

Robust and Automated Space System Design

Martin Fuchs^{1,2}, Daniela Girimonte², Dario Izzo², and Arnold Neumaier¹

¹ University of Vienna, Faculty of Mathematics, Nordbergstr. 15, 1090 Wien, Austria martin.fuchs@univie.ac.at, arnold.neumaier@univie.ac.at

² European Space Agency, Advanced Concepts Team, ESTEC, EUI-ACT, Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands
daniela.girimonte@esa.int, dario.izzo@esa.int

Summary. Over the last few years, much research has been dedicated to the creation of decisions support systems for space system engineers or even for completely automated design methods capturing the reasoning of system experts. However, the problem of taking into account the uncertainties of variables and models defining an optimal and robust spacecraft design have not been tackled effectively yet. This chapter proposes a novel, simple approach based on the *clouds* formalism to elicit and process the uncertainty information provided by expert designers and to incorporate this information into the automated search for a robust, optimal design.

1.1 Introduction

The design of a spacecraft is a demanding challenge. The complexity of the task and its multidisciplinary nature make it difficult to obtain a complete survey and a deep understanding of the whole design process.

In a classical approach to multidisciplinary design (cf. [Roy, 1996], [Belton and Stewart, 2002]), each specialist would prepare a subsystem design rather independently, using stand-alone tools. Design iterations among the different discipline experts would take place in meetings at intervals of a few weeks. This well-established approach reduces the opportunity of finding interdisciplinary solutions and to create system awareness in the specialists. A considerable step towards a multidisciplinary approach in the early phases of space system design has been achieved by concurrent engineering where a sequential iterative routine is replaced by a parallel and cooperative procedure. Design facilities where these methodologies are implemented are, among others, the ESA Concurrent Design Facility [Bandecchi et al., 1999], the NASA Goddard Integrated Mission Design Center [Karpatei et al., 2003] and the Concept Design Center at the AeroSpace Corporation [Aguilar et al., 1998].

In these facilities it is common practice of preliminary spacecraft design to handle uncertainties by assigning intervals, or safety margins, to the uncertain variables, usually combined with an iterative process of refining the

intervals while converging to a robust optimal design. The refinement of the intervals is done by the system experts who assess whether the worst-case scenario, which has been determined for the design at the current stage of the iteration process, is too pessimistic or too optimistic. How to assign the intervals and how to choose the endpoint of the assigned intervals to get the worst-case scenario is usually not computed but assessed by a system expert. The goal of the whole iteration process includes both optimization of the design and safeguarding against uncertainties. The available uncertainty information in the early phase of a spacecraft design is often very limited, mostly there are only interval bounds on the uncertain variables, sometimes probability distributions for single variables without correlation information. When the amount of available uncertainty information is small, traditional methods face several problems. To make use of well known current methods from probability theory or fuzzy theory (e.g., fuzzy clustering) more such information would be required. Simulation techniques like Monte Carlo also require a larger amount of information to be reliable. The lack of information typically causes these methods to underestimate the effects of the uncertain tails of the probability distribution (cf. [Ferson, 1996]). Similarly, a reduction of the problem to an interval analysis after assigning intervals to the uncertain variables as described before (e.g., 3σ boxes) entails a loss of valuable uncertainty information which would actually be available but not involved in the uncertainty model. Moreover, in higher dimensions the numerical computation of the error probabilities is very expensive, if not impossible, even given the knowledge of the multivariate probability distributions.

Many previous works deal with uncertainty modelling applied to space systems design. In [Pate-Cornell and Fischbeck, 1993] probability risk analysis is applied to the uncertainties in space shuttle design; an approach from fuzzy theory can be found e.g., in [Ross, 1995]; in [Thunnissen, 2005] a general qualitative and quantitative investigation of uncertainties in space design is given. The work by [Amata et al., 2004] presents studies harmonizing the interests from different disciplines in multidisciplinary design optimization; and a decision support tool for spacecraft design is implemented in [Zonca, 2004]. The attempt to incorporate both uncertainty and autonomy in the design process was made e.g., in [McCormick and Olds, 2002] using Monte-Carlo simulation techniques, or with a fuzzy logic approach in [Lavagna and Finzi, 2002]. The ESA Advanced Concepts Team in cooperation with the University of Vienna performed an Ariadna study on the application of the clouds theory in space design optimization [Neumaier et al., 2007]. This study presented an initial step on how clouds could be applied to handle uncertainties in spacecraft design. This chapter goes a step further in this direction.

Generally speaking, the task of robust and automated space system design cannot be regarded as a single task, but consists of two tasks that have to be accomplished concurrently. First, the design should be robust; in other words: the design should be safeguarded against uncertain perturbations. Second, the design should be found automatically; this indicates the existence of a method

which is able to find the optimal design choice automatically. The optimality of a spacecraft design can depend on multiple objectives, such as the cost or the mass of the spacecraft or both at the same time. Continuing the work from [Neumaier et al., 2007] to accomplish the two tasks, this chapter presents a newly developed methodology to gather all available uncertainty information from system experts, process it to a reliable worst-case analysis and finally optimize the design seeking the optimal robust design.

The presented approach handles and processes information gathered by the clouds formalism introduced in [Neumaier, 2004a]. Clouds allow the representation of incomplete stochastic information in a clearly understandable and computationally attractive way, mediating between aspects of fuzzy set theory and probability distributions (cf. [Dubois and Prade, 2005]). The use of clouds permits an adaptive worst-case analysis without losing track of important probabilistic information. At the same time, all computed probabilities, and hence the resulting designs, are reasonably safeguarded against perturbations due to unmodelled and possibly unavailable information. For given confidence levels, the clouds provide regions of relevant scenarios affecting the worst-case for a given design. This work has the ambitious goal to achieve a quantification of reliability close to classical probability theory methods, but in higher dimensional spaces of uncertain scenarios so that one can deal with real-life system design.

To find a robust optimal design automatically, heuristic optimization methods were developed that take advantage of inherent characteristics of spacecraft design problems. This allows to investigate applications to concrete spacecraft design problems.

Figure 1.1 illustrates the basic concept of the approach. The expert provides the underlying system model, given as a black-box model, and all currently available uncertainty information on the input variables of the model. The information is processed to generate a *cloud* that provides a nested collection of regions of relevant scenarios parameterized by a confidence level α , thus produces safety constraints for the optimization. The optimization minimizes a certain objective function (e.g., cost, mass) subject to the safety constraints to account for the robustness of the design, and subject to the functional constraints which are represented by the system model. The results of the optimization, i.e., the automatically found optimal design point and the worst-case analysis, are returned to the expert, who is given an interactive possibility to provide additional uncertainty information afterwards and rerun the procedure.

