optimisation Under Uncertainties (OUU). Three techniques are proposed to solve the OUU problem: (a) an evolutionary multi-objective approach, (b) a step technique consisting of maximising the belief for different levels of performance, and (c) a clustering method that firstly identifies feasible regions. The three methods are applied to the BepiColombo mission and their effectiveness at solving the OUU problem are compared.

**Keywords:** Evidence Theory, Optimisation Under Uncertainties, Preliminary Design.

## 1 INTRODUCTION

In the early phase of the design of a space mission, it is generally desirable to investigate as many feasible alternative solutions as possible. At this particular stage, an insufficient consideration for uncertainty would lead to a wrong decision on the feasibility of the mission. Traditionally, a system margin approach is used in order to take into account the inherent uncertainties related to the computation of the system budgets. The reliability of the mission is then independently computed in parallel. An iterative, though integrated, process between the solution design and the reliability assessment should finally converge to an acceptable solution.

This paper proposes a way to model uncertainties and to introduce them explicitly in the design process. The overall system design is then optimised, minimising the impact of uncertainties on the optimal value of the design criteria (e.g. minimum system mass, minimum system power, etc.). Using Evidence Theory, also know as Dempster-Shafer's theory [1][2], both aleatory and epistemic uncertainties, coming from a poor or incomplete knowledge of the design parameters, can be effectively modeled.

The values of uncertain or vague design parameters are expressed by means of intervals with associated probability. Each expert participating in the design, assigns an interval and a probability according to their experience. Ultimately, all the pieces of information associated to each interval are fused together to yield two cumulative values, Belief and Plausibility, that express the confidence range in the optimal design point. In particular the value of Belief expresses the lower probability that the selected design point remains optimal (and feasible) even under uncertainties.

The use of Evidence Theory for robust engineering design was proposed in 2002 by Oberkampf et al. [3] and was more recently applied to the ro-

bust design of space systems and space trajectories [4][5].

In this paper, Evidence Theory is applied to the optimal design of the mission BepiColombo. The main spacecraft subsystems are modeled using Evidence Theory to deal with uncertain parameters. The design process is then formulated as an Optimisation Under Uncertainties (OUU) problem and the Belief is optimised (maximised) together with all the other criteria that define the optimality of the design point.

Three techniques are proposed to solve the OUU problem: (a) an evolutionary multi-objective approach aiming at minimising the effect of uncertainties on the objective function, while optimising the mission goals, (b) a step technique consisting of maximising the belief for different levels of performance using a local optimiser, and (c) a clustering method that identifies, in the space of design and uncertain parameters, the set of points for which the design criteria assume values below a given threshold.

The results in the paper show the effectiveness of the three proposed techniques at solving efficiently the OUU problem.

# 2 UNCERTAINTY MODELING THROUGH EVIDENCE THEORY

The most common way to deal with uncertainty is the probabilistic approach. However, this theory does not suit to represent every type of uncertainties, and unavoidable assumptions may significantly modify the result of the analysis. Thus new theories have been developed. The Evidence Theory is one of them and is currently the most common alternative to probability.

In this section, the different types of uncertainties are exposed, leading to the justification of why probability theory may not be suitable in our application. Finally, the Evidence Theory is presented and proposed as an alternative.

## 2.1 Types of uncertainty

Uncertainties are usually classified in two distinct categories, aleatory and epistemic uncertainty. According to Helton [6], the definition of each type is[1]:

**Aleatory Uncertainty** The type of uncertainty which results from the fact that a system can behave in random ways.

**Epistemic Uncertainty** The type of uncertainty which results from the lack of knowledge about a system and is a property of the analysts performing the analysis.

W.L. Oberkampf considers a third category, **Error**, also called numerical uncertainty, which "*is defined as a recognizable deficiency in any phase or activity of modelling and simulation that is not due to lack of knowledge*" [8]. Such uncertainties are well-known, and a good estimation of the error should be easily available. This point distinguishes errors to epistemic uncertainties.

Aleatory uncertainties are due to the random nature of input data while epistemic ones are generally linked to incomplete modelling of the physical system, the boundary conditions, unexpected failure modes, etc. In the particular case of preliminary space mission design, analysts face both types of uncertainty. For example, the initial velocity of the spacecraft, the gravity model or the solar radiation presents aleatory uncertainties. However, most of the parameters of the spacecraft subsystems are first assessed by a group of experts, expressing their opinion on ranges of values. The uncertainty associated to those parameters is therefore epistemic.

The classical way to treat uncertainty is through probability theory. A probability density function is well suited to mathematically model aleatory uncertainties, as far as enough data (experimental for instance) are available [8]. Even though, the analyst still has to assume the distribution function and estimate its parameters. Moreover, Bae, Grandhi and Canfield [9] pointed out that aleatory uncertainty could be in fact epistemic uncertainty when "*insufficient data are available to construct a probability distribution*".

Probability fails to represent epistemic uncertainties because there is no reason to prefer one distribution function over another [3]. When uncertainties are express by means of intervals, based on experts opinion or rare experimental data, as it is the case in space mission design, this representation becomes even more questionable.

A few modern theories exist to better represent epistemic uncertainties, without the need to make

[1] This comes actually from [7], citing Helton

additional assumption. The most common one is the Evidence Theory, which we propose to use in the framework of preliminary space mission design.

## 2.2 Overview of the Evidence Theory

The Evidence Theory has been developed by Shafer [2] based on Dempster's original work [1] and has been proven to model adequately both types of uncertainty. The theory does not request additional assumptions when the available information is poor or incomplete. For instance, evidence on the event $\{A \ or \ B\}$ does not imply/require information on both events $\{A\}$ and $\{B\}$. Similarly, the knowledge of an event does not imply knowledge of its opposite (for the probability theory, $P(\overline{A}) = 1 - P(A)$). Moreover, it is common to have the information from diverse sources, such as different experts or experiments. There is no reason, a priori, to choose one source from the others. The Evidence Theory offers the possibility to combine information from different sources. A number of rules have been developed to combine evidence depending on the context or how much they conflict. K. Sentz and S. Ferson have collecting a wide list in [7].

