# Multidisciplinary Design Optimization of a Re-entry Vehicle for the Earth Return Services from the ISS

Sweety Pate

Master of Science Thesis

Delft University of Technology

Astrodynamics and Space Missions

# Multidisciplinary Design Optimization of a Re-entry Vehicle for the Earth Return Services from the ISS

Master of Science Thesis

For the degree of Master of Science in Aerospace Engineering at Delft University of Technology

Sweety Pate

January 31, 2017

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology



The image on the cover-page indicates the motivation behind this thesis, where the conceptual aircraft and spacecraft designs indicate the previous work performed by author, Furthermore, the combination of different trajectories and shapes of IXV like configuration, indicates the trajectory and the shape optimization of the vehicle using the concurrent engineering approach. Image courtesy to DLR and ESA.

# Preface

This thesis report describes the thesis research on the topic, the system study of an expandable/reusable re-entry vehicle designs, which enables a frequent space transportation for the Earth-return service from the International Space Station (ISS). The thesis is conducted at the Department of Aeromechanics and Propulsion of Thales Alenia Space, Italia (TASI), and this report is written to discuss the work done to answers the research objectives.

The interest towards the field of conceptual aircraft and spacecraft designing started in the early years of my Bachelors of Technology (B.Tech) in Aerospace Engineering. I actively participated and successfully won awards in the international competitions organized by National Aeronautics and Space Administration (NASA) and American Institute of Aeronautics and Astronautics (AIAA) dedicated to designing an innovative conceptual aircraft and spacecraft respectively. During those times, I followed a traditional design approach and if a common platform for integration of all design disciplines was available, the designs would have been an optimal configuration.

Later, I realized that there are such integrated multi-disciplinary environment and was a current research topic at the Department of Integrated Aircraft Design at German Aerospace Center (DLR), Hamburg. To explore my curiosity to implement such methodology in my research projects I joined DLR, where I gained experience in the feasibility study of Blended Wing Body (BWB) aircraft design using an integrated multi-disciplinary environment. This experience inspired me to implement a similar approach in the field of spacecraft designs. There are few research papers available on the application of the similar methodology for spacecraft design, but it is are not yet implemented in most of the space industries. Thales Alenia Space is now looking forward to adapting the concurrent engineering approach that can enable one to perform the feasibility study of a re-entry vehicle in an efficient way considering the cost and time constraints. Thus, this thesis answers my curiosity to develop a better methodology for the preliminary study of a re-entry vehicle design.

I would like to thank the Head of the Department, Dr.ir. Cosimo Chiarelli and the engineers team of Vincenzo Mareschi, Raffaele Aulisio and Martins Sudars for their valuable guidance, stimulating input and support. I would also like to thank Walter Cugno, Eugenio Gargioli and Cesare Lobascio for arranging my master thesis position at TASI and giving me this golden opportunity to learn, contribute and excel with the global leaders of aerospace. I would like to thank the whole team of TASI for the pleasant working environment. I would also like to thank Dr.ir. Erwin Mooij (Delft University of Technology) for his valuable guidance and evaluation of the work done. All these would have not been possible, without the support of my family, friends and the people I met during this complete journey, who continuously inspired and supported me.

# Summary


The ambition and curiosity of humans drive advancements in space exploration technologies and to accomplish this dream, we need safe space transportation. With the development of the International Space Station (ISS) in 1998, which allowed for scientific experiments in micro-gravity and space environment, it became necessary to have an efficient cargo retrieval system. Thus, most of the space industries are looking forward to developing new spacecraft designs that can enable frequent transportation of the cargo from the ISS to the Earth. To perform the feasibility study of a complex engineering system like re-entry vehicle using a traditional approach is inefficient and consumes time and resources. Concurrent Engineering (CE) approach has been increasingly used and implemented by enterprise environment in the space industry. This approach ensures that the customer needs are satisfied with the required quality, promoting a reduction of costs and development time.