The uncertainty information can be provided on the one hand as bounds or marginal probability distributions on the uncertain variables. On the other hand, the engineers can adaptively improve the uncertainty model, even if their expert knowledge is only little formalized, by adding correlation constraints to exclude scenarios deemed irrelevant. The information can also be provided as real sample data, if available.

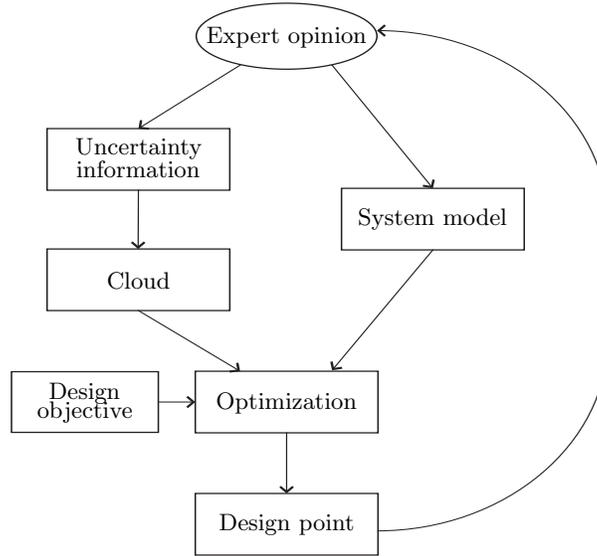


Fig. 1.1. Basic concept.

This chapter is organized as follows. Section 1.2 and Section 1.3 present a more detailed investigation of the uncertainty modelling and the design optimization methods. These techniques are applied to an example from spacecraft system design described in Section 1.4. Section 1.5 discusses general and detailed aspects, remarks, problems and advantages of the approach. Section 1.6 concludes the chapter with a summary of findings and an outlook on possible future work.

1.2 Uncertainty Modelling

The concept of clouds was introduced in [Neumaier, 2004a] as a new notion for handling uncertainty. Clouds describe the rough shapes of typical samples of various size, without fixing the details of the distribution. The special case of interest for large-scale models is a potential-based cloud. This section deals with these potential clouds, with their properties, computation, and their illustration through examples. Initially, the section gives a theoretical introduction to potential clouds, i.e., how they are defined, their probabilistic properties, and how they can be interpreted intuitively. The section then describes how they can be generated computationally, if the uncertainty information is given by marginal distributions or boxes for uncertain variables, and how they will constrain the optimization problem.

1.2.1 Potential Based Clouds

A potential based cloud is defined by a continuous potential V which assigns to each scenario ε from a set $\mathbb{M} \subseteq \mathbb{R}^n$ a value $V(\varepsilon) \in \mathbb{R}^+$ defining the shape of the cloud (see e.g., Figure 1.2), and a lower probability $\underline{\alpha}(U)$ and an upper probability $\overline{\alpha}(U)$ defining the boundary of the cloud, such that, for all $U \in \mathbb{R}^+$:

$$\underline{\alpha}(U) \leq \Pr(V(\varepsilon) < U) \leq \Pr(V(\varepsilon) \leq U) \leq \overline{\alpha}(U), \quad (1.1)$$

where $\varepsilon \in \mathbb{M}$ a random variable, $\underline{\alpha}$ and $\overline{\alpha}$ strictly increasing continuous functions of U mapping the range of V to $[0, 1]$.

The mapping $x \rightarrow [\underline{\alpha}(V(x)), \overline{\alpha}(V(x))]$, $x \in \mathbb{M}$, is the potential based cloud. The functions $\underline{\alpha}$ and $\overline{\alpha}$ can be interpreted as boundings on the distribution of $V(\varepsilon)$.

For a given failure probability p_{err} and $\alpha := 1 - p_{\text{err}}$, the so-called α -cut describes an inner region \underline{C}_α (lower α -cut) of α -relevant scenarios and a (generally larger) region \overline{C}_α (upper α -cut) of α -reasonable scenarios. Define $\underline{C}_\alpha := \{\varepsilon \in \mathbb{M} \mid V(\varepsilon) \leq \underline{U}_\alpha\}$ if $\underline{U}_\alpha := \min\{U_\alpha \in V(\mathbb{M}) \mid \overline{\alpha}(U_\alpha) = \alpha\}$ exists, and $\underline{C}_\alpha := \emptyset$ otherwise; analogously $\overline{C}_\alpha := \{\varepsilon \in \mathbb{M} \mid V(\varepsilon) \leq \overline{U}_\alpha\}$ if $\overline{U}_\alpha := \max\{U_\alpha \in V(\mathbb{M}) \mid \underline{\alpha}(U_\alpha) = \alpha\}$ exists, and $\overline{C}_\alpha := \mathbb{M}$ otherwise. The conditions defining the cloud (i.e., $\underline{\alpha} \leq \overline{\alpha}$, $\underline{\alpha}$ and $\overline{\alpha}$ strictly increasing and monotone) guarantee, that $\underline{U}_\alpha \leq \overline{U}_\alpha$, and that there is a region C_α with $\underline{C}_\alpha \subseteq C_\alpha \subseteq \overline{C}_\alpha$ containing a fraction of α of all scenarios considered possible.

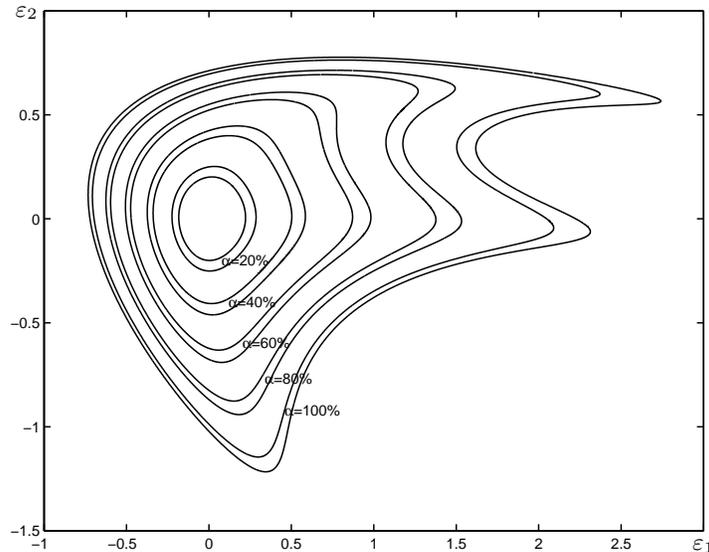


Fig. 1.2. α -cuts of a 2-dimensional potential based cloud.