### 2.2.1 Frame of discernment $\mathcal{U}$ and Basic Probability Assignment

The frame of discernment $\mathcal{U}$, also know as the universal set, is "*a set of mutually exclusive elementary propositions*" [9]. In most engineering applications of the Evidence Theory, experts express their belief of an uncertain parameter $u$ being within various intervals. $u \in [a, b]$ is then in this case an elementary proposition, thus an element of $\mathcal{U}$. The level of confidence an expert has on an elementary proposition is quantified using the Basic Probability Assignment (BPA) that satisfies the three following axioms:

1. $m(E) \geq 0, \forall E \in \mathcal{U}$
2. $m(\varnothing) = 0$
3. $\sum_{E \in \mathcal{U}} m(E) = 1$

An element of $\mathcal{U}$ that has a non-zero BPA is named a focal element ($FE$).

When more than one parameter are considered uncertain (e.g. $u_1$ and $u_2$), the frame of discernment is composed of all the cartesian products of the intervals. The BPA of a given cartesian product is then the product of the BPA of each interval:

$$m\big((u_1, u_2) \in [a_1, b_1] \times [a_2, b_2]\big) = m\big(u_1 \in [a_1, b_1]\big) * m\big(u_2 \in [a_2, b_2]\big) \quad (1)$$

### 2.2.2 Belief and Plausibility functions

The Belief ($Bel$) and Plausibility ($Pl$) functions represent the lower and upper bounds of the uncertainty quantification. Indeed, as we explained earlier, the Evidence Theory does not request additional assumption when there is a lack of information. To come out with a single uncertainty quantification value, the probability theory for example fills incomplete information or ignores it. It is more reasonable to present the bounds than a singular value.

The belief and plausibility function are defined as follow:

$$Bel(A) = \sum_{\substack{FE \subset A \\ FE \in \mathcal{U}}} m(FE) \quad (2)$$

$$Pl(A) = \sum_{\substack{FE \cap A \neq \varnothing \\ FE \in \mathcal{U}}} m(FE) \quad (3)$$

Thus, propositions intercepting **A** but not included in **A** are considered in the $Pl$ but not in the $Bel$. Let us have a look at an example. The figure 1 represents a BPA structure of two uncertain parameters $u_1$ and $u_2$. $u_1$ can take its value within the four intervals $[a_1, b_1]$, $[b_1, c_1]$, $[c_1, d_1]$ and $[d_1, e_1]$ while the domain of $u_2$ is divided in three parts $[a_2, b_2]$, $[b_2, c_2]$ and $[c_2, d_2]$. Thus there is a total of twelve focal elements $FE_1, \ldots, FE_{12}$. Let us define the proposition **A** as the area within the dash curve $\mathscr{C}$. Only the focal elements $FE_1$, $FE_6$ and $FE_{12}$ (gray in the figure) are entirely surrounded by $\mathscr{C}$. In addition, $FE_2$, $FE_3$, $FE_5$, $FE_7$, $FE_9$ and $FE_{11}$ are partly inside $\mathscr{C}$ (doted in the figure), therefore only partially implying the proposition **A**. Therefore the belief and plausibility of **A** are:

$$Bel(\mathbf{A}) = m(FE_1) + m(FE_6) + m(FE_{12})$$

$$\begin{aligned} Pl(\mathbf{A}) = \ & m(FE_1) + m(FE_2) + m(FE_3) \\ & + m(FE_5) + m(FE_6) + m(FE_7) \\ & + m(FE_9) + m(FE_{11}) + m(FE_{12}) \end{aligned}$$

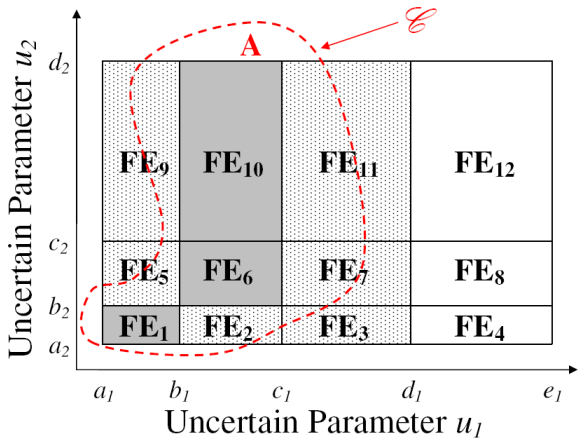


Figure 1: Belief and Plausibility of proposition A in a given $BPA$ structure of two uncertain parameters.

If the pair $(u_1, u_2)$ takes their value within $[b_1, c_1] \times [c_2, d_2]$, it fulfils the proposition **A**. However, if it is inside $[c_1, d_1] \times [b_2, c_2]$, it may verify **A** but not for sure. Therefore, the belief represents our confidence in **A** guaranteed to be true while the plausibility our confidence of **A** being possibly fulfilled.

An important and meaningful relation between belief and plausibility functions comes directly from the fact that all basic assignments must sum to 1.

$$Pl(A) + Bel(\overline{A}) = 1$$

where $(\overline{A})$ represents the complement of $A$. This means that $Pl$ considers the uncertainty, while $Bel$ does not (cf. figure 2).

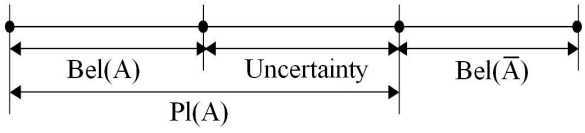


Figure 2: Interpretation of the relation between Belief, Plausibility and uncertainty

## 2.3 Generic Problem of Optimisation Under Uncertainty

Let's consider a function $f$ characterising a system to be optimised. $f$ is function from $\mathbb{R}^{\gtrdot+\ltimes}$ to $\mathbb{R}$ of some uncertain parameters $\mathbf{u} = [u_1, u_2, \ldots, u_m]$ and design variables $\mathbf{d} = [d_1, d_2, \ldots, d_n]$. A BPA is associated to the frame of discernment $\mathcal{U}$ of the uncertain parameters $\mathbf{u}$. The design variables are well know and can be adjusted at will by the designer within a domain $\mathcal{D}$ to optimise the system. For a given constant $\nu$, named the threshold, the generic problem of optimisation under uncertainty (OUU) can be defined as follow:

$$\max_{\mathbf{d}\in\mathcal{D}} \; \underset{\mathcal{U}}{Bel}\big(f(\mathbf{d},\mathbf{u}) < \nu\big) \tag{4}$$

