HiSST: 3rd International Conference on High-Speed Vehicle Science Technology
14 -19 April 2024, Busan, Korea

Data Models for the Probabilistic Design of the Thermal Protection System of a Reusable Launch Vehicle Stage

Aaron D. Koch[1], Jascha Wilken[2], Marko Alder[3]

Abstract

Early design phases significantly impact a system's life-cycle costs, yet they are fraught with large uncertainties. Hence, it is important to incorporate uncertainties in preliminary design activities. However, sampling the uncertain design space instead of analyzing a single system entails a considerable computational cost. It also demands a high degree of automation, with data models being a crucial component. Consolidating all data describing a system in a structured way in a single source helps to streamline processes. This applies to the technical system itself as well as the description of the probabilistic study. The use of the Extensible Markup Language (XML) along with XML Schema Definition (XSD), complemented by libraries written in C++ with Python bindings via Boost.Python, has proven to be effective for implementing data models. This paper demonstrates the application of such data models through the sizing of a thermal protection system (TPS) of a reusable launch vehicle stage. The results indicate that a probabilistic design of a TPS can lead to a reduction in the required material thickness compared to a worst-case scenario.

Keywords: *probabilistic design, uncertainty quantification, data model, reusable space transportation system, thermal protection system*

1. Introduction

Blair et. al. estimate that at least 80% of a launch vehicle's life-cycle cost is determined in the conceptual design phase [1]. At the same time, the highest level of uncertainty is present in the earliest stages, i.e., the conceptual and the preliminary design phase. Essential methodological tools during these phases are trade and sensitivity studies. Incorporating the effects of uncertainties into conceptual design studies can improve the robustness of the results and thus support the decision-making process when comparing different designs. Instead of just evaluating the nominal (deterministic) performance of different vehicles, a system designer can also consider the level of uncertainty surrounding a particular performance metric. During preliminary design, incorporating uncertainty can help reduce the required margins of safety. Rather than relying on worst-case assumptions, a probabilistic approach allows for a more realistic combination of various factors, especially those that affect the overall mass of a reusable launch vehicle (RLV). If the payload into a specific orbit is given as a requirement and a specific RLV design underperforms, then this can be rectified by scaling up the system. This, however, leads to an overall heavier and thus more expensive system.

The state of the art in Uncertainty Quantification (UQ) in (reusable) launch vehicle conceptual design encompasses a variety of methodologies and applications aimed at addressing the inherent uncertainties in the design process. The fundamental task in uncertainty quantification is to propagate the uncertainties in the system input variables and the uncertainties imposed in the disciplinary models to the system outputs (i.e., results). While input uncertainties can be classified into epistemic (i.e., lack of knowledge which can be reduced by additional knowledge) and aleatory uncertainty (i.e., cannot be

[1] *German Aerospace Center (DLR), Robert-Hooke-Str. 7, 28359 Bremen, Germany, aaron.koch@dlr.de*
[2] *German Aerospace Center (DLR), Robert-Hooke-Str. 7, 28359 Bremen, Germany, jascha.wilken@dlr.de*
[3] *German Aerospace Center (DLR), Hein-Saß-Weg 22, 21129 Hamburg, Germany, marko.alder@dlr.de*

reduced), most of the studies in the context of conceptual and preliminary design focus on the first category [2,3]. This is because this phase of the design process is dominated by rather large assumptions on performance characteristics and technology assumptions for a future entry into service of the vehicle. Furthermore, decision making (e.g., in which discipline to invest resources) requires ranking the importance of inputs with respect to output uncertainties through sensitivity analysis, where Sobol indices have proven to be a valuable indication of the sensitivity of the global design space. Within the probability formalism, techniques based on the Monte Carlo method are typically used in the literature because they are easy to implement in multidisciplinary design tasks (disciplinary models can be considered as black boxes) and the propagated output uncertainty can be characterized by all statistical moments, such as mean, variance, skewness, or kurtosis. However, nearly all studies recently found in the literature aim at reducing the computational cost of classical Monte-Carlo approaches. Thunnissen et al. [4] advance the latter through subset simulation, a modified version of an efficient probabilistic method applied to conceptual design problems such as estimating fuel mass for attitude control, providing computational efficiency especially for risk-averse decision makers. Brevault et al. [5] apply surrogate models such as reduced-order models and spectral methods to the trajectory optimization of two-stage-to-orbit launch vehicles. Smith [6] employ reliability-based design optimization (RBDO) of a reusable launch vehicle using sequential optimization and reliability assessment (SORA) as proposed by Du and Chen [7].

While mathematical methods exist to address these issues [2], they are not yet standard practice for launch vehicle design and analysis. Their broader adoption is hindered by the additional effort they entail. This effort is twofold. Firstly, there is a substantial computational burden because a single simulation is replaced by a sampling of the uncertain design space, often involving thousands of system evaluations. Secondly, designers need to acquire additional knowledge regarding methods and tools for defining, executing, and evaluating probabilistic studies.

The German Aerospace Center's (DLR) project for the probabilistic technology assessment of complex transportation systems (PROTEKT) aims at the development of a framework for the analysis and propagation of uncertainties in multidisciplinary design processes, which essentially consists of three pillars: (1) the modeling of uncertainties and the necessary management of uncertainty data, (2) the propagation of uncertainties in the multidisciplinary design process, and (3) the assessment, analysis and visualization of the results.