During the conceptual design phase, despite of having efficient models for each discipline, we lack the specific and standardized design analysis techniques to be used by engineers and designer. The initial design process of the Intermediate eXperimental Vehicle (IXV) configuration followed a traditional approach, where engineers worked sequentially passing the design from engineer to engineer and then followed by iterations. This process eventually resulted to be an expensive methodology in-terms of cost, time and resources. Currently, there is a need to provide structured and common design methods involving all the design disciplines at the same time along with engineering data-exchange between the experts. This new design process can be useful for preliminary design analysis of the on-going Spacerider re-entry vehicle, which is derived from IXV with a payload carrying capacity. Thus, the main objective of the thesis is to derive an optimized re-entry vehicle similar to the Intermediate eXperimental Vehicle (IXV) configuration during the preliminary design phase, where the vehicle is designed for a specific mission scenario and system requirements using a design process developed within an open source Dakota framework. To solve this industrial problem within the work of thesis the main research question is framed as follows:

*"To what extent can an optimal re-entry vehicle similar to the IXV configuration can be developed in the preliminary design phase using an integrated design process, where the vehicle is designed for a specific mission scenario and system requirements?"*

To answer the main research question, an integrated design process to perform the Multidisciplinary Design Optimization (MDO) of a re-entry vehicle is required. Thus, the two other sub-goals are formulated as follows:

*"To perform the feasibility study of the re-entry vehicles using the MDO techniques, to what extent is it possible to integrate the design disciplines on a single platform and automate the design process within the Dakota framework?"*

*"How and to what extent can the design techniques available within the framework, assist the engineering team during the conceptual design of complex systems to obtain better, faster and eventually cheaper design process?"*

To achieve the sub-goals, an integrated design methodology within an open source Dakota framework is developed. Initially, a simple axis-symmetric Raduga capsule is considered to verify the design process, for the re-entry vehicle application and then extended to the lifting body vehicle similar to IXV. This approach allowed the user to reduce the complexities from the vehicle point of view and focus on improving the design process for the feasibility study of the re-entry vehicle application using the MDO techniques.

During the feasibility study of Raduga capsule, initially the sampling of the design space is performed by using Latin Hypercube Sampling (LHS) method. From these samples, the correlations between the design variables, cost functions and set of the design factors are derived using the Spearman's correlation formulation. From these results, the user can get the better understanding of the range in the output for a particular domain. Additionally, the user can explore the design space to determine, which inputs have the most influence on the results and how changes in the inputs change the results. This sensitivity analysis gives the insight of influence of design parameters on the results, and is considered as starting point for the optimization using the Multi Objective Genetic Algorithm (MOGA). Using this methodology, 30 optimized configurations are derived that satisfy the constraints and meets the objectives to maximize payload carrying capabilities. Based on the user's choice, the best optimized configuration is selected, which gave improvement of 4 kg payload mass capacity and 0.03 $m^3$ payload volume capacity, compared to the reference Raduga configuration.

Once, the design process is verified for a re-entry vehicle application using simple axis-symmetric configuration, it is then extended to trajectory optimization of a lifting body re-entry vehicle. This trajectory optimization is performed using the MOGA as optimization technique, for a fixed configuration of IXV. These results indicated that the derived optimized trajectory gives improved performance than the initial (open-loop simulation) as well as the nominal reference trajectory of IXV, with respect to the values of the cost functions considered for the trajectory simulation.

This step-by-step verification approach allowed to develop a relatively more generalized design process, that enables one to manage the inputs and the corresponding response function through a generalized interface template. The design disciplines are integrated on a single platform to automate the design process using this interface. User can intervene the process through the interfaces, which gives the possibilities of faster and efficient debugging capabilities. The architecture is based on object oriented approach, where plug and play of all the involved discipline tools is easily possible. The design process also allows the user to accept a black-box tool from a developer as well as it is extensible to any new features addition to the tool, which is most appealing, when developers cannot share their software details. Additionally, the in-build features provided by the Dakota framework allows the data management and visualization of the on-going process, with a flexibility to define a feasibility study strategy.

The main research question is answered by the results of the design case, where the geometry and the trajectory optimization is performed simultaneously for a lifting body re-entry vehicle. The design case configuration is a payload carrying re-entry vehicle, which is derived from the shape morphing of the baseline IXV configuration. As the design process allows easy plug and play, this tool is introduced in the workflow along with the tools dedicated for the aerodynamic database computation, trajectory simulation and aerothermodynamic database computation. Similar to the non-lifting body optimization (Raduga), samples are generated using the LHS method for the design case problem. Using these samples, the correlations between the input and output and set of the design factors are derived using the Spearman's correlation formulation. This sensitivity analysis provided an idea of the influencing design parameters and the range of the design parameters at which the optimized solution can be obtained. Considering this as a starting point, the optimization is performed using the MOGA, with the major objectives of minimizing the integral of the heat-load, maximizing the payload carrying capacity and the landing accuracy of the re-entry vehicle.