The plausibility could be used instead of the belief, or the proposition could be replaced by $f > \nu$ depending on the system and goal of the optimisation. The remaining of this paper however would still be applicable.
To better understand how the belief of $f$ being lower than the threshold $\nu$ is computed, let us define for a given design vector $\mathbf{d}$:

$$\begin{aligned}\mathcal{U}_\nu &= \{FE \in \mathcal{U} | \forall \mathbf{u} \in FE, f(\mathbf{d},\mathbf{u}) < \nu\} \\ &= \left\{FE \in \mathcal{U} | \max_{\mathbf{u}\in FE} (f(\mathbf{d},\mathbf{u})) < \nu\right\}\end{aligned} \tag{5}$$

Thus using Eq. (2) we have:

$$\underset{\mathcal{U}}{Bel}\big(f(\mathbf{d},\mathbf{u}) < \nu\big) = \sum_{FE\in\mathcal{U}_\nu} m(FE) \tag{6}$$

¿From the above definitions, it clearly appears that the computational time required to solve such an optimisation problem can becomes quickly prohibitive. Indeed, to identify the focal elements of $\mathcal{U}_\nu$, the maximum of $f$ over every focal element of $\mathcal{U}$ must be computed and compared to $\nu$. In the event that the system function is convex, this maximum lies at one of the vertices of the focal element. Otherwise, a classic optimisation problem has to be solved over every focal element. These operations have to be repeated for each new design vector during the optimisation process.
Finally, designers are usually interested in the variation of the optimal belief with the threshold. Indeed, it may be relevant to take a little more risk (a slightly lower value of the belief) if the performance gain is significant. Therefore, the practical problem is a bi-objective optimisation problem defined as follow:

$$\begin{cases} \max\limits_{\nu\in\mathbb{R},\mathbf{d}\in\mathcal{D}} & \underset{\mathcal{U}}{Bel}\Big(f(\mathbf{d},\mathbf{u}) < \nu\Big) \\ \min\limits_{\nu\in\mathbb{R},\mathbf{d}\in\mathcal{D}} & \nu \end{cases} \tag{7}$$

The typical shape of the Pareto front of problem 7 is given in figure 3. As the belief function, it has a stair shape.

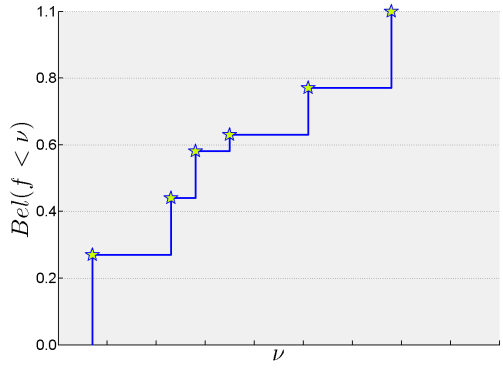


Figure 3: Typical shape of the Pareto front of the optimisation under uncertainty problem. The stars represent the Pareto optimal points. (This is also representative of the shape of the belief function for a specific design vector)

# 3 THREE APPROACHES TO SOLVE THE OUU PROBLEM

The computational cost of the belief function and its step nature makes its maximisation difficult. In this section three different approaches to solve the problem defined in Eq. (7) are presented.

## 3.1 Direct Approach: Using a multi-objective optimiser

The most natural way to solve problem (7) is to use a multi-objective optimiser (MOO) [4][5]. It allows to investigate globally the design space and provides optimal designs for various level of the belief. This is desirable as it would enable the decision makers to do a trade-off among them.
However, considering the threshold $\nu$ as a standard objective creates a problem: for a given design vector the $Bel - \nu$ plot presents a whole set of points, each one corresponding to different values of $\nu$ (the stars of figure 3). If the classical Pareto dominance index:

$$I_i = \Big|\{j \mid Bel(\mathbf{d}_j) > Bel(\mathbf{d}_i) \wedge \nu_j < \nu_i \quad j = 1, \ldots, n_{pop} \wedge j \neq i\}\Big| \quad (8)$$

is used to define the Pareto optimality of a design vector $\mathbf{d}_i$, where $|.|$ denotes the cardinality of a set, then the optimiser cannot evaluate correctly the local Pareto optimality of a point on the $Bel - \nu$ plane since for each $\mathbf{d}$ there is a whole curve of points in the $Bel - \nu$ plane. Moreover, the same Pareto front could correspond to different design points. As we will see in the reminder of this paper, all these difficulties can lead to a significant overhead in the computational effort required to a MOO algorithm to find the global Pareto front.

For the tests in this paper we will use a population-based optimiser called `EPIC` with a standard formulation of the dominance index (8) ($n_{pop}$ is the number of agents). It is based on a an algorithm that combines a deterministic domain decomposition technique and a stochastic-based multi-agent collaborative search [10].

In order to address some of the problems related to the direct application of an MOO algorithm to the solution of (7) we propose two other approaches to the problem. One treats $\nu$ separately maximising $Bel$ for given values of $\nu$, the other identifies in the space of the design and uncertain variables the regions subsets that satisfy the condition $f < \nu$ and therefore that can contribute to the value of the $Bel$ function.

## 3.2 Step Method

To be able to compute the Belief, a threshold has to be set in some way. The Step Method is very straight forward as it computes the optimum belief for discrete value of the threshold. An initial value is initially fixed by the user such that it exists a design vector $\mathbf{d}$ for which the Belief equals 1. Then the threshold is decreased (or increased, depending of the problem) and a local optimiser is used to find the optimal design starting from the previous one. This process stops when the belief reaches the minimum possible value, i.e. 0. The algorithm is detailed in algorithm 1.

Due to the non derivative nature of the belief function, a gradient-based optimiser is not applicable. Therefore, a derivative-free algorithm (the MatLab `fminsearch` algorithm) has been used. Additionally, the use of the previous optimal design to start a new step helps the local optimiser to converge quickly but make it fails to identify a completely different design that may be optimal. To overcome this issue, an other starting design point (or multiple starting design points) could be chosen, or even a global optimiser used. However, the associated extra computational time would be significant.