This study focuses on the first pillar, whereby a probabilistic approach based on Monte-Carlo (MC) techniques was initially chosen as the method for quantifying uncertainties. The associated sampling-based approach not only requires high computing capacities, but also calls for structured and standardized management of the data generated in the design process. Section 2 present an approach to overcoming this challenge by means of a standardized data model and an associated software library. The benefits of such an approach are demonstrated in Section 3 by performing a preliminary sizing of a thermal protection system (TPS) of a reusable launch vehicle stage. Section 4 then concludes with a summary of the results and outlook on future research.

2. Data Management in Uncertainty Quantification

The strategy for managing UQ data in probabilistic design studies is based on the following basic assumptions: (1) MC-based design processes generate a large number of samples, which requires efficient processing and memory management, (2) disciplinary analysis modules are regarded as black boxes, which requires non-intrinsic propagation methods, and (3) a higher-level discipline (aerospace, aviation, transportation, etc.) employs a domain-specific data model, which should not be modified in the context of UQ and which can be referenced to supplement probabilistic information. The following sections first discuss the use of domain-specific data models in the context of multidisciplinary design processes. Then, the use of another data model for uncertainty data is introduced using XML Schema Definition (XSD). Finally, the implementation of a C++ software library is presented as a means of efficient and robust data handling.

2.1. Domain-specific Data Models in Deterministic Design Process

As a prerequisite for the results presented in this study, classical deterministic design processes are first introduced. The task of designing complex transportation systems, such as reusable launch vehicles, is to find a consistent (and ideally an optimal) solution to a multidisciplinary system of equations. This is called Multi-disciplinary analysis (and optimization, MDA(O)). Experts from different disciplines contribute a variety of analysis modules, which are usually considered as black-box modules due to the use of different solution algorithms and software architectures, and therefore need to be connected via partitioned coupling algorithms. Previous studies by the authors of this paper have shown that such coupling benefits greatly from the application of a standardized data transfer [8,9], where a parametric description of the actual transport system serves as a central source of truth. This is accomplished by a hierarchically structured data modelling approach via Extensible Markup Language (XML) and XML Schema Definition (XSD) [10], as it is already well established for domain-specific data models CLAVA [9] and CPACS [11] for aerospace and aeronautical applications, respectively. In this context, however, it should be emphasized that such an implementation is independent of the actual data model, which could for example also be converted to RDF/SHACL, as the authors will show in a future publication.

The use case described in section 3 addresses the sizing of a thermal protection system (TPS). For this very specific case, the CLAVA data model contains on the one hand too many parameters and on the other hand some relevant parameters are missing. So, the goal within the PROTEKT project is to create a simplified domain-specific data model for TPS sizing. At the moment, the parameters for this use case are directly stored in the Python scripts.

2.2. UQ Data Model in a Probabilistic Design Process

Figure 1 shows a probabilistic design and analysis process in the form of an Extended Design Structure Matrix (XDSM), which is comprised of three distinct steps: (1) a sampling algorithm is used to generate the probabilistic input factor sets $\mathbf{X}$ for the actual design process, which is composed of (2) at least one disciplinary black-box model (usually more) solving for the solutions y_i of deterministic equations (in this study this step is substituted with a complex design process as shown in section 0). In a third step (3) the results are collected and analysed with respect to its statistical moments or sensitivity measures. A detailed description of this process is given in [12].

A data management strategy should be able to map these three steps, considering that the disciplinary modules should be treated non-intrusively and the domain-specific data model, s. section 2.1, should be left unchanged so as not to interfere with its use in deterministic analysis.

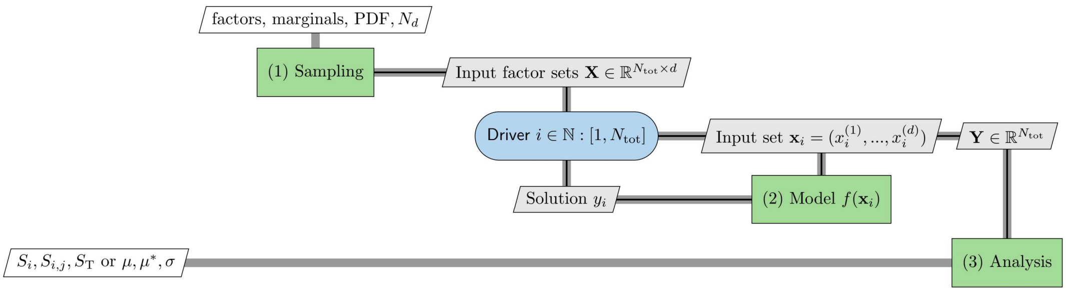

Figure 1. XDSM representation of the data exchange in a probabilistic design process

Figure 2 shows the basic structure of the data model in terms of an XSD diagram. This type of representation visualizes the hierarchical dependency of the corresponding elements, with solid and dashed frames indicating mandatory and optional elements, respectively, while deviating occurrences are represented by a stacked block symbol with an indication of the minimum and maximum number of occurrence (e.g. from 1 to infinity for "study").