1.2.2 Cloud Generation and Constraints

This text now describes a method to generate computationally a cloud that matches the above definition, assuming the uncertainty information is given by marginal distributions or boxes for the vector of uncertain variables. At first, a sample S of N sample points is generated. The sample points are chosen from a grid fulfilling the well-known Latin hypercube condition (see e.g., [McKay et al., 1979]). If only boxes are given, then the grid is equidistant, if marginal distributions are given the grid is transformed with respect to them to ensure that each grid interval has the same marginal probability. Thus the generated sample represents the marginal distributions. However, after a modification of S , e.g., by cutting off sample points as is done later, an assignment of weights to the sample is necessary to preserve the marginal distributions.

The weights are computed by satisfying the following conditions. Let $S = \{x_1, \dots, x_N\}$ be a set of N sample points in \mathbb{R}^n . Let x_i^k , $i = 1, \dots, N$ be the projection of x_i to the k^{th} coordinate, π_k a sorting permutation of $\{1, \dots, N\}$, such that $x_{\pi_k(1)}^k \leq x_{\pi_k(2)}^k \leq \dots \leq x_{\pi_k(N)}^k$. Let I be the index set of those entries of the uncertainty vector ε where a marginal probability distribution F_k , $k \in I$ is given. For all $k \in I$ and $i = 1, \dots, N$ the weights $\omega_1, \dots, \omega_N$ have to be nonnegative and satisfy the following constraints:

$$\sum_{j=1}^i \omega_{\pi_k(j)} \in [F_k(x_{\pi_k(i)}^k) - d, F_k(x_{\pi_k(i)}^k) + d], \quad \sum_{i=1}^N \omega_i = 1. \quad (1.2)$$

The constraints (Eq. 1.2) require the weights to represent the marginal distributions with some reasonable margin d . In other words, the weighted empirical distribution of the sample projected to a margin should not differ from the given marginal probability distribution by more than d . In practice, one chooses d with Kolmogorov-Smirnov statistics.

Initially, one chooses the potential function V to be boxed shaped, i.e., $V(\varepsilon) := \max_k \frac{|\varepsilon^k - \mu^k|}{r^k}$, where $\varepsilon, \mu, r \in \mathbb{R}^n$, $\varepsilon^k, \mu^k, r^k$ are the k^{th} components of the vectors, μ the mode, r the radius of the sample S . With the computed weights one achieves an empirical distribution for $\{V(\varepsilon), \varepsilon \in S\}$ approximating the distribution function of $V(\varepsilon)$. Smooth lower bounds $\underline{\alpha}(V(\varepsilon))$ and upper bounds $\overline{\alpha}(V(\varepsilon))$ are fitted for the empirical distribution (with a piecewise cubic Hermite spline, cf. Figure 1.3) considering the size of the sample and the quality of the approximation of the distribution function of $V(\varepsilon)$, i.e., a violation of the constraints (Eq. 1.2). The shorter the sample, or the lower the acceptable incidence of violations, the larger the width of the bounds is chosen. Having found an appropriate bounding of the distribution function of $V(\varepsilon)$ a potential cloud was generated which fulfills the conditions that define the cloud according to the remarks to Theorem 4.3 in [Neumaier, 2004a]. Thus it represents the given uncertainty information.

As already mentioned, initially a box shaped potential has been chosen, but the expert will have the option to cut off scenarios that are not considered

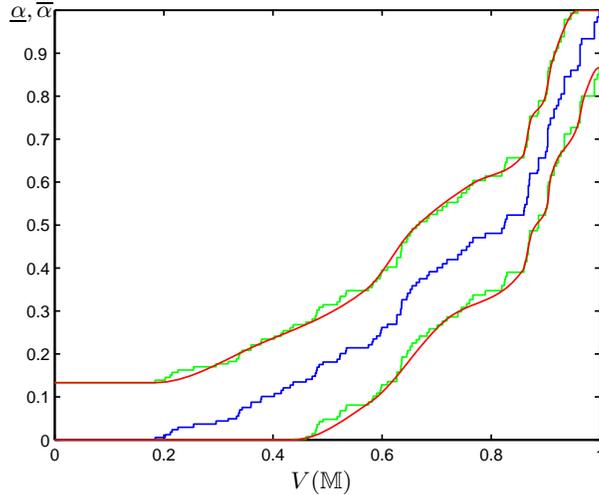


Fig. 1.3. The smooth lower bounds $\underline{\alpha}(V(\varepsilon))$ and upper bounds $\overline{\alpha}(V(\varepsilon))$ enclosing the empirical distribution of $V(\varepsilon)$. The mapping $\varepsilon \rightarrow [\underline{\alpha}(V(\varepsilon)), \overline{\alpha}(V(\varepsilon))]$ is a potential cloud (cf. Section 1.2.1).

relevant and thus specify the uncertainty information in the form of correlation bounds adaptively resulting in a polyhedron shaped potential. This iterative step imitates the procedure of decision making of real life applications and is particularly important if there is only little uncertainty information available.

As soon as the cloud representing the uncertainty information is generated, one can produce the worst-case relevant region C for a given potential. For a given confidence level α one numerically computes the solution V_α of $\overline{\alpha}(V_\alpha) = \alpha$, cf. Section 1.2.1, and defines the region $C := \{\varepsilon \mid V(\varepsilon) \leq V_\alpha\}$ if a solution V_α exists and $C := \emptyset$ otherwise, which is only the case for very low confidence levels. The next section shows that the region C can be used to constrain the design optimization problem.

1.3 Design Optimization

The focus of this section is on design optimization. The section starts with a general optimization problem formulation, points out the characteristics and difficulties that come with it, and finally give a short introduction to the heuristic approach that was developed to solve the problem.

It is assumed that the optimization problem can be formulated as a mixed-integer, bi-level problem of the following form:

$$\begin{aligned}
\min_{\theta} \max_{x, z, \varepsilon} f(x) & && \text{(objective functions)} \\
\text{s.t.} \quad z = Z(\theta) + \varepsilon & && \text{(table constraints)} \\
F(x, z) = 0 & && \text{(functional constraints)} \\
V(\varepsilon) \leq V_{\alpha} & && \text{(cloud constraint)} \\
\theta \in T & && \text{(selection constraints)}
\end{aligned} \tag{1.3}$$

Here ε is the vector of uncertainties, θ is the vector of choice variables (with one variable for each independent table of choices), z is the vector consisting of all global input variables and all design variables and x is the vector consisting of all output variables of the underlying model. The table constraints assign to each choice θ a design vector z whose value is the nominal table entry $Z(\theta)$ plus its (unknown) error ε with uncertainty specified by the cloud. The functional constraints express the functional relationships defined in the underlying system model which usually comes as a black-box as mentioned in Section 1.1. It is assumed that the number of equations and the number of output variables is the same (i.e., $\dim F = \dim x$), and that the equations are (at least locally) uniquely solvable for x . This holds in the particular case that the black-box is given in the form $F(x, z) = \overline{F}(z) - x$ with some black-box function \overline{F} and $\dim \overline{F} = \dim x$. The cloud constraint involves the potential function V as described in the last section. Because of the polyhedral structure of our clouds, the cloud constraint can be written as a collection of linear inequalities parameterized by the confidence level, a feature important for the implementation. The selection constraints specify how many choices are allowed for each choice variable.