From the optimized solutions, two design cases are selected, which gives payload mass and payload volume higher than 450 kg and 0.71 m$^3$ respectively. Additionally, these design cases gives an integral of the heat load, which is less than 300 MJ/m$^2$ and landing accuracy is within 25 km. Two optimized design cases are selected, which gives the payload mass and payload volume capacity of 482 kg, 0.79 m$^3$ and 520 kg, 0.74 m$^3$ respectively. Furthermore, this design cases indicated an improvement in landing accuracy, where the vehicle with payload capacity of 482 kg reached the within the landing the accuracy of 0.4 km whereas the design case with payload mass capacity of 520 kg reached within 18 km. The results showed improved compared to the open-loop simulation, which indicates that using the developed design process a better solution can be derived.

This answers the main research question, where a optimized configuration similar to IXV is derived for the given mission scenario and systems requirements. As it is a multi-objective problem more than one solutions are derived and the user can select the configuration based on the requirements. The design case with payload mass capacity of 482 kg with landing accuracy of 0.4 km is considered as a final solution. This design case also gives better trajectory profile than the other selected design case. The results of the design cases also indicate a possibility to refine the results by separately performing the trajectory optimization of fixed configuration as fixed and vice-verse.

# Table of Contents

# Nomenclature

## Acronyms

| | |
|---|---|
| AAO | All-At-Once |
| ACO | Ant Colony Optimisation |
| AIO | All In Once |
| API | Application Programming Interface |
| ASI | Italian Space Agency |
| ATV | Automated Transfer Vehicles |
| BB | Box-Behnken Design |
| BDH | Block Diagonal Hessian |
| CCD | Central Composite Design |
| CCDev | Commercial Crew Development |
| CD | Concurrent Design |
| CDF | Concurrent Design Facility |
| CE | Concurrent Engineering |
| CEM | Concurrent Engineering Methodology |
| CFD | Computational Fluid Dynamics |
| CNES | French Space Agency |
| CO | Collaborative Optimization |
| COG | Center of Gravity |
| COM | Center of Mass |
| CoRADS | Code for Rapid Aerodynamic Database Synthesis |
| CST | Crew Space Transportation |
| DA | Disciplinary Analysis |
| DACE | Design and Analysis of Computer Experiments |
| DC | Disciplinary Coordinator |
| DDACE | Distributed Design and Analysis of Computer Experiments |
| DE | Disciplinary Evaluator |
| DLR | German Aerospace Center |
| DOE | Design of Experiments |
| DoE | Design of Experiments |
| DOF | Degree of Freedom |
| DUT | Delft University of Technology |
| EA | Evolutionary Algorithms |

| | |
|---|---|
| EIP | Entry Interface Point |
| ESA | European Space Agency |
| FCS | Flight Control System |
| FMF | Free Molecular Flow |
| FMST | Flight Mechanics Simulation Tool |
| GA | Genetic Algorithm |
| GGMET | Geometry Generation and Mass Estimation Tool |
| GNC | Guidance Navigation Control |
| GSFC | Goddard Space Flight Center |
| HD | Hierarchical Decomposition |
| IAD | Inflatable Aerodynamic Decelerator |
| IDC | Integrated Design Center |
| IDF | Individual Discipline Feasible |
| ISRO | Indian Space Research Organisation |
| ISS | International Space Station |
| IXV | Intermediate eXperimental Vehicle |
| JAXA | Japan Aerospace eXploration Agency |
| LaRC | Langley Research Center |
| LEO | Low Earth Orbit |
| LHS | Latin Hypercube Sampling |
| MA | Memetic Algorithm |
| MCV | Multi-purpose Crew Vehicle |
| MDF | Multi Discipline Feasible |
| MDO | Multi-disciplinary Design Optimization |
| MOGA | Multi Objective Genetic Algorithm |
| NASA | National Aeronautics and Space Administration |
| NBI | Normal Boundary Intersection |
| NC | Normal Constraint |
| ND | No-Decomposition |
| NHD | Non-Hierarchical Decomposition |
| PDC | Project Design Center |
| PSO | Particle Swarm Optimisation |
| RCS | Reaction Control System |
| SC | System Coordinator |
| SE | Systems Engineering |
| SFL | Shuffled Frog Leaping algorithm |
| SMMET | Shape Morphing and Mass Estimation Tool |
| SNC | Sierra Nevada Corporation |
| SO | System Optimizer |
| SQP | Sequential Quadratic Programming |
| SSS | Single SAND-SAND |
| SSTO | single-stage-to-orbit |
| TACOT | Theoretical Ablative Composite for Open Testing |
| TASI | Thales Alenia Space, Italia |
| TPS | Thermal Protection System |