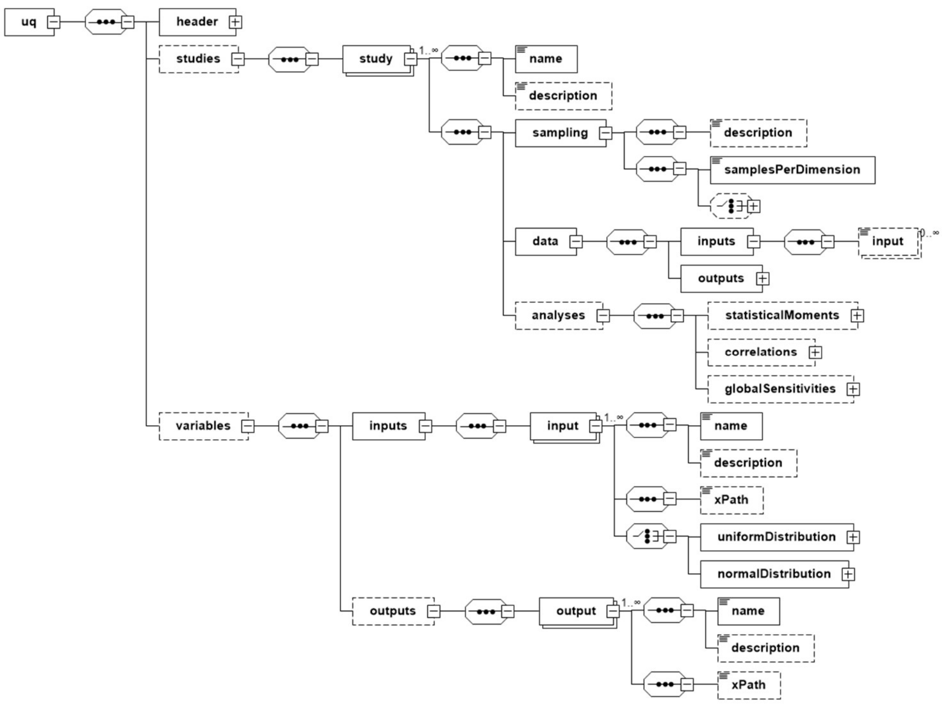

Figure 2. Hierarchical structure of the UQ data model

The first step of the probabilistic design process (see Figure 1) is represented by a *variables* element, which allows to pre-define the *inputs* and *outputs* of the process in terms of a *name*, *description*, and a reference to the corresponding domain-specific variables. The latter is currently implemented via *xPath* [13], but could in future also be extended by other means of references, such as indices for cells and rows in csv tables. This approach allows to enrich the domain-specific data by uncertainty information, without modifying it (i.e., keeping it independent form the research in UQ). Input variables (referred to as $\mathbf{factores}$ in Figure 2) are furthermore specified by a probability density function ($\mathbf{PDF}$, including $\mathbf{marginals}$), which is currently covered by a parametrization of uniform and (truncated) normal distributions. This list can easily be extended by further PDFs after the first testing phase is successful.

Pre-defining the variables has the advantage that they can be reused in different studies, for example to compare different sensitivity analysis methods or to perform convergence studies where the sampling size is varied. For this purpose, each study carries specific sampling information (e.g., number of *samplesPerDimension* and meta-parameters for selected sampling algorithms), as well as input and output data (i.e., $\mathbf{x}_i$ and $\boldsymbol{y}_i$) of the domain-specific models shown as green component (2) in Figure 2.

Finally, the third component (3) in Figure 2 comprised the analysis of the uncertainty data $\mathbf{Y}$. This is accomplished by a dedicated analyses element containing information on *statisticalMoments*, *correlations* of variables, and *globalSensitivity* measures, such as Sobol, Morris and Fast sensitivity indices (see [12] for further details).

2.3. Implementation of a C++ software library

Creating a standardized, hierarchically structured data model like in Figure 2 is only the first step. The next step is to provide a common programming library for simplifying data access. The approach originally used for the CLAVA data model was to create a library using C++ and provide Python bindings using Boost.Python [14]. This hybrid approach combines the speed of C++ with the user-friendliness of Python [15]. This code base has been reused within PROTEKT to develop the programming library for the UQ data model. Reading data from XML files and writing data to XML files is performed via the pugixml C++ library [16].

Figure 3 shows a simple Python code example for importing the library *uq*, creating a model, and then specifying the domain specific file, here *clava.xml*. Next, an input variable, named *radius*, is defined and an xPath pointing towards an entry in the CLAVA model is specified. In the same manner the output variable *payload* is added. A study called *mcs1* with 100 samples using the Latin Hypercube sampling is created. The two previously defined variables are then added to this study. The steps that would follow are the generation of random samples, the workflow execution for each sample and the reading of the results for each sample. Once the entire process has finished the model can be written to an XML file for storage.

```
import uq
model = uq.Model()
header.domainFile = 'file:///clava.xml'
input = model.variables.add_input('radius')
input.xPath = "/clava/rockets/rocket[@id='rocketExample']/geometry/radius"
input.distribution = uq.UniformDistribution(2.4, 2.6)
output = model.variables.add_output('payload')
output.xPath = "/clava/trajectories/trajectory[@id='trajectoryAscent']/payload"
study = model.add_study('mcs1')
study.sampling.samplesPerDimension = 100
study.sampling.method = 'LatinHypercube'
study.add_input('radius')
study.add_output('payload')
# Create random samples, run the workflow for each, read outputs
model.write('model.xml')
```