This optimization problem (Eq. 1.3) features difficulties of the most complex nature: it is a mixed integer nonlinear program (MINLP) with a bi-level structure which cannot be handled directly with standard optimization tools. Therefore the problem is approached with heuristic methods that were developed to exploit the characteristics of the above design optimization problem.

1.3.1 Heuristics

The inner level of the problem (Eq. 1.3), i.e., finding the worst-case scenario for f over the polyhedron C for a fixed design choice θ , is solved using linear programming. To this end f is linearly approximated within a box b that contains C .

To tackle the outer level of problem (Eq. 1.3), i.e., to find the design choice θ with the minimal worst-case objective function value, two different methods are used; one based on a quadratic model, the other on separable underestimation.

In the first method, one fits a quadratic model of the objective function and minimizes this model to get a guess for the optimal solution of the problem. In this approach the discrete nature of the choice variables is ignored.

The second method, on the other hand, takes advantage of the discrete nature of θ and finds a separable underestimator $q(\theta)$ for the objective of the form:

$$q(\theta) := \sum_{i=1}^{n_0} q_i(\theta^i), \quad (1.4)$$

where n_0 is the dimension of the choice variable, and θ^i the i^{th} coordinate of θ . Let N_0 be the number of function evaluations f_1, \dots, f_{N_0} that have been made in advance for the design choices $\theta_1, \dots, \theta_{N_0}$. The expression $q_i(\theta^i)$ is simply a constant q_{i,θ^i} for integer choice variables θ^i , and $q_i(\theta^i) = q_{i1} \cdot \theta^i + q_{i2} \cdot \theta^i{}^2$, with constants q_{i1}, q_{i2} , for continuous choice variables θ^i . The constants q_i are treated as variables q_i in a linear optimization program (LP) satisfying the constraints

$$\sum_{i=1}^{n_0} q_i(\theta^i) \leq f_l \quad l = 1, \dots, N_0, \quad (1.5)$$

and are computed by an LP solver. This ensures that many constraints in (Eq. 1.5) will be active. The underestimator $q(\theta)$ can then be easily minimized afterwards.

Finally the minimizers that result from these methods are used as starting points for a limited global search that consists of an integer line search for the discrete choice variables and multilevel coordinate search [Huyer and Neumaier, 1999] for the continuous choice variables. Thus, the hope is to find the global optimal solution, but as heuristics are used there is no guarantee.

1.4 Case Study

This section applies the proposed robust and automated design method to a case study of spacecraft engineering, i.e., the Attitude Determination and Control Subsystem (ADCS) for the NASA's Mars Exploration Rover (MER) mission cf. [MER, 2003], [Erickson, 2004] whose scientific goal is to investigate the history of water on Mars. The spacecraft has no main propulsion subsystem onboard and the ADCS is composed by eight thrusters aligned in two clusters. The mission sequence after orbit injection includes a number of spin maneuvers and slew maneuvers as reported in Table 1.3. Spin maneuvers are required for keeping the gyroscopic stability of the spacecraft, whereas slew maneuvers serve to control the direction of the spacecraft and to fight the effects of solar torque. Fault protection is considered to correct possible errors made when performing nominal maneuvers.

The goal is to select the type of thrusters (from a set of possible candidates as listed in Table 1.4) considering the design objective of minimizing the total mass m_{tot} , and the worst possible performance of a thruster with respect to

m_{tot} , i.e., to find the thruster with the minimal mass worst-case scenario. The total mass consists of the fuel needed for attitude control (computed as the sum of the fuel needed for each maneuver) plus the mass of the eight thrusters that need to be mounted on the spacecraft. The variable structure and the model equations to compute the fuel mass for attitude control can be found in Appendices A and B, respectively. According to the notation introduced in (Eq. 1.3), the choice variable θ , i.e., the type of thruster, can be selected as an integer between 1 and 30.

The uncertainty specification for the model variables are taken from [Thunissen, 2005] and reported in Table 1.5 of Appendix E. The number of uncertain variables in this application example is 33 plus 1 uncertain design variable.

1.4.1 Results

The cloud constraints for the optimization are generated for a confidence level of $\alpha = 95\%$ and a generated sample size of $N = 1000$. For the computation of the results the presented methods have been implemented as MATLAB code. The study makes use of the Statistics Toolbox of MATLAB to evaluate probability distributions. The study also uses CVX [Grant and Boyd, 2007] to solve linear programs, SNOBFIT [Huyer and Neumaier, 2006] and MCS [Huyer and Neumaier, 1999] as external optimization routines, and NLEQ [Deuffhard, 2004], [Nowak and Weimann, 1990] to solve systems of nonlinear equations. The results for optimization are divided into four different configurations of uncertainty handling and specifications:

- a. The uncertainties are as specified in Table 1.5. Here they are treated in a classical engineering way, assigning 3σ boxes to the uncertain variables which corresponds to a 99.7% confidence interval for a single normally distributed variable. Then the optimal design choice is $\theta = 9$ with an objective function value of $m_{tot} = 3.24$ kg in the nominal case and $m_{tot} = 5.56$ kg in the worst case.
- b. The uncertainties are again as in Table 1.5. Here we use the methods presented in the last sections and find $V_\alpha = 0.9922$ and the optimal design choice $\theta = 9$, as in Configuration a. However, when comparing the worst-case analysis of b and a, it is apparent that the results for the 3σ boxes are far too optimistic to represent a reliable worst-case scenario, the value of m_{tot} is now 8.08 kg instead of 5.56 kg for the 3σ boxes.
- c. This configuration does not take any uncertainties into account, generally assuming the nominal case for all uncertain input variables. The optimal design choice then is $\theta = 3$ with a value of $m_{tot} = 2.68$ kg in the nominal, but $m_{tot} = 8.75$ kg in the worst case, which is significantly worse than in Configuration b.
- d. The uncertainties are obtained by taking the values from Table 1.5 and doubling the standard deviation of the normally distributed variables. It

is interesting to report that if one increases the uncertainty in the normally distributed uncertain variables simply in this way, the optimal design choice changes to $\theta = 17$ with a value of $m_{tot} = 3.38$ kg in the nominal and $m_{tot} = 9.49$ kg in the worst case.