**Algorithm 1**: Algorithm of the step method

**Input**: $\nu$, *step*

**Output**: Matrix *Out* where each row corresponds to a step. The i$^{\text{th}}$ row of *Out* is composed of the value of the threshold, the optimum design vector and the maximum belief found at the i$^{\text{th}}$ step.

```
/* This first while loop should be
   avoided by setting a high enough
   threshold                          */
```
**while** $\nexists \mathbf{d} \in \mathcal{D} \mid Bel^*(\nu) = 1$ **do**
  $\nu \leftarrow \nu + step$
**endw**
$Bel_{max} \leftarrow 1$
**while** $Bel_{max} > 0$ **do**
```
  /* Update the threshold            */
```
  $\nu \leftarrow \nu - step$
```
  /* Optimisation of the belief for
     the given threshold             */
```
  $[Bel_{opt}, \mathbf{d}_{opt}] \leftarrow \max_{\mathbf{d}\in\mathcal{D}} \underset{\mathcal{U}}{Bel}\left(f(\mathbf{d},\mathbf{u}) < \nu\right)$
```
  /* Add a line at the end of the
     output matrix and save the
     results                         */
```
  $Out(end+1,:) = [\nu, Bel_{opt}, \mathbf{d}_{opt}]$
```
  /* Update the optimum belief
     variable                        */
```
  $Bel_{max} \leftarrow Bel_{opt}$
**endw**

## 3.3 Cluster Approximation

The cluster approximation is an indirect way to solve problem (7). The main idea here is to identify within the complete domain (i.e. the product of the uncertain parameters domain and the design domain) the set $\mathcal{S}_\nu$ of subdomains where the system function verify the proposition $f < \nu$. An approximation $\tilde{Bel}$ of the Belief (resp. Plausibility $\tilde{Pl}$) of the proposition $f < \nu$ can then be cheaply computed by adding the mass of the focal elements included (resp. intersecting) any element of $\mathcal{S}_\nu$.

$$\tilde{Bel}\left(f < \nu\right) = \sum_{\substack{FE \subset s \\ s \in \mathcal{S}_\nu}} m\left(FE\right) \tag{9}$$

$$\tilde{Pl}\left(f < \nu\right) = \sum_{\substack{FE \cap s \neq 0 \\ s \in \mathcal{S}_\nu}} m\left(FE\right) \tag{10}$$

The figure 4 illustrates the proposed method. In this example, there are only three focal elements $FE_1$, $FE_2$ and $FE_3$. The set of subdomains where the system function verify the proposition is:

$$\mathcal{S}_\nu = \{s_1, s_2, s_3\}$$

Two different design points $\mathbf{d}_1$ and $\mathbf{d}_2$ are represented. For $\mathbf{d}_1$, the approximations of the belief and plausibility are:

$$\begin{aligned} \tilde{Bel}_{\mathbf{d}_1}\left(f < \nu\right) &= m\left(FE_1\right) \\ \tilde{Pl}_{\mathbf{d}_1}\left(f < \nu\right) &= m\left(FE_1\right) + m\left(FE_2\right) \end{aligned}$$

For $\mathbf{d}_2$, the approximations of the belief and plausibility are:

$$\begin{aligned} \tilde{Bel}_{\mathbf{d}_2}\left(f < \nu\right) &= m\left(FE_2\right) + m\left(FE_3\right) \\ \tilde{Pl}_{\mathbf{d}_2}\left(f < \nu\right) &= m\left(FE_2\right) + m\left(FE_3\right) \end{aligned}$$

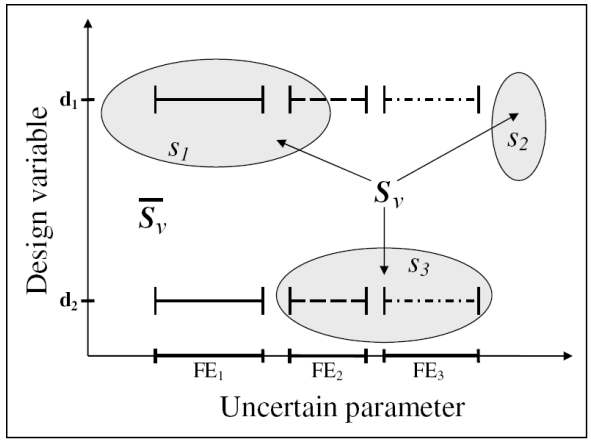


Figure 4: Illustration of the cluster method with 3 focal elements $\text{FE}_1$, $\text{FE}_2$ and $\text{FE}_3$. The proposition $f < \nu$ is true only within the subdomains $s_1$, $s_2$ and $s_3$. Two examples of design point $\mathbf{d}_1$ and $\mathbf{d}_2$ are given.

**Algorithm** To compute the approximation of the *Bel* function, the set $\mathcal{S}_\nu$ of subdomains is computed for increasing values of the threshold until a belief of 1 is found. At each step, sample points verifying the proposition $f(\mathbf{d},\mathbf{u}) < \nu$ are identified, then classified in clusters. The points of a given cluster defines one subdomain $s_i$ of $\mathcal{S}_\nu$. Then, the design maximising the approximation of the belief $\tilde{Bel}$ is selected. The algorithm used here is described in Algorithm 2.

To speed up the computation, Axis-Aligned Box (AAB) are used. Each subdomain $s_i$ is associated with its outer AAB (called also the Axis-Aligned Boundary Box) and an inner AAB. If $s_i$ is defined

by the set of points of $\mathbb{R}^{m+n}$ $(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_p})$, then its axis-aligned boundary box $oAAB(s_i)$ is defined as:

$$oAAB(s_i) = \big\{\mathbf{y} \in \mathbb{R}^{m+n} \mid \forall k, 1 \leq k \leq (m+n), \min_{1\leq j\leq l} x_j(k) \leq y(k) \leq \max_{1\leq j\leq l} x_j(k)\big\} \quad (11)$$

The inner AAB is an axis-aligned box that is contained within the subdomain $s_i$. As opposite to the outer AAB, the definition of the inner AAB is not unique. It has been chosen here to centre the inner AAB on the barycentre of the sample points defining $s_i$ and to maximise its relative size such that it remains within $s_i$.

The idea behind the inner and outer AABs is that it is extremely cheap to check if a focal element is outside or inside an AAB. The focal elements that are outside the outer AAB are guaranteed not to be within $\mathcal{U}_\nu$ and the one inside the inner AAB are guaranteed to be within $\mathcal{U}_\nu$. Once this selection process done, only the focal elements that do not enter in any of those categories need to be checked to compute $\tilde{Bel}$.