Figure 3. Simple Python code example for creating a UQ data model using the programming library

Using the above example for illustration, custom complex XSD types like *study* are implemented in C++ as classes. All classes are derived from a common Base class that provides the basic functionality for reading data from XML and writing data to XML. In contrast, native XSD types like *double*, used e.g. for radius, are implemented by matching them with native C++ types, *double* in this case. An important aspect of data models are references to other objects. For example, the study *mcs1* contains a reference to the definition of input variable *radius*. The node *variables*, s. Figure 2, contains a sequence of inputs and outputs. What is not shown in the figure is that each input and each output has an XML attribute named *id* of type xsd:ID. It used to reference them in the study, where each input and output has an attribute *idref* of type xsd:IDREF. This referencing mechanism is realized in C++ using shared pointers.

In addition to implementing the schema as defined in XSD, the library offers additional functionality for often used tasks. The example in Figure 3 uses a domain specific file. The library offers the function Study::realize_sample, which will automatically create a copy of the domain specific file with the values that were generated for the inputs belonging to a single sample. Once the respective design workflow has been executed for the sample, the values for the study outputs can be retrieved from the XML file using the function Study::load_sample.

3. Use Case

A full design process for a launch vehicle, necessarily includes the assessment of many subsystems which are tightly coupled. In order to keep the number of degrees of freedom to a manageable level and avoid time consuming iterations, for this use case a portion of the workflow was singled out that directly affected the design of the TPS system.

For now, the use case contains the following analysis steps:

- Assessment of trajectory performance
- Derivation of an aerothermal database
- TPS sizing for a given wing profile

Given the computational cost of repeatedly sampling the toolchain, rapid methods are required to assess the effect of the uncertainties on the individual subsystems.

Instead of a full optimization the descent trajectory is controlled with a PD controller that ensures that the maximum permissible lateral acceleration of 2 g is not exceeded. After the initial deceleration a constant flight path angle of -1° is targeted.

The aerodynamic and aerothermodynamic properties along the trajectory are assessed with the DLR tool HOTOSE [17] which can rapidly estimate the values for hypersonic flow with surface inclination methods.

Finally, the local TPS thickness is sized with the DLR tool top3, which integrates the thermal response of the TPS in one dimension and subsequently varies the insulation thickness until the temperature limits of the underlying structure are not exceeded.

Within these three steps a number of uncertainties are considered:

- Aerodynamic reference area: Within the trajectory integration the reference area is used to scale the forces resulting from the aerodynamic coefficients. This uncertainty essentially models the uncertainty in the magnitude of the aerodynamic forces during reentry.
- Reentry mass: This is the initial mass at the start of the reentry trajectory integration. This uncertainty can stem from the assumptions made for the vehicles dry mass during conceptual design but can also appear during the actual flight, when for example it is uncertain how much of the residual and reserve propellants still remain in the stage. Another possible source are variations in the production of each individual vehicle.
- Initial conditions for reentry: Depending on external factors during ascent and the performance of the GNC systems, the stage separation states can vary from mission to mission. Herein uncertainties in the initial flight path angle, altitude and velocity are considered. As can be expected, the initial conditions have a major impact on the trajectory including the thermal loads experienced during the reentry.
- Transition Reynolds number: The transition from laminar to turbulent flow has a significant impact on the local heat flux and its predication is notoriously uncertain. Local flow perturbations can trigger the transition early and lead to local hot spots.
- TPS panel emissivity: The majority of the heat flux that enters the TPS is radiated outward, the thermal insulation assures that only a small fraction arrives at the vehicles main structure. Thus, the emissivity of the TPS surface panels have an impact on the necessary insulation thickness.
- Heat flux uncertainty: This parameter accounts for the general uncertainty in deriving the local heat loads numerically.
- Heat transfer coefficient (HTC) and ambient temperature during non-hypersonic flight: For the estimation of the heat loads, the DLR tool HOTSOSE [17] is used which estimates the aerodynamic and aerothermodynamic properties with the surface inclination method. This approach works well for hypersonic speeds but is not applicable for lower supersonic and subsonic speeds. While the highest heat loads are present during the hypersonic phases, the slower phases also have to be modelled in order to correctly account for cooling of the structures. Within this study a fixed heat transfer coefficient and ambient temperature are assumed and uncertainties applied to them.

In this use case all uncertainties were considered as uniform variations around the reference value. The specific relative variation and reference values are given in Table 1. The values are chosen arbitrary and do not necessarily reflect relevant cases. The variation for the velocity is chosen very low, because otherwise the impact of the initial velocity dominates almost everything else.

Table 1. Considered uncertainties and their variation and reference value. All sampled uniformly.