The results are summarized in Table 1.1, showing the optimal design choice for each configuration and the corresponding value of the objective function m_{tot} for the nominal case and for the worst case, respectively. Configuration b is our reference configuration as in this case we apply our new methods given the original uncertainty information from Table 1.5.

Table 1.1. Nominal and worst-case values of m_{tot} for different design choices obtained by the four different configurations.

Configuration	Design Choice θ	Nominal value m_{tot}	Worst-case m_{tot}
a	9	3.24	5.56
b	9	3.24	8.08
c	3	2.68	8.75
d	17	3.38	9.49

The results reported above show a number of important facts related to spacecraft design. The comparison between the configurations b and d suggests that in a preliminary stage of the spacecraft systems modelling the optimal design point is quite sensitive to the uncertainty description, a fact well-known to the system engineers who see their spacecraft design changing frequently during preliminary phases when new information becomes continuously available. The presented method captures this important dynamics and describes it in rigorous mathematical terms.

The comparison between the configurations b and c suggests that the uncertainties need to be accounted for at an early stage in order not to critically overestimate the spacecraft performances.

Finally, the comparison between configurations b and a suggests that the simple 3σ analysis of uncertainties, very frequent in real practice, produces a quite different estimation of the spacecraft performances with respect to a more rigorous accounting of the uncertainty information.

We see that Configuration b is the best approach to the goal presented in this paper, i.e., minimizing the design worst-case while reasonably taking account of uncertainties.

1.5 Discussion

The importance of a robust design has been the starting point and main motivation of this research work and the results obtained from a case study confirm that the optimal spacecraft design is strongly sensitive to uncertainties. At the present stage, it can be confirmed that neglecting uncertainties results in a design that completely lacks robustness and a simplified uncertainty model (like a 3σ approach) may yield critical underestimations of worst-case scenarios.

When trying to collect the uncertainty information, it turned out to be very difficult to get useful, formalized information directly from expert engineers. To collect all information, both formalized and unformalized, an interactive dialogue between expert and computer can be realized by a graphical user interface where the engineers can specify uncertainties, provide sample data, cut off worst-case irrelevant scenarios, and adjust the quality of the uncertainty model. It is expected that this kind of interaction is an inevitable next step in space design.

Now, the discussion continues with more detailed considerations on the study.

- In the theory of clouds (cf. Section 1.2.1) there is a distinction between regions of α -relevant, α -reasonable scenarios and borderline cases (which is the set difference of the α -reasonable and the α -relevant regions). In robust design the possibly uncertain scenarios are required to satisfy safety constraints. With respect to the presented terminology the regions above have the following interpretation: if at least one of the α -relevant scenarios fails to satisfy the safety constraints, the design is unsafe; if all of the α -reasonable scenarios satisfy the safety constraints, the design is safe. Between these two cases there is the borderline region where no precise statement can be made without additional uncertainty information. The width of the borderline region is increasing if the width of the cloud increases and vice versa. So widening the cloud enlarges the borderline region, corresponding to a lack of uncertainty information. This fact is reflected in our approach as both a smaller sample size and an increased dimension of the uncertainty result in a wider cloud.
- The width of the cloud is defined as the difference between the mappings $\underline{\alpha}$ and $\bar{\alpha}$ (cf. Section 1.2.1). The mappings were constructed to fulfill the conditions that define a cloud with an algorithm which is non-rigorous, but can grant a high, adjustable reliability of the fulfillment of the conditions. Thus the user of the algorithm is able to control the desired level of reliability.
- The choice of the potential function is arbitrary. Different shapes of the cloud (i.e., shapes of the potential) can make the worst-case analysis more pessimistic or optimistic. It is necessary to emphasize that a poor choice of the potential makes the worst-case analysis more pessimistic, but will still result in a valid robust design. This study allows a variation of the

potential by switching from a box-shaped to a polyhedron-shaped potential to enable the experts to improve the uncertainty model iteratively.

- A good weight computation (cf. Section 1.2.2) is the key to a good uncertainty representation with clouds. In higher dimensions the weight computation is very expensive. To overcome this problem and to allow the adjustment of the computation time, the relaxation radius d (cf. Section 1.2.2) must be increased carefully. The presented algorithm respects the relaxation property, widening the cloud by the amount of relaxation after evaluating the quality of the weights.
- As mentioned before, this work is limited to the use of heuristic methods since the design problem (Eq. 1.3) is highly complex and not suitable for standard optimization methods. The problem formulation here therefore asks to find the design with the optimal worst-case scenario. It is possible to trade off between the worst-case scenario and the nominal case of a design, but this would lead to a multi-objective optimization problem formulation.
- The number 34 of uncertain variables in the presented application example is large enough to make our problem representative for uncertainty handling in real-life applications.
- Though global optimality for the solution in the presented case study is very likely, as the choice variable is 1-dimensional and discrete, the heuristical methods cannot guarantee global optimality of the problem solution in general.
- The method of separable underestimation introduced in this chapter takes advantage of the discrete nature of many of the variables involved in spacecraft design, supporting, at the same time, continuous choice variables. Details about the heuristic approach for design optimization introduced in Section 1.3 will be published elsewhere.

1.6 Conclusions and Future Work

This chapter presented a new approach to robust and automated space system design. Starting from the background of the cloud theory the chapter developed methodologies to process the uncertainty information from expert knowledge towards a reliable worst-case analysis and an optimal and robust design. The presented approach is applicable to real-life problems of early phase spacecraft system design. At present, in most instances of the spacecraft design, reliability is only assessed qualitatively by the experts. This work presents a step forward towards quantitative statements about the design reliability.

The adaptive nature is one of the key features of the presented uncertainty model as it imitates real-life design strategies. The iteration steps significantly improve the uncertainty information and it is possible to process the new information to an improved uncertainty model afterwards.

The presented approach is generally applicable to problems of robust design optimization, especially with discrete design choices. The advantage of achieving the optimal design automatically is undeniable. Although the new methods have already been applied to space system design problems (cf. [Neumaier et al., 2007]), one future goal is to apply them to more problem classes in order to learn from new challenges. Another aspect of future improvements of the uncertainty model with clouds is the investigation of different shapes of the potential function.

The presented approach makes it possible to process the available uncertainty information to perform a reliable worst-case analysis linked to an adjustable confidence level. An additional value of the uncertainty model is the fact that one can capture various forms of uncertainty information, even those less formalized. There is no loss of valuable information, and the methods are capable of handling the uncertainties reliably, even if the amount of information is very limited.

In summary, the presented methods offer an exciting new approach to face the highly complex problem of robust and automated system design, an approach which is easily understandable, reliable and computationally realizable.