In order to identify if any of the remaining focal elements fulfils the proposition $f(\mathbf{d}, \mathbf{u}) < \nu, \forall \mathbf{u} \in FE$, one only need to check if its vertices are within the same subdomain $s_i$. In our implementation $s_i$ is the convex hull of the sample points of the i$^{\text{th}}$ cluster. If $\mathbf{v}$ is a point of $\mathbb{R}^p$, we have:

$$\mathbf{v} \in s_i \Longleftrightarrow \exists \boldsymbol{\lambda} \in \left(\mathbb{R}^+\right)^p \mid \quad \mathbf{v} = \sum_{k=1}^{p} \lambda(k) * \mathbf{x_k} \quad \text{and} \quad \sum_{k=1}^{p} \lambda(k) = 1 \quad (12)$$

Thus the phase 1 of the revised simplex method used to find a feasible solution to a linear programming problem has been implemented in order to determine whether or not such a vector $\boldsymbol{\lambda}$ exits [11][12].

It is important to highlight that in this method, no assumption are made on the convexity of the system function $f$. Only the subdomains $s_i$ are considered as convex which in the practical application related to space design appears reasonable. If that would not be the case, another way to define the subdomains $s_i$ could be used. Another advantage of this method is that it shall identify all the local optimum design regions and thus identifying various classes of interesting design (as in

**Algorithm 2**: Algorithm of the cluster approximation method

**Input** : $\nu$, *step*
`/* Fix a low value for threshold` $\nu$ `*/`
**Output**: Matrix *Out* where each row corresponds to a step. The i$^{\text{th}}$ row of *Out* is composed of the value of the threshold, the optimum design vector and the maximum approximated belief found at the i$^{\text{th}}$ step.
$X = \{\}$; $X_{new} = \{\}$
`/* Initial sample points */`
$X \leftarrow \{[\mathbf{d}, \mathbf{u}] \mid f(\mathbf{d}, \mathbf{u}) \leq \nu\}$
`/* Initialise` $\tilde{Bel}_{max}$ `*/`
$\tilde{Bel}_{max} \leftarrow 0$
**while** $\tilde{Bel}_{max} < 1$ **do**
  `/* Update the threshold */`
  $\nu \leftarrow \nu - step$
  `/* New sampling points */`
  $X_{new} \leftarrow \{[\mathbf{d}, \mathbf{u}] | (\nu - step) < f(\mathbf{d}, \mathbf{u}) \leq \nu\}$
  `/* Update the set of valid sampled point */`
  $X \leftarrow \{X, X_{new}\}$
  `/* Identify the valid subdomains */`
  Partition in *clusters* the sample points $X$
  **foreach** *cluster* **do**
    Compute the associated convex hull
    Compute the oAAB and an iAAB
  **endfch**
  `/* Find the design point giving the highest` $\tilde{Bel}$ `*/`
  $[\tilde{Bel}_{opt}, \mathbf{d}_{opt}] \leftarrow \max_{\mathbf{d}\in\mathcal{D}} \tilde{Bel}(\mathbf{d}, \nu)$
  `/* Add a line at the end of the output matrix and save the results */`
  $Out(end+1, :) = [\nu, \tilde{Bel}_{opt}, \mathbf{d}_{opt}]$
  `/* Update the optimum belief variable */`
  $\tilde{Bel}_{max} \leftarrow \tilde{Bel}_{opt}$
**endw**

the multi-objective method). Finally the global optimum is likely to be found using a simple local optimiser, starting for instance from the barycentre of each cluster.

# 4 EXAMPLE OF APPLICATION

In this section, we will present the results of the three previously described approaches applied to the preliminary design of the BepiColombo mission. The objective is to minimise the wet mass of the spacecraft (for the low-thrust part of the mission) considering uncertainties on a few parameters. The first part of this section present the mass modelling of the spacecraft, i.e. the system function $f$.

## 4.1 Wet Mass modelling

The mass model presented here is a generic one used for preliminary system mass assessment of a Solar Electric Propulsion (SEP) mission. It enables the mass dependence on thrust profile and specific impulse to be evaluated. This model has been kindly provided by S. Kemble and is mainly based on [13].

The total SEP related mass is given by the following equation:

$$m_{\underset{wet}{SEP}} = m_{tank} + m_{array} + m_{rad} + m_{harness} + m_{PPU} + m_{thrusters} + m_{xenon} \quad (13)$$

where $m_{tank}$ is the mass of tanks and associated equipment, $m_{array}$ is the mass of the solar arrays, $m_{rad}$ is the radiator and associated structural mass, $m_{harness}$ is the mass of harness equipment, $m_{PPU}$ is the mass of power processing subsystem, $m_{thrusters}$ is the mass of the thrusters and structural related mass and $m_{xenon}$ is the mass of Xenon required to perform the low thrust transfer. The expressions of all these quantities are given in the following subsections.

### 4.1.1 Mass of propellant

The mass of xenon is estimated from the $\Delta V$ budget using the rocket equation.

$$m_{xenon} = m_{TLO}\left(1 - e^{-\frac{\Delta V}{\overline{I_{SP}} * g_0}}\right) \quad (14)$$

where $m_{TLO}$ is the trans lunar orbit mass, i.e. the wet mass of the spacecraft just after the Earth-Moon system escape (*specific to this mission*, $m_{TLO} = 2400$ kg), $g_0$ is the gravitational acceleration ($g_0 = 9.80665$ m.s$^{-2}$), $\Delta V$ is the delta V budget for the SEP transfer from the Earth-Moon system escape to the Mercury capture (in m.s$^{-1}$) and $\overline{I_{SP}}$ is the mean specific impulse of the SEP transfer, given in seconds by Eq. (15).

$$\overline{I_{SP}} = \frac{4670}{4720} * I_{\underset{max\ T}{SP}} \quad (15)$$

In Eq.(15), $I_{\underset{max\ T}{SP}}$ is the specific impulse at maximum thrust (in seconds).

**Delta V budget** The delta V budget is composed of the deep space $\Delta V$ (cf. below), the $\Delta V$ for second Lunar Gravity Assist (40 m.s$^{-1}$), the $\Delta V$ for SAA control (100 m.s$^{-1}$), the $\Delta V$ for flyby navigation (260 m.s$^{-1}$), the $\Delta V$ for other navigation (280 m.s$^{-1}$) and the contingency (+5% of the deep space $\Delta V$).