# Latin Symbols

| | | |
|---|---|---|
| $C_L$ | Coefficient of lift | [ - ] |
| $C_p$ | Pressure coefficient | [ - ] |
| $C_{p,max}$ | Maximum value of pressure coefficient | [ - ] |
| $f_i$ | Disciplinary objectives | [ - ] |

| | | |
|---|---|---|
| $f1$ | First objective function | [ - ] |
| $f2$ | Second objective function | [ - ] |
| $g_i$ | Disciplinary constraints | [ - ] |
| $h_E$ | Entry altitude | [ m ] |
| $K$ | Ballistic Coefficient | [ kg/m$^2$] |
| $L1$ | Length of first section | [ m ] |
| $L2$ | Length of second section | [ m ] |
| $L3$ | Length of third section | [ m ] |
| $Lf1$ | Slant length of first section | [ m ] |
| $Lf2$ | Slant length of second section | [ m ] |
| $Lf3$ | Slant length of third section | [ m ] |
| $L_{total}$ | Total length of vehicle | [ m ] |
| M | Mach number | [ - ] |
| $M_\infty$ | Mach number tends to $\infty$ | [ - ] |
| $RB1$ | Base radius of first section | [ m ] |
| $RB2$ | Base radius of second section | [ m ] |
| $RB3$ | Base radius of third section | [ m ] |
| $RBo$ | Front cone nose radius | [ m ] |
| $r_i$ | Disciplinary residual | [ - ] |
| $RN$ | Radius of nose | [ m ] |
| $S$ | Surface area | [ m$^2$ ] |
| $s_i$ | Internal coupling variable | [ - ] |
| $S_{ref}$ | Reference surface area | [ m$^2$ ] |
| $s_s$ | System state variable | [ - ] |
| $TPSthk$ | TPS thickness of nose section | [ m ] |
| $TPSthk1$ | TPS thickness of first section | [ m ] |
| $TPSthk2$ | TPS thickness of second section | [ m ] |
| $TPSthk3$ | TPS thickness of third section | [ m ] |
| $V_E$ | Entry velocity | [ m/s ] |
| $W$ | Weight of vehicle | [ kg ] |
| $x1$ | First variable | [ - ] |
| $x2$ | Second variable | [ - ] |
| $X_{body}$ | Scaling factor of mainbody section along length | [ - ] |
| $X_{flap}$ | Scaling factor of flap section along length | [ - ] |
| $x_i$ | Disciplinary design variable | [ - ] |
| $X_{nose}$ | Scaling factor of nose section along length | [ - ] |
| $x_s$ | System design variables | [ - ] |
| $Y_{end}$ | Scaling factor of mainbody-flap intersection along width | [ - ] |
| $Y_{start}$ | Scaling factor of nose-mainbody intersection along width | [ - ] |
| $Z_{end}$ | Scaling factor of mainbody-flap intersection along height | [ - ] |
| $Z_{start}$ | Scaling factor of nose-mainbody intersection along height | [ - ] |

# Greek Symbols

| | | |
|---|---|---|
| $\alpha$ | Angle of attack | [ deg ] |
| $\alpha$ | Nose cone angle | [ deg ] |
| $\alpha_E$ | Entry angle-of-attack angle | [ deg ] |
| $\gamma_E$ | Entry flight-path angle | [ deg ] |
| $\gamma$ | Flight-path angle | [ deg ] |
| $\gamma$ | Adiabatic gas constant | [ - ] |
| $\gamma_E$ | Entry flight-path angle | [ deg ] |
| $\delta$ | Latitude | [ deg ] |

| | | |
|---|---|---|
| $\delta_E$ | Entry latitude | [ deg ] |
| $\delta_T$ | Target latitude | [ deg ] |
| $\sigma$ | Bank angle | [ deg ] |
| $\theta$ | Local deflection angle | [ deg ] |
| $\theta 1$ | Cone angle of first section | [ deg ] |
| $\theta 2$ | Cone angle of second section | [ deg ] |
| $\theta 3$ | Cone angle of third section | [ deg ] |
| $\tau$ | Longitude | [ deg ] |
| $\tau_E$ | Entry longitude | [ deg ] |
| $\tau_T$ | Target longitude | [ deg ] |
| $\chi$ | Heading angle | [ deg ] |
| $\chi_E$ | Entry heading angle | [ deg ] |