Appendix

A Model Variable Structure

The 49 variables involved in the model fall into the following four categories:

- **7 fixed parameters.**
Input variables for the model with fixed values and no uncertainty (for the values see Table 1.2).
 1. c_0 , speed of light in a vacuum
 2. d , average distance from the spacecraft to the sun in AU
 3. g_0 , gravity constant
 4. t , total mission time
 5. θ_i , sunlight angle of incidence
 6. χ , the specific impulse efficiency parameter is a property of a particular thruster. Lacking the specification of χ for several thrusters we fixed χ to take the same values for all thrusters.
 7. c_1 , the numerical solution x of $\tan(x) - x/(1 - \chi) = 0$
- **33 Uncertain input variables.**
The uncertainties are specified by probability distributions for each of these variables (cf. Appendix E).
 1. A_{max} , maximal cross-sectional area
 2. J_{xx} , J_{zz} , moments of inertia
 3. R , engine moment arm

Table 1.2. Values of the fixed parameters.

Fixed parameter	Value
c_0	$3 \cdot 10^8$ m/s
d	1.26 AU
g_0	9.81 m/s ²
t	216 days
θ_i	0°
χ	0.0375
c_1	0.334

4. δ_1, δ_2 , engine misalignment angle
5. g_s , solar constant at 1 AU
6. κ , distance from the center of pressure to the center of mass
7. ω_{spin_i} , spin rates, $i = 0 \dots 3$, given in rpm
8. ψ_{slew_i} , slew angles, $i = 1 \dots 19$, given in $^\circ$
9. q , spacecraft surface reflectivity
10. $uncfuel$, additive uncertain constant that represents inaccuracies in the equations used for the calculation of the fuel masses

- **3 Design variables.**

Thruster specifications relevant for the model. There is uncertainty information given on one of them (the thrust).

1. F , thrust
2. I_{sp} , specific impulse
3. m_{thrust} , mass of a thruster

- **6 Output variables.**

Result variables containing the objective for the optimization m_{tot} .

1. m_{fp} , fuel mass needed for fault protection maneuvers
2. m_{fuel} , total fuel mass needed for all maneuvers
3. m_{slew} , fuel mass needed for slew maneuvers
4. m_{slews} , fuel mass needed for slew maneuvers fighting solar torque
5. m_{spin} , fuel mass needed for spin maneuvers
6. m_{tot} , total mass of the subsystem

B Model Equations

The background for the equations of the ADCS subsystem model are the equations from Chapter 9 of [Thunnissen, 2005]. The basic equations are as follows:

$c = I_{sp} \cdot g_0$	(1.6)	$F_{ideal_{tot}} = 2 \cdot F$	(1.8)
$r = \sin(40^\circ) \cdot R$	(1.7)	$F_{act_{tot}} = (\cos(\delta_1) + \cos(\delta_2)) \cdot F$	(1.9)

To calculate the fuel mass m_{spin} needed for one spin maneuver (change in spin rate from ω_{spin_i} to $\omega_{spin_{i+1}}$, $i = 0...2$) the following equations are given:

$\Delta\omega_{ideal} = \omega_{spin_i} - \omega_{spin_{i+1}} $ (1.10)	$I_{actual} = t_{spin} \cdot F_{act_{tot}}$ (1.13)
$I_{ideal} = \frac{\Delta\omega_{ideal} \cdot J_{zz}}{r}$ (1.11)	$m_{spin} = \frac{I_{actual}}{c}$ (1.14)
$t_{spin} = \frac{I_{ideal}}{F_{ideal_{tot}}}$ (1.12)	

To calculate the fuel mass m_{slew} needed for one slew maneuver the following equations are given (requires the slew angle ψ_{slew} for the maneuver and the current spin rate ω_{spin} at the time the maneuver is performed):

$t_{half_rev} = \frac{\pi}{\omega_{spin}}$ (1.15)	$\Delta\psi = \frac{\psi_{slew}}{n_{pulses_{ideal}}}$ (1.22)
$t_{on_{ideal}} = \frac{2 \cdot c_1}{\omega_{spin}}$ (1.16)	$\Delta I_{torque} = H \cdot \Delta\psi$ (1.23)
$\Delta\phi_{ideal} = t_{on_{ideal}} \cdot \omega_{spin}$ (1.17)	$\Delta\phi = 2 \cdot \arcsin\left(\frac{\Delta I_{torque} \cdot \omega_{spin}}{2 \cdot F_{ideal_{tot}} \cdot r}\right)$ (1.24)
$\Delta\tau = \frac{2 \cdot F_{ideal_{tot}} \cdot r}{\Delta\phi_{ideal} \cdot \sin\left(\frac{\Delta\phi_{ideal}}{2}\right)}$ (1.18)	$t_{on} = \frac{\Delta\phi}{\omega_{spin}}$ (1.25)
$H = J_{zz} \cdot \omega_{spin}$ (1.19)	$\eta = \frac{t_{on}}{t_{half_rev}}$ (1.26)
$\Delta\psi_{ideal} = \frac{\Delta\tau \cdot t_{on_{ideal}}}{H}$ (1.20)	$c_{sd} = c \cdot \eta^x$ (1.27)
$n_{pulses_{ideal}} = \left\lceil \frac{\psi_{slew}}{\Delta\psi_{ideal}} \right\rceil$ (1.21)	$m_{slew} = n_{pulses_{ideal}} \cdot \frac{F_{act_{tot}} \cdot t_{on}}{c_{sd}}$ (1.28)

To calculate the total fuel mass m_{fuel} needed for all maneuvers we compute for each maneuver to be performed the mass m_{spin} or m_{slew} (depends on the maneuver type), and achieve m_{fuel} as the sum of these masses. To calculate the total mass m_{tot} we compute

$$m_{tot} = m_{fuel} \cdot (1 + uncfuel) + 8 \cdot m_{thrust} \quad (1.29)$$

C MER Mission Sequence

The sequence of maneuvers for the MER mission is listed in Table 1.3.

Table 1.3. Mission sequence (cf. [Thunnissen, 2005]).