The deep space $\Delta V$ is a quantity essential to any optimisation of spacecraft design. Indeed, it has a direct impact on the propellant mass (cf. Eq.(14)) and the tank mass (cf. Eq.(16)). In the frame of the BepiColombo test case, computing this value is computationally expensive and only partly automatic. Therefore, it is not feasible to consider it within the model as it is. In order to overcome this issue, a surrogate model has been built based on 180 different transfers priorly computed using the *EADS-Astrium Stevenage* in-house software *OrbitOptimisationFacility* [13] for various values of $P_{1AU}$ the power to be generated by the solar arrays at 1 Astronomical Unit (AU) and $T_{max}$ the maximum thrust. Moreover, the surrogate reduces significantly the computational time but at the expense of accuracy. For this study, Kriging has been selected via the DACE package [14], with a first order polynomial regression model and an exponential correlation model (cf. figure 5).

### 4.1.2 Tank mass

$$m_{tank} = \sigma_{tank} * m_{xenon} \quad (16)$$

where $\sigma_{tank}$ is the specific ratio of the tank subsystem ($\sigma_{tank} = 11\%$).

### 4.1.3 Solar arrays

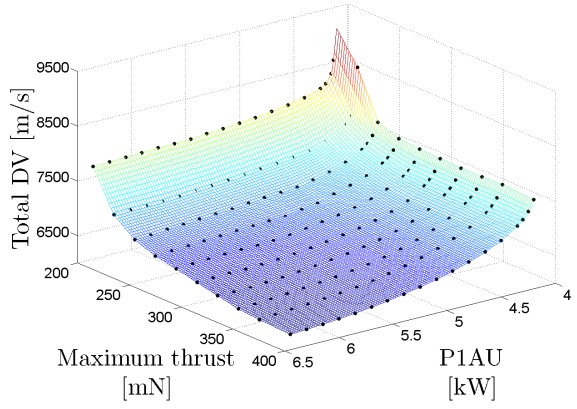


Figure 5: Kriging surrogate of the deep space $\Delta V$ for the low thrust mission of BepiColombo.

**Array area**

$$A = \frac{P_{1AU}}{\eta_p * G_s} * \kappa_A \tag{17}$$

where $\eta_p$ is the power conversion efficiency ($\eta_p = 0.22751$), $G_s$ is the solar constant ($G_s = 1767$ W.m$^{-2}$) and $\kappa_A$ is the area margin for the solar arrays ($\kappa_A = 1.2$).

**Array mass**

$$m_{array} = \left( A * \rho_{SA} + m0_{\,array} \right) * \kappa_{SA} \tag{18}$$

where $\rho_{SA}$ is the specific ratio mass/area of the solar arrays ($\rho_{SA} = 2.89$ kg.m$^{-2}$), $m0_{\,array}$ is the inevitable structural mass of the solar arrays and $\kappa_{SA}$ is the mass margin for the solar arrays ($\kappa_{SA} = 1.1$).

#### 4.1.4 Radiator and associated structural mass

This is sized based on the maximum power, thus at the shortest distance to the Sun. In the case of BepiColombo mission, this is at the perihelion of Mercury's orbit, i.e. 0.3 AU.

**Maximum power** $P_{max}$ is the total power used by the thrusters at maximum thrust, i.e. at 0.3AU. It is calculated using a system of two equations linking the power used by the thrusters, the thrust, the specific impulse and the voltage. The power is a linear function of the thrust and the square root of the voltage. The specific impulse on the other hand is a second order polynomial of the trust with coefficient linear of the square root of the voltage. The voltage is first computed using Eq. (19). $P_{max}$ is then calculated using Eq. (20) with $T = T_{max}$ and $I_{SP} = I_{SP}{}_{max\ T}$.

$$I_{SP} = b_2 T^2 + b_1 T + b_0 \tag{19}$$
$$P = c * (a_1 T + a_0) \tag{20}$$

where $a_1$, $a_0$, $b_2$, $b_1$ and $b_0$ are linear function of $\sqrt{V}$. $V$ is the voltage in volt and $c$ a constant.

**Dissipated power at Max power**

$$P_{dis} = \delta_p P_{max} + Q \tag{21}$$

where $\delta_p$ is the percentage of the maximal power that is wasted ($\delta_p = 0.15$), $Q$ is the heat to be dissipated at the perihelion of Mercury's orbit and $P_{max}$ is the total power used by the thrusters at maximum thrust, i.e. at 0.3AU, calculated for $T_{max}$ and $I_{SP}{}_{max\ T}$.

**Radiator and associated structural mass** Two different types of radiator can be envisaged for the BepiColombo mission. The choice depends on the value of the dissipated power (cf. Eq.(21)) being above or bellow a given threshold $P_{dis}{}_{lim}$. The mass of the radiator and its associated structure is calculated using the following equation:

$$m_{rad} = \begin{cases} \left( c_0 + c_1 \frac{P_{dis}}{P_{dis}{}_{lim}} \right) * \kappa_{rad} \\ \text{if } P_{dis} < P_{dis}{}_{lim}, \\ \left( c_2 + c_3 \frac{P_{dis}}{P_{dis}{}_{lim}} + c_4 \left( \frac{P_{dis}}{P_{dis}{}_{lim}} \right)^2 \right) * \kappa_{rad} \\ \text{otherwise.} \end{cases} \tag{22}$$

where $c_0$, $c_1$, $c_2$, $c_3$ and $c_4$ are constants and $\kappa_{rad}$ is the mass margin for the radiator ($\kappa_{rad} = 1.15$).

#### 4.1.5 Harness

The harness mass is given by the following equation:

$$m_{harness} = m0_{\,harness} + \rho_{harness} P_{max} \kappa_{harness} \tag{23}$$

where $m0_{\,harness}$ is the inevitable mass of the harness subsystem, $\rho_{harness}$ is the specific ratio mass/power of the harness subsystem ($\rho_{harness} = 1.3763 \cdot 10^{-3}$ kg.W$^{-1}$) and $\kappa_{harness}$ is the mass margin for the harness subsystem ($\kappa_{harness} = 1.2$).

#### 4.1.6 Power Processing Unit

The mission of BepiColombo is designed with 4 power processing unit (PPU). The mass of each of them is estimated using an equation linear with the maximum power $P_{max}$ (cf. Eq.(20)) and the square of the mean specific impulse (cf. Eq.(15)).

#### 4.1.7 Thrusters mass

$$m_{thrusters} = m_{0}_{thrusters} + n_{thruster} m_{nominal}_{thrusters} \quad (24)$$

where $m_{0}_{thrusters}$ is the inevitable mass of the thrusters subsystem, $m_{nominal}_{thrusters}$ is the nominal mass of one thruster and $n_{thruster}$ is the number of thrusters installed aboard the spacecraft ($n_{thruster} = 2$).