Uncertainty	Relative variation	Reference value
Surface area	[0.9,1.1]	461 m²
Reentry mass		229657 kg
Initial velocity	[0.99,1.01]	3772 m/s
Initial flight path angle	[0.9,1.1]	5.06°
Initial altitude		62.26 km
Transition Reynolds number	[1,50]	1000
TPS Panel emissivity	[0.9,1.1]	0.81
Heat flux		Varies for each sample
Heat transfer coefficient for non-hypersonic flight		20 W/(K*m²)
Ambient temperature for non-hypersonic flight		200 K

3.1. Results

In total the uncertainty space was sampled 11264 times for this exemplary use case. Figure 4 shows the resulting reentry trajectories. The therein showed estimation for stagnation point is derived for a reference nose radius of 0.5m with a modified chapman equation:

$$\dot{q} = 20254.4\ \mathrm{W/cm^2} \cdot \sqrt{\frac{\rho}{\rho_R}\frac{R_{N,r}}{R_{,N}}}\left(\frac{v}{v_r}\right)^{3.05} \tag{1}$$

It can be seen that the selected uncertainty distribution lead to a fairly broad range of initial condition and subsequently also heat fluxes encountered throughout the trajectory.

For each of the shown trajectory samples an aerothermal database is derived. As the local heat flux depends on the wall temperature, the calculations are repeated for 4 different wall temperatures in order to be able to interpolate within the thermal simulation for the TPS design. For this use case the analysis is done for the wing profile at the root of the wing of the SpaceLiner 8 Booster stage, as described in [18]. In the following only the bottom side of the wing is considered, as it sees the larger thermal loads.

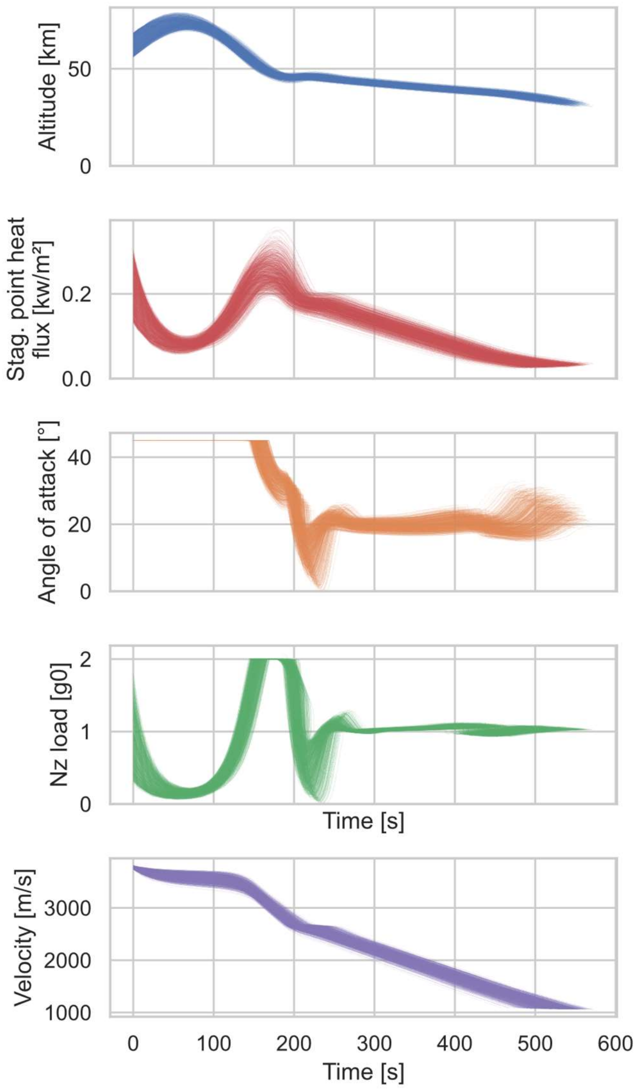

Figure 4. Effect of uncertain separation conditions on re-entry trajectory

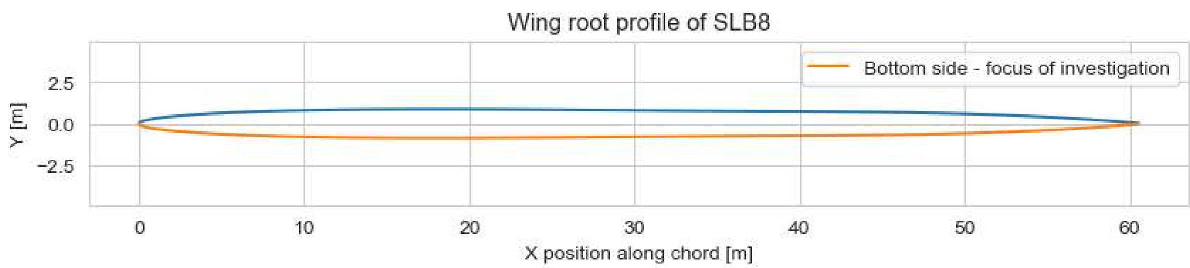

Figure 5. Wing root profile of SLB8

Figure 5 shows the wing profile and Figure 6 shows the heat flux along the x-axis of this profile resulting from the trajectory variations. In addition, the uncertainty of the transition from laminar to turbulent transition is also considered. The transition can be seen in the sudden jumps in heat flux between two trajectory points. The variations seen at ~ 220 s into the trajectory are likely caused by a temporary reduction in the angle of attack, that can also be seen in Figure 4, which leads to temporary lower heat flux on the rearward portions of the wing, as they are less exposed to the incoming flow.