Mission sequence event	Maneuver type	parameter	value	unit
De-spin from 3 rd stg.	spin	ω_{spin1}	2.000	rpm
A-practice	slew	ψ_{slew1}	5.000	°
ACS-B1	slew	ψ_{slew2}	50.45	°
ACS-B2	slew	ψ_{slew3}	5.130	°
ACS-B3	slew	ψ_{slew4}	6.350	°
ACS-B4	slew	ψ_{slew5}	2.760	°
ACS-B5	slew	ψ_{slew6}	8.510	°
ACS-B6	slew	ψ_{slew7}	9.880	°
ACS-B7	slew	ψ_{slew8}	5.640	°
ACS-B8	slew	ψ_{slew9}	5.040	°
ACS-B9	slew	ψ_{slew10}	5.750	°
ACS-B10	slew	ψ_{slew11}	4.470	°
ACS-B11	slew	ψ_{slew12}	5.530	°
ACS-B12	slew	ψ_{slew13}	5.850	°
FP: spin event	spin	ω_{spin2}	2.750	rpm
FP: spin recovery	spin	ω_{spin3}	7.410	rpm
FP: emergency slew 1	slew	ψ_{slew14}	15.75	°
FP: emergency slew 2	slew	ψ_{slew15}	15.75	°
FP: emergency slew 3	slew	ψ_{slew16}	15.75	°
FP: emergency slew 4	slew	ψ_{slew17}	15.75	°
FP: emergency slew 5	slew	ψ_{slew18}	15.75	°
FP: emergency slew 6	slew	ψ_{slew19}	15.75	°

D Thruster Specification

Table 1.4 shows the thruster specifications and the linked choice variable θ . The table entries are sorted by the thrust F . The difference between the so-called design and choice variables can be seen easily in this table: the table represents 30 discrete choices in \mathbb{R}^3 . The 3 design variables are the 3 components of these points in \mathbb{R}^3 . The choice variable θ is 1-dimensional and has an integer value between 1 and 30. The various sources for the data contained in Table 1.4 are [EADS, 2007], [Thunnissen, 2005], [Purdue School of Aeronautics and Astronautics, 1998], [Zonca, 2004], [Personal communication with ESA engineers, 2007].

E Uncertainty Specification

All uncertainty specifications taken from [Thunnissen, 2005] are reported in Table 1.5. The notation used for the distributions is:

- $U(a, b)$: uniform distribution in (a, b) ,
- $N(\mu, \sigma)$: normal distribution with mean μ and variance σ^2 ,

Table 1.4. Thruster specifications and the linked choice variable θ : Thrust F in Newtons, specific impulse I_{sp} in seconds, mass m_{thrust} in grams.

θ	Thruster	F	I_{sp}	m_{thrust}
1	Aerojet MR-111C	0.27	210.0	200
2	EADS CHT 0.5	0.50	227.3	200
3	MBB Erno CHT 0.5	0.75	227.0	190
4	TRW MRE 0.1	0.80	216.0	500
5	Kaiser-Marquardt KMHS Model 10	1.0	226.0	330
6	EADS CHT 1	1.1	223.0	290
7	MBB Erno CHT 2.0	2.0	227.0	200
8	EADS CHT 2	2.0	227.0	200
9	EADS S4	4.0	284.9	290
10	Kaiser-Marquardt KMHS Model 17	4.5	230.0	380
11	MBB Erno CHT 5.0	6.0	228.0	220
12	EADS CHT 5	6.0	228.0	220
13	Kaiser-Marquardt R-53	10	295.0	410
14	MBB Erno CHT 10.0	10	230.0	240
15	EADS CHT 10	10	230.0	240
16	EADS S10 - 01	10	286.0	350
17	EADS S10 - 02	10	291.5	310
18	Aerojet MR-106E	12	220.9	476
19	SnM 15N	15	234.0	335
20	TRW MRE 4	18	217.0	500
21	Kaiser-Marquardt R-6D	22	295.0	450
22	Kaiser-Marquardt KMHS Model 16	22	235.0	520
23	EADS S22 - 02	22	290.0	650
24	ARC MONARC-22	22	235.0	476
25	ARC Leros 20	22	293.0	567
26	ARC Leros 20H	22	300.0	408
27	ARC Leros 20R	22	307.0	567
28	MBB Erno CHT 20.0	24	234.0	360
29	EADS CHT 20	25	230.0	395
30	Daimler-Benz CHT 400	400	228.6	325

- $L(\mu, \sigma)$: lognormal distribution, distribution parameters μ and σ (mean and standard deviation of the associated normal distribution),
- $\Gamma(\alpha, \beta)$: gamma distribution with mean $\alpha\beta$ and variance $\alpha\beta^2$.

The uncertainty information on the design variable F should be interpreted as follows: The actual thrust of a thruster is normally distributed, has the mean F_{table} ($:=$ the nominal value for F specified in Table 1.4) and standard deviation $\frac{7}{300}F_{table}$.

Table 1.5. ADCS uncertainty specifications.

Variable	Probability Distribution	Variable	Probability Distribution
A_{max}	$N(5.31, 0.053)$	ψ_{slew6}	$N(8.51, 0.4)$
J_{xx}	$U(300, 450)$	ψ_{slew7}	$N(9.88, 0.5)$
J_{zz}	$U(450, 600)$	ψ_{slew8}	$N(5.64, 0.2)$
R	$N(1.3, 0.0013)$	ψ_{slew9}	$N(5.04, 0.2)$
δ_1	$N(0, 0.5)$	ψ_{slew10}	$N(5.75, 0.2)$
δ_2	$N(0, 0.5)$	ψ_{slew11}	$N(4.47, 0.1)$
g_s	$N(1400, 14)$	ψ_{slew12}	$N(5.53, 0.1)$
K	$U(0.6, 0.7)$	ψ_{slew13}	$N(5.85, 0.1)$
ω_{spin0}	$N(12, 1.33)$	ψ_{slew14}	$\Gamma(1.5, 10.5)$
ω_{spin1}	$N(2, 0.0667)$	ψ_{slew15}	$\Gamma(1.5, 10.5)$
ω_{spin2}	$\Gamma(11, 0.25)$	ψ_{slew16}	$\Gamma(1.5, 10.5)$
ω_{spin3}	$L(2, 0.0667)$	ψ_{slew17}	$\Gamma(1.5, 10.5)$
ψ_{slew1}	$N(5, 0.5)$	ψ_{slew18}	$\Gamma(1.5, 10.5)$
ψ_{slew2}	$N(50.45, 5)$	ψ_{slew19}	$\Gamma(1.5, 10.5)$
ψ_{slew3}	$N(5.13, 0.5)$	q	$N(0.6, 0.06)$
ψ_{slew4}	$N(6.35, 0.6)$	$uncfuel$	$N(0, 0.05)$
ψ_{slew5}	$N(2.76, 0.2)$	F	$N(F_{table}, 7/300F_{table})$