#### 4.1.8 Remarks

The simple model presented here enables to estimate the mass of the main subsystems of a low thrust spacecraft with only three inputs:

- $P_{1AU}$: the power to be generated by the solar arrays at 1AU
- $T_{max}$: the maximum thrust
- $I_{SP}_{max\ T}$: the specific impulse at maximum thrust

Moreover, margins are conventionally used to take into account uncertainties on the modelling. These margins are summarised in table 1.

| Margins | Value | Subsystem |
|---|---|---|
| $\Delta V$ | +5% | $\Delta V$ contingency |
| $\kappa_A$ | 1.20 | Area of the solar arrays |
| $\kappa_{SA}$ | 1.10 | Mass of the solar arrays |
| $\kappa_{rad}$ | 1.15 | Mass of the radiator |
| $\kappa_{harness}$ | 1.20 | Mass of the harness subsystem |

Table 1: Margins applied in the low thrust spacecraft model.

### 4.2 The BPA structure

We have selected to consider in this application three parameters as uncertain: $\eta_p$, $\rho_{SA}$ and $\rho_{harness}$ that appear in respectively Eq. (17), (18) and (23).

The table 2 presents their BPA structure.

The use of system margins is classically to compensate for uncertainties. As we are aiming here at crystallising the uncertainties with the Evidence Theory, we selected parameters as uncertain when they were associated to a system margin. In our example, these are $\kappa_A$, $\kappa_{SA}$ and $\kappa_{harness}$. Therefore they are set to 0 for the OUU problem. Note that the BPA structure is such that the effect of the 3 parameters being considered as uncertain is artificially equivalent to applying the default system margins. The consequence is that the optimal design of the OUU is the same as the deterministic one. This is obviously not generally the case but helps here to better comprehend the results.

### 4.3 Problem formulation

Using the notation of section 2.3, let us define $\mathbf{d}$ the vector of the design variables, $\mathbf{u}$ the vector of the uncertain parameters and $\mathcal{D}$ and $\mathcal{U}$ their respective domain:

$$\mathbf{d} = \left[P_{1AU}, T_{max}, I_{SP}_{max\ T}\right] \quad (25)$$

$$\mathcal{D} = [4200\text{ W}, 6450\text{ W}] \times [210\text{ mN}, 400\text{ mN}] \times [4000\text{ s}, 8000\text{ s}] \quad (26)$$

$$\mathbf{u} = [\eta_p, \rho_{SA}, \rho_{harness}] \quad (27)$$

$$\mathcal{U} = [0.18959, 0.22751] \times [2.89, 3.3105] \times \left[1.3763 \cdot 10^{-3}, 1.6515 \cdot 10^{-3}\right] \quad (28)$$

The problem that is aimed at being solved here is thus:

$$\begin{cases} \max\limits_{m^*_{SEP}{}_{wet} \in \mathbb{R}^+, \mathbf{d} \in \mathcal{D}} & Bel_{\mathcal{U}}\Big(m_{SEP}{}_{wet}(\mathbf{d}, \mathbf{u}) < m^*_{SEP}{}_{wet}\Big) \\ \max\limits_{m^*_{SEP}{}_{wet} \in \mathbb{R}^+, \mathbf{d} \in \mathcal{D}} & m^*_{SEP}{}_{wet} \end{cases} \quad (29)$$

## 5 COMPARISON OF THE THREE PROPOSED METHODS

### 5.1 Results

For the step method, the initial threshold has been set to 920 kg and the threshold decreased by 0.5 kg

| Uncertain parameter | Intervals Lower bound | Intervals Upper bound | Basic probability assignment |
|---|---|---|---|
| $\eta_p$ | 0.18959 | 0.195 | 0.05 |
| | 0.195 | 0.205 | 0.15 |
| | 0.205 | 0.215 | 0.25 |
| | 0.215 | 0.22751 | 0.55 |
| $\rho_{SA}$ | 2.89 | 3.00 | 0.10 |
| | 3.00 | 3.10 | 0.15 |
| | 3.10 | 3.25 | 0.35 |
| | 3.25 | 3.3105 | 0.40 |
| $\rho_{harness}$ | $1.3763 \cdot 10^{-3}$ | $1.4500 \cdot 10^{-3}$ | 0.05 |
| | $1.4500 \cdot 10^{-3}$ | $1.5500 \cdot 10^{-3}$ | 0.25 |
| | $1.5500 \cdot 10^{-3}$ | $1.6000 \cdot 10^{-3}$ | 0.30 |
| | $1.6000 \cdot 10^{-3}$ | $1.6515 \cdot 10^{-3}$ | 0.40 |

Table 2: Uncertainty representation through the Evidence theory

at each iteration. The initial design chosen was the centre of the domain $\mathcal{D}$. 72 iterations where required to reach an optimal belief lower than 1 ($m^*_{SEP\ wet} = 883.5$ kg), and 30 more to arrive at a belief of 0 for any design. Note that the first 72 iterations are very quick as the optimisation is stopped as soon as a belief equal to 1 is found.

For the cluster approximation, the initial threshold has been set at 872 kg and the step was 0.5 kg between iteration. Remember that while the step method decreases the threshold, the cluster approximation increases it. 34 iterations where required to reach for the first time a belief of 1.

Concerning the direct method using the MOO, the threshold $m^*_{SEP\ wet}$ has been bound to [860 kg, 900 kg]. In order to speed up the computation of the belief for a given design vector $\mathbf{d}$ and a given threshold, an initial Pareto front is required. The belief function of the design point at the centre of the design domain $\mathcal{D}$ has been used. After about 100,000 evaluation of the belief, 53 different Pareto optimal points have been found, all in the same neighbourhood.

## 5.2 Pareto Front

The figure 6 shows the Pareto front identified by the three proposed method. It can bee seen that the Direct Approach and Step Method give the best results. They are in fact extremely close to the Belief function of the optimal deterministic design (not shown here for clarity). Indeed, the BPA structure was artificially built to compensate for the system margins traditionally used. The cluster approximation gives the worst characterisation but still remains quite accurate. Moreover, the result is conservative (higher threshold of the same belief level) thus more robust.