# Chapter 1

# Introduction

The ambition and curiosity of humans drive advancement in space exploration technologies and to accomplish this dream; we need safe space transportation. It is required to have an efficient cargo-retrieval system, to conduct scientific experiments in microgravity and space environment at International Space Station (ISS). After the end of the Space Shuttle program in 2011, the Russian Soyuz rocket became the prime provider of space transportation for the astronauts. The ISS supply missions use the Russian Progress spacecraft, European Automated Transfer Vehicles (ATV), Japanese Kounotori vehicles, SpaceX Dragon and Cygnus spacecraft, whereas currently, the Russian Soyuz spacecraft and SpaceX Dragon have the capability to return crew and cargo to the Earth. Thus, most of the space industries are looking forward to developing new spacecraft designs, that can enable frequent transportation of the cargo from the ISS to the Earth.

## 1-1 Background

This section describes the background information considered for this thesis. Subsection 1-1-1 highlights the traditional and the concurrent engineering approach to demonstrate the possibilities to integrate the design disciplines on a common platform for the preliminary study of a re-entry vehicle. Subsection 1-1-2 gives the mission heritage, which highlights some re-entry vehicles, that are developed from the benchmark designs and furthermore describes the design process used for multidisciplinary optimization of a re-entry vehicle. Furthermore, Subsection 1-1-3 gives the MDO architecture, which highlights the methods for problem decomposition, problem formulation, MDO techniques and sampling methods considered for this thesis.

### 1-1-1 Traditional and Concurrent Engineering Approach

Ridolfi (2013) describes that the life of the system can be divided into the set of phases. It starts with the conceptual design of the system and completes at the end of the system's operational life. At the end of each phase of the life-cycle, formal meetings are conducted, where the stakeholders discuss the achieved goals and agree upon the next design phase. Experience in space system design demonstrated that although most of the costs are expended in the advanced phases of the life cycle, a large amount of them is determined by the choices taken during the conceptual design as shown in Figure 1-1.