References

- [Aguilar et al., 1998] Aguilar, J. A., Dawdy, A. B., and Law, G. W. (1998). The Aerospace Corporations concept design center. In *8th Annual International Symposium of the International Council on Systems Engineering*.
- [Amata et al., 2004] Amata, V., Fasano, G., Arcaro, L., Della Croce, F., Norese, M., Palamara, S., Tadei, R., and Fragnelli, F. (2004). Multidisciplinary optimisation in mission analysis and design process. GSP programme ref: GSP 03/N16 contract number: 17828/03/NL/MV, European Space Agency.
- [Bandedecchi et al., 1999] Bandedecchi, M., Melton, S., and Ongaro, F. (1999). *Concurrent engineering applied to space mission assessment and design*. ESA Bulletin.
- [Belton and Stewart, 2002] Belton, V. and Stewart, T. J. (2002). *Multiple criteria decision analysis: an integrated approach*. Kluwer Academic Publishers.
- [Deuffhard, 2004] Deuffhard, P. (2004). *Newton Methods for Nonlinear Problems. - Affine Invariance and Adaptive Algorithms.*, volume 35 of *Series in Computational Mathematics*. Springer.
- [Dubois and Prade, 1986] Dubois, D. and Prade, H. (1986). *Possibility Theory: An Approach to Computerized Processing of Uncertainty*. New York: Plenum Press.
- [Dubois and Prade, 2005] Dubois, D. and Prade, H. (2005). Interval-valued fuzzy sets, possibility theory and imprecise probability. In *Proceedings of International Conference in Fuzzy Logic and Technology*.
- [EADS, 2007] EADS (2007). Space Propulsion Web Page <http://cs.space.eads.net/sp/>.
- [Erickson, 2004] Erickson, J. K. (2004). Mars exploration rover: Launch, cruise, entry, descent, and landing. In *55th International Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law*, Vancouver, Canada.

- [Ferson, 1996] Ferson, S. (1996). What monte carlo methods cannot do. *Human and Ecological Risk Assessment*, 2:990–1007.
- [Ferson et al., 1996] Ferson, S., Ginzburg, L., and Akcakaya, R. (1996). Whereof one cannot speak: When input distributions are unknown. *Risk Analysis*, in press, <http://www.ramas.com/whereof.pdf>.
- [Grant and Boyd, 2007] Grant, M. C. and Boyd, S. P. (2007). Cvx: A system for disciplined convex programming. http://www.stanford.edu/~boyd/cvx/cvx_usrguide.pdf
<http://www.stanford.edu/~boyd/cvx/>.
- [Huyer and Neumaier, 1999] Huyer, W. and Neumaier, A. (1999). Global optimization by multilevel coordinate search. *J. Global Optimization* 14, pages 331–355. <http://www.mat.univie.ac.at/~neum/software/mcs/>.
- [Huyer and Neumaier, 2006] Huyer, W. and Neumaier, A. (2006). SNOBFIT - stable noisy optimization by branch and fit. submitted preprint, <http://www.mat.univie.ac.at/~neum/ms/snobfit.pdf>
<http://www.mat.univie.ac.at/~neum/software/snobfit/>.
- [Karpati et al., 2003] Karpati, G., Martin, J., Steiner, M., and Reinhardt, K. (2003). The Integrated Mission Design Center (IMDC) at NASA Goddard Space Flight Center. In *IEEE Aerospace Conference*, volume 8, pages 3657–3667.
- [Kreinovich, 1997] Kreinovich, V. (1997). *Random Sets: Theory and Applications*, chapter Random sets unify, explain, and aid known uncertainty methods in expert systems, pages 321–345. Springer-Verlag.
- [Larson and Wertz, 1999] Larson, W. J. and Wertz, J. R. (1999). *Space Mission Analysis and Design*. Microcosm Press, 3 edition.
- [Lavagna and Finzi, 2002] Lavagna, M. and Finzi, A. E. (2002). A multi-attribute decision-making approach towards space system design automation through a fuzzy logic-based analytic hierarchical process. In *Proceedings of the 15th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*.
- [McCormick and Olds, 2002] McCormick, D. J. and Olds, J. R. (2002). A distributed framework for probabilistic analysis. In *AIAA/ISSMO Symposium On Multidisciplinary Analysis And Design Optimization*.
- [McKay et al., 1979] McKay, M., Conover, W., and Beckman, R. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 221:239–245.
- [MER, 2003] MER (2003). Mars Exploration Rover Project <http://marsrovers.nasa.gov/mission/spacecraft.html>.
- [Neumaier, 2003] Neumaier, A. (2003). On the structure of clouds. Manuscript, <http://www.mat.univie.ac.at/~neum/ms/struc.pdf>.
- [Neumaier, 2004a] Neumaier, A. (2004a). Clouds, fuzzy sets and probability intervals. *Reliable Computing* 10, pages 249–272. <http://www.mat.univie.ac.at/~neum/ms/cloud.pdf>.
- [Neumaier, 2004b] Neumaier, A. (2004b). Uncertainty modeling for robust verifiable design. Slides, <http://www.mat.univie.ac.at/~neum/ms/uncslides.pdf>.
- [Neumaier et al., 2007] Neumaier, A., Fuchs, M., Dolejsi, E., Csendes, T., Dombi, J., Banhelyi, B., and Gera, Z. (2007). Application of clouds for modeling uncertainties in robust space system design. ACT Ariadna Research ACT-RPT-05-5201, European Space Agency. Available on-line at <http://www.esa.int/gsp/ACT/ariadna/completed.htm>.

- [Nowak and Weimann, 1990] Nowak, U. and Weimann, L. (1990). A family of Newton codes for systems of highly nonlinear equations - algorithm, implementation, application. Technical report, Konrad-Zuse-Zentrum fuer Informationstechnik Berlin.
http://www.zib.de/Numerik/numsoft/CodeLib/codes/nleq1_m/nleq1.m.
- [Pate-Cornell and Fischbeck, 1993] Pate-Cornell, M. and Fischbeck, P. (1993). Probabilistic risk analysis and risk based priority scale for the tiles of the space shuttle. *Reliability Engineering and System Safety*, 40(3):221–238.
- [Personal communication with ESA engineers, 2007] Personal communication with ESA engineers (2007).
- [Purdue School of Aeronautics and Astronautics, 1998] Purdue School of Aeronautics and Astronautics (1998). Satellite Propulsion Web Page
<http://cobweb.ecn.purdue.edu/~propulsi/propulsion/rockets/satellites.html>.
- [Ross, 1995] Ross, T. J. (1995). *Fuzzy Logic with Engineering Applications*. New York, NY: McGraw-Hill.
- [Roy, 1996] Roy, B. (1996). *Multicriteria methodology for decision aiding*. Kluwer Academic Publishers.
- [Shafer, 1976] Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton, NJ: Princeton University Press.
- [Thunnissen, 2005] Thunnissen, D. P. (2005). *Propagating and Mitigating Uncertainty in the Design of Complex Multidisciplinary Systems*. PhD thesis, California Institute of Technology Pasadena.
- [Williamson, 1989] Williamson, R. C. (1989). *Probabilistic Arithmetic*. PhD thesis, University of Queensland.
- [Zonca, 2004] Zonca, A. (2004). Modelling and optimisation of space mission prephase A design process in a concurrent engineering environment through a decision-making software based on expert systems theory.