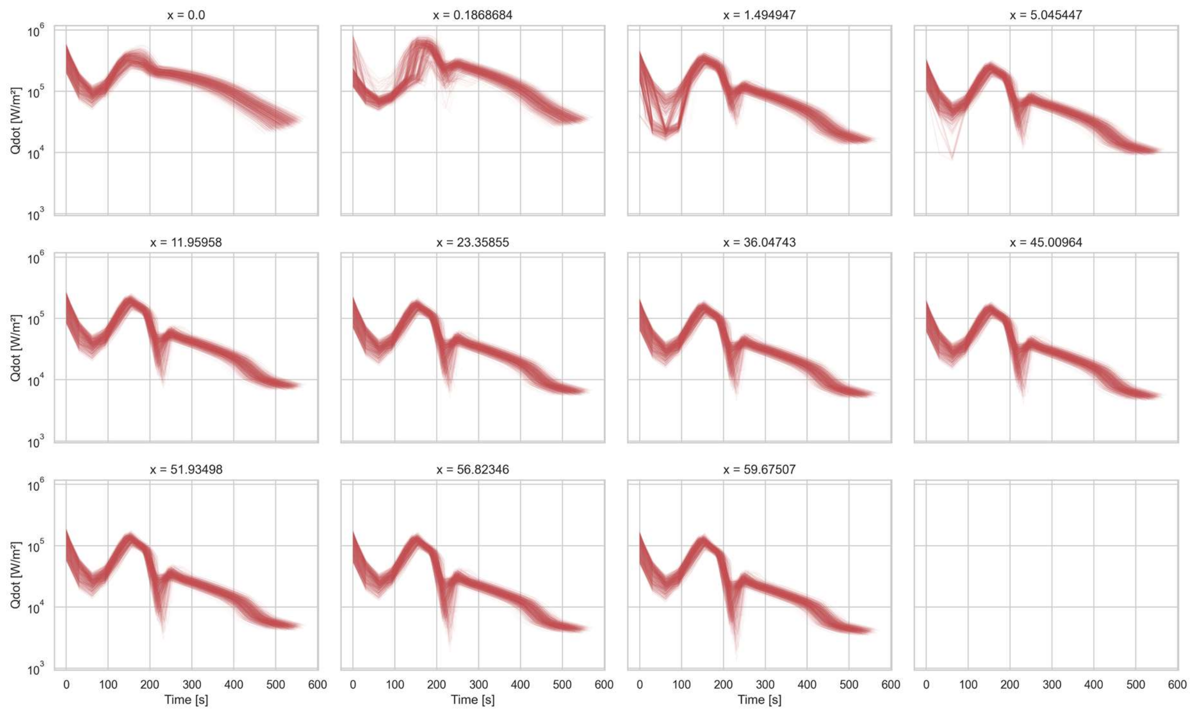

Figure 6. Variability of heat flux over time and position along wing profile for a wall temperature of 400 K

The final objective of this use case was the determination of the required TPS thickness to keep the temperature of the structure below 400 K. In this case, an idealized minimal insulation thickness was assessed along the wing chord. In reality, only a limited number of different thicknesses would be used in order to reduce production and maintenance effort. Figure 7 shows the resulting minimal thickness of the insulation layer of the chord of the wing root.

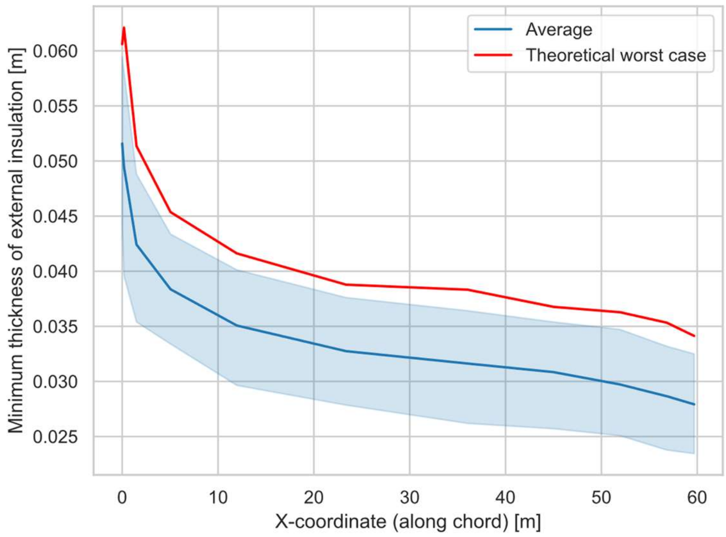

Figure 7. Ideal TPS insulation thickness of wing chord. Average values and range of all 11264 samples are shown alongside the theoretical worst case

As expected, the TPS needs to be thickest nearest to the stagnation point which is in line with the heat load distribution.

For the vehicle design the local TPS thickness is of less importance than the total TPS mass, in order to assess that, the total insulation cross section along the wing profile was calculated and the resulting values are shown in Figure 8.