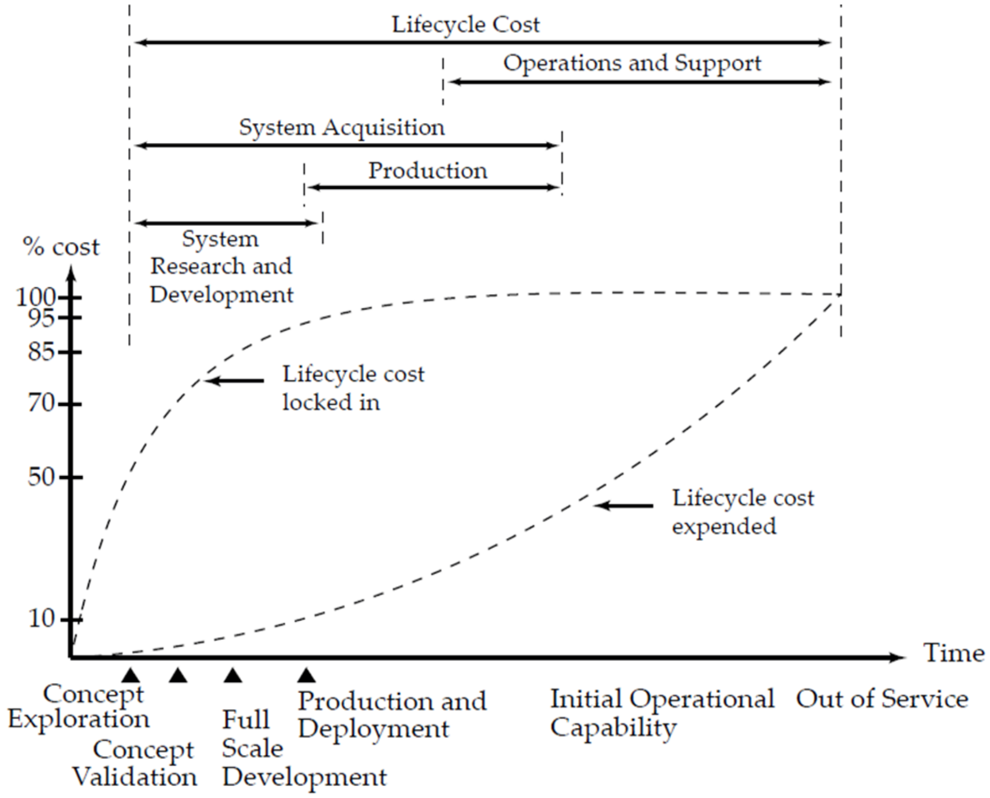



**Figure 1-1:** Percentage of costs locked-in and costs expended by life-cycle phase, (Larson, 1999)

The conceptual design phase is generally completed in a few weeks or months, where the main objective of the conceptual design is the definition of the mission to perform to satisfy the customer's requirements. This is achieved by evaluating multiple system design concepts and eventually, defining the system baseline with technology, programmatic and cost assessment. The level of detail increases enormously with the increase in the level of the design phase. Depending on the complexity of the system and the available resources, the advanced design phases may take months to years to fully design the system. Thus, if a poor design is chosen during the conceptual design phase, this will lead to a worse and expensive system at the end of the process.

To design a system, traditionally engineers work sequentially, that is one step at a time, passing the design from engineer to engineer as shown in Figure 1-2. Here, the design goes through several iterations until the requirements from all disciplines are satisfied. To perform the feasibility study of complex engineering systems, like re-entry vehicles using traditional approach is inefficient and consumes time and resources.

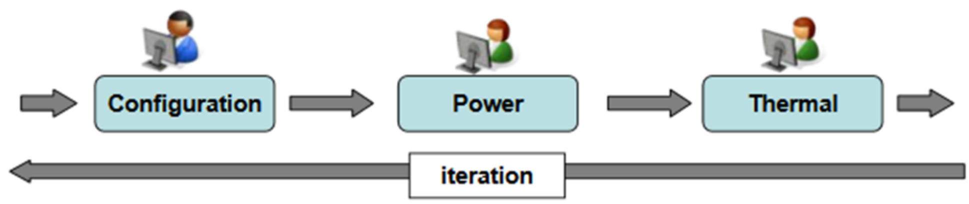



**Figure 1-2:** Traditional engineering approach, (Xu et al., 2016)

Systems Engineering (SE) approach has been increasingly used and implemented by enterprise environment in the space industry. This approach ensures that the customer needs are satisfied with the required quality, promoting a reduction of costs and development time. The space community is looking forward to using the concurrent design approach over the traditional design method as shown in Figure 1-3. From the figure, it can be seen that using concurrent design approach, all the aspects related to the spacecraft and the mission it will perform, are taken into account at the same time (concurrently) from the very beginning of the life cycle. All the technical discipline-experts, with risk, cost and programmatic engineers together with the customer are in constant communication between each other enabling the possibility to efficiently keep track of the system requirements and their evolution.