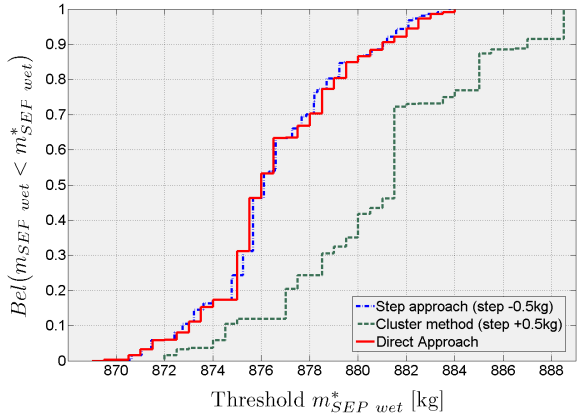


Figure 6: Comparison of the results given by the 3 proposed methods (direct, step and cluster) to solve the OUU problem of the BepiColombo test case.

## 5.3 Computational Time

The computational time is given in the table 3.

The step approach takes 13 times less than the use of the MOO for a very similar result, while the cluster approximation requires about 3 times less and producing a not so good Pareto front. However, the number of system function evaluation is in favour of the cluster approximation has it requires 2.5 times less evaluation of $f$ than the step

| Method | Total CPU time | CPU time for $f$ | Number of $f$ evaluations |
|---|---|---|---|
| Step | 288s | 133s | 330,375 |
| Cluster | 1206s | 59s | 135,803 |
| Direct | 3859s | 1926s | 7,122,580 |

Table 3: computational time for the 3 proposed approaches.

method and an impressive 52 times less than with the multi-objective optimiser. This information is important as the system function can be computationally expensive, making the cluster approximation more attractive. To give an idea, in our present case, a system function evaluation requires around $4 \cdot 10^{-4}$ seconds. If instead the CPU time for computing the wet mass of the SEP system was over $5 \cdot 10^{-3}$, the cluster approximation would become faster.

### 5.4 Influence of the number of focal elements

The same simulations have been run with only the cluster approximation and the step method for increasing number of focal elements (i.e. the number of intervals per incertain parameters has been increased to 8 and 16). The figure 7 shows how the cluster approximation can become even more attractive in term of computational time. However, it has to be highlighted that such numerous intervals per uncertain parameters might not be frequent in practice. 4 to 6 intervals seems a reasonable number.

## 6 CONCLUSION AND FUTURE WORK

In this paper, we have presented an alternative to the system margin approach to deal with uncertainties inherent to the preliminary phase of a system design. The Evidence Theory appeared to be well adapted to preliminary design as epistemic uncertainties are frequently encountered. Three different approaches to solve the optimisation under uncertainties problem has been proposed and tested on a classical preliminary space mission design. Each of them had some pros and cons. The step method appeared efficient for problem with convex and cheap to compute system function. However, it does not guaranty the global optimum

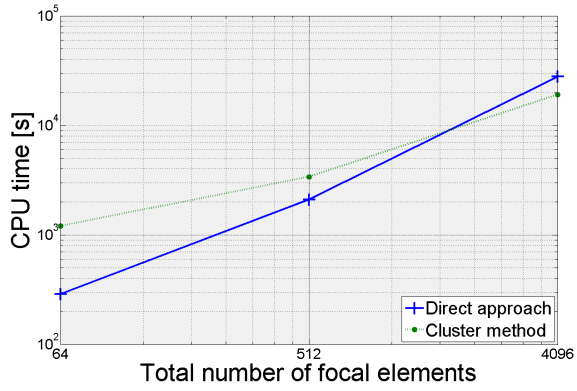


Figure 7: Variation of the computational time for the cluster approximation and the step method for increasing number of focal elements (simulations based on the low thrust mission of BepiColombo (step of 0.5 kg)).

to be found for each level of the threshold. The cluster approximation produces quite accurate results and appears to be very attractive for non convex and multimodal system functions or expensive ones. Finally, the use of an MOO removes the issue of selecting the step between two successive thresholds at the expense of a higher computational time. In this respect a redefinition of the local Pareto optimality criterion is needed and will be presented in a future work.

Furthermore, different and more efficient options to characterise the feasible subdomain are under investigation. Finally, other test cases with more uncertain parameters and with more than one local minimum are required to complete the comparison among different approaches.

## References

[1] A. P. Dempster. Upper and lower probabilities induced by a multivalued mapping. *Ann. Math. Statist.*, 38, 1967.

[2] G. Shafer. *A Mathematical Theory of Evidence*. Book, 1976.

[3] W.L. Oberkampf and J.C. Helton. Investigation of evidence theory for engineering applications. In *4th Non-Deterministic Approaches Forum*, volume 1569. AIAA, April 2002.

[4] M. Vasile. Robustness Optimisation of Aerocapture Trajectory Design Using a Hybrid

Co-Evolutionary Approach. In *18th International Symposium on Space Flight Dynamics*, volume 548 of *ESA Special Publication*, 2004.

[5] M. Vasile and D. Bonetti. Evolution of the concurrent design process under uncertainties. In *International Concurrent Engineering Workshop*. ESA, 2004.

[6] J. C. Helton. Uncertainty and sensitivity analysis in the presence of stochastic and subjective uncertainty. *Journal of Statistical Computation and Simulation*, 57, 1997.

[7] Kari Sentz and Scott Ferson. Combination of evidence in Dempster-Shafer theory. In *6th World Multi-conference on Systemics*. Cybernetics and Informatics, July 2002.

[8] Harish Agarwal, John E. Renaud, and Evan L. Preston. Trust region managed reliability based design optimization using evidence theory. *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 5:3449 – 3463, 2003. Evidence theory;Multidisciplinary systems analysis;Discontinuous function;.

[9] Ha-Rok Bae, Ramana V. Grandhi, and Robert A. Canfield. Uncertainty quantification of structural response using evidence theory. *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 4:2135 – 2145, 2002. Dempster Shafer theory;.

[10] M. Vasile. Robust mission design through evidence theory and multiagent collaborative search. *Annals of the New York Academy of Sciences*, 1065:152–173, December 2005.

[11] George B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1965.

[12] Brian D. Bunday. *Basic Linear programming*. Hodder Arnold, 1984.

[13] S. Kemble. *Interplanetary Mission Analysis and Design*. Springer Praxis Books, 2006.

[14] Soren N. Lophaven, Hans B. Nielsen, and Jacob Sondergaard. DACE: a MatLab kriging toolbox. Technical Report IMM-TR-2002-12, Technical University of Denmark, 2002.