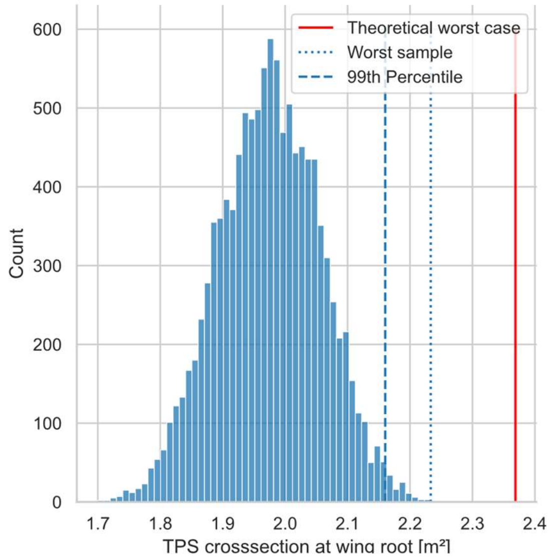

Figure 8. TPS cross section surface for all 11264 samples and theoretical worst case

In order to underline the motivation for assessing the system performance within an uncertainty framework, Figure 7 and Figure 8 also show the theoretically possible worst case. In that case every single uncertain parameter was chosen to arrive at the highest TPS cross section possible. This value corresponds to a conservative design approach.

3.2. Sensitivity analysis

In total ten uncertain parameters were modelled in the use case given above. In order to determine the effect on the outputs (both the local TPS thickness as well as the total cross-section area) a Sobol Analysis [19] was conducted.

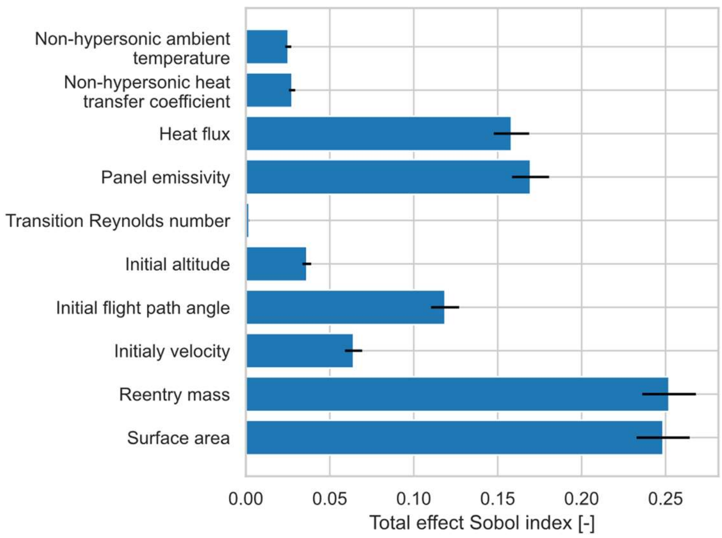

Figure 9. Total effect sensitivity indices with regard to TPS cross section. Errorbars indicate confidence interval.

Figure 9 shows the total effects indices with regard to the TPS cross-section along the wing profile. It is clearly apparent that the uncertainty in the reentry mass and the reference surface area play a major role. Since both effectively alter the ratio of vehicle mass to aerodynamic forces it also is credible that the magnitude of the impact is similar. Within this use case, the uncertainty in the ambient conditions and heat transfer coefficient in subsonic flight appear to play only a minor role, as well as the transition Reynolds number.

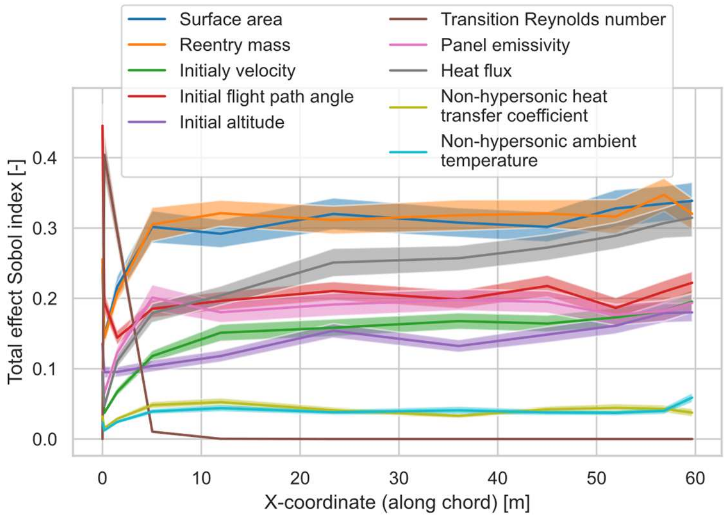

Figure 10. Total effect sensitivity indices for local TPS thickness for different point along the wing profile including confidence intervals

Figure 10 also shows the total effect indices but for the local TPS thickness over the wing profile. Herein some interesting contrasts to the sensitivity of the global results appear. For most of the wing profile (starting from ~5 m being the leading edge) the sensitives are very similar to the result for the total cross section surface. However, in the first portion of the wing profile, other factors have a large impact on the necessary TPS thickness. Especially, the transition Reynolds number has the largest effect on the sizing of the TPS right behind the leading edge. This results from the laminar to turbulent transition occurring in this area at the periods of the trajectory that exhibit the high heat loads. At the leading edge, the flow is always laminar by definition, and some meters behind the leading edge the flow is always turbulent (at the relevant points of the trajectory). The area in between is subsequently sensitive to uncertainties in the transition Reynolds number.

Even though this effect is critical for certain parts of the geometry, its effect on the total TPS cross section is minor, as shown in Figure 9 as the majority of the wing profile is unaffected by this uncertainty. This result is specific to the trajectory flown and has to be re-assessed for other use cases.

4. Conclusion

The use case addressing the TPS sizing of a reusable stage demonstrates that a mass reduction is possible when employing a probabilistic approach compared to a worst-case design. Standardizing the approach to probabilistic design greatly facilitates such analyses. Data models play a pivotal role in this endeavor, as handling thousands of simulations necessitates automated data management. The solution presented here is based on XML schema and a C++/Python library for creating easily accessible data models. The implementation of the TPS related parameters into a custom domain-specific data model is still ongoing. The long-term goal is to perform an uncertainty-based design using a full launch vehicle model described via the CLAVA model. In addition, Python has proven to be a valuable environment for integrating different engineering tasks. This encompasses creating workflows that execute several engineering tools, as well as reading data from and writing data to various sources, and finally analyzing the probabilistic results.

References

1. Blair, J.C., Ryan, R.S., Schutzenhofer, L.A., Humphries, W.R.: Launch vehicle design process: characterization, technical integration, and lessons learned. NASA, TP-2001-210992 (2001)

2. Yao, W., Chen, X., Luo, W., van Tooren, M., Guo, J.: Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles. Progress in Aerospace Sciences, 47(6), 450-479 (2011)
3. King, J.M., Grandhi, R.V., Benanzer, T.W.: Quantification of epistemic uncertainty in re-usable launch vehicle aero-elastic design. Engineering Optimization, 44(4), 489-504 (2012)
4. Thunnissen, D.P., Au, S.K., Swenka, E.R.: Uncertainty quantification in conceptual design via an advanced Monte Carlo method. Journal of Aerospace Computing, Information, and Communication, 4, 902-917 (2007)
5. Brevault, L., Balesdent, M., Morio, J.: Aerospace system analysis and optimization under uncertainty. Springer, Cham (2020)
6. Smith, N., Mahadevan, S.: Integrating system-level and component-level designs under uncertainty. Journal of Spacecrafts and Rockets, 42(4) (2005)
7. Du, X., Chen, W.: Sequential optimization and reliability assessment method for efficient probabilistic design. ASME 2002 Design Engineering Technical Conferences and Computer and Information in Engineering Conference, Montreal, Canada, 29 September – 2 October, 2002
8. Alder, M., Skopnik, A.: Development of as software library for performant and consistent CPACS data processing. Deutscher Luft- und Raumfahrtkongress, 27-29 September 2022, Dresden
9. Fischer, P.M., Deshmukh, M., Koch, A., Mischke, R., Martelo Gomez, A., Schreiber, A., Gerndt, A.: Enabling a conceptual data model and workflow integration environment for concurrent launch vehicle analysis. 69th International Astronautical Congress (IAC), 1-5 October 2018, Bremen, Germany
10. Fallside, D.C., Walmsley, P. (eds.): XML Schema Part 0: Primer Second Edition. World Wide Web Consortium (W3C). https://www.w3.org/TR/xmlschema-0/ (2004). Accessed 29 September 2023
11. Alder, M., Moerland, E., Jepsen, J., Nagel, B.: Recent advances in establishing a common language for aircraft design with CPACS. Aerospace Europe Conference, 25-28 February 2020, Bordeaux, Frankreich
12. Alder, M., Ahmed, T., Fröhler, B., Skopnik, A: Assessment of techniques for global sensitivity analyses in conceptual aircraft design. AIAA Aviation Forum 2023, San Diego, USA
13. Clark, J., DeRose, S. (eds.): XML Path Language (XPath). Version 1.0. World Wide Web Consortium (W3C). https://www.w3.org/TR/1999/REC-xpath-19991116/ (1999). Accessed 5 April 2024
14. Abrahams, D., Seefeld, S.: Boost.Python. https://www.boost.org/doc/libs/1_83_0/libs/python/doc/html/index.html. Accessed 29 September 2023
15. Abrahams, D., Grosse-Kunstleve, R.W.: Building hybrid systems with Boost.Python. C/C++ Users J. 21(7), 29-36 (2003)
16. Kapoulkine, A.: pugixml. https://pugixml.org/. Accessed 12 May 2023
17. Reisch, U., Streit, T.: Surface inclination and heat transfer methods for reacting hypersonic flow in thermochemical equilibrium. New results in numerical and experimental fluid mechanics: Contributions to the 10th AG STAB/DGLR Symposium Braunschweig, Germany (1996)
18. Sippel, M., Stappert, S., Bayrak, Y.M., Bussler, L., Callsen, S.: Systematic assessment of SpaceLiner passenger cabin emergency separation using multibody simulations, 2nd HiSST-conference, Bruges, September 2022
19. Sobol, I. M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Mathematics and computers in simulation, 55(1-3), 271-280 (2001)