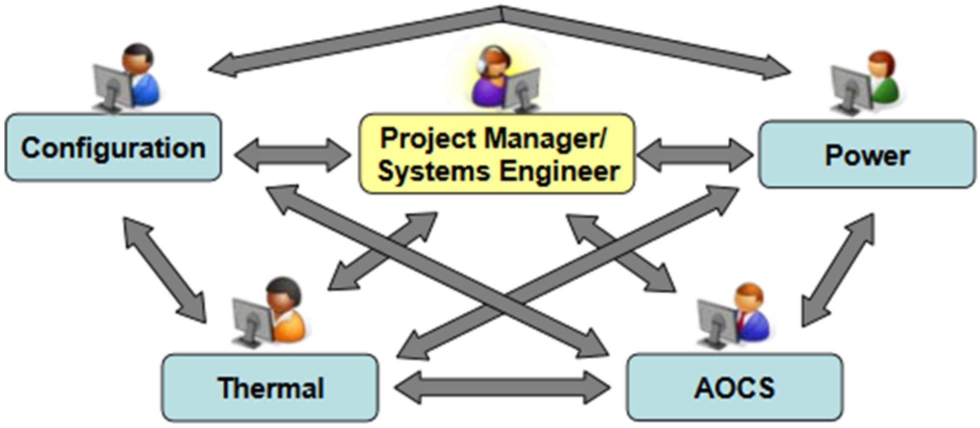



**Figure 1-3:** Concurrent Engineering Approach, (Xu et al., 2016)

During the preliminary design phase, the mathematical models are used for the analysis of the system performance. After analyzing this performance, the trade-off studies are carried out, to make the right choices to build a system. This is true, even if concurrent engineering approach is not followed. The complete preliminary mathematical model of a space system and the mission it will perform may involve more disciplines linked to each other and thus, it becomes very hard to manage. During the conceptual design, in general, the engineers and designer lack the specific and standardized design analysis techniques, despite having efficient models for each discipline. To find a reasonable solution, one needs to use these models properly, such that the user can find the driving factors and the interactions between the elements and disciplines. From this analysis, the user can thereby improve the system performance as a whole, by tuning the design factors according to the requirements of the problem.

The techniques should be specific in the sense that they should allow quantitative analyse to be performed quickly because time and resources are limited for conceptual design. The optimization techniques or quantitative sensitivity analysis are considered for more advanced design life cycle, but if applied in the conceptual design phase it can result to find the best optimal design configuration to be studied in the higher level of the design phase. Currently, there is a need to provide structured and common design methods involving all the design disciplines at the same time along with engineering data-exchange between the experts.

Aguilar et al. (1998) describe that, in 1997, the Aerospace Corporation developed the distributed Concurrent Engineering Methodology (CEM) architecture for their Project Design Center (PDC) facility, where a member of the team can work together in a focused environment utilizing models developed by the PDC. In the late 1990s, National Aeronautics and Space Administration (NASA) Goddard Space Flight Center (GSFC) developed the Integrated Design Center (IDC). It is a NASA Goddard facility that is used to perform rapid conceptual design studies for future NASA missions. During this process, IDC creates an environment that performs intensive sessions to examine all key technical aspects of the proposed concept and outlines the feasibility of the concept under the study and delivers a roadmap for how to make it a reality.

Ridolfi (2013) describes that in 1998, the European Space Agency (ESA), developed a Concurrent Design Facility (CDF). The CDF is a design meeting room that makes use of state-of-art information technology to create an integrated design environment, where the communication between the experts is made possible and efficient. In the CDF communication happens at all the levels, starting from the level of mathematical models that the experts use for the preliminary analysis. Modification in a single discipline or subsystem immediately reflects on the other disciplines and subsystems, creating a much higher level of knowledge of the evolution of the design amongst the members of the engineering team. The experience of the CDF has radically modified the classical sequential design approach, allowing capturing more knowledge at the beginning of the process and preserving design freedom for later phases to give the possibility to fully benefit from the additional knowledge gained by analysis, experimentation and human reasoning. Thus, an integrated design environment is an efficient approach that allows, responding of the increasing requirements of complete conceptual solutions in a short period of time and with limited resources. There are many other space agencies and research institutes who are now following a similar approach such are the German Aerospace Center (DLR), the Italian Space Agency (ASI), the French Space Agency (CNES), the Japan Aerospace eXploration Agency (JAXA). This approach is rapidly developing in the industries and universities like EADS Astrium, TU Munich, MIT, Stanford, Cranfield and many others.

Currently, Thales Alenia Space, Italia (TASI) use the Matlab/Simulink environment or ISight framework as an integrated platform, but these are costly solutions. Thus, TASI is also looking for a cheaper solution that can enable one to develop this design process for the multidisciplinary optimization of a re-entry vehicle. The developed methodology should be independent of the chosen integration design environment so that it can be easily implemented into any available integration design environment in future. To perform the feasibility study of the re-entry vehicle at the conceptual design phase using the developed methodology, it is required to choose a design framework. Mareschi (2014) describes the feasibility study on MDO for aerospace application, where it is discussed the activities performed by TASI for defining and developing a reference technology in the frame of concurrent engineering design as a tool for the multidisciplinary optimization. To validate the MDO approach, a market survey of the available commercial tools has been performed, according to the specific requirements as discussed further in Section 2-2-1. Furthermore, several MDO frameworks were split into commercial and academic or government tools categories and based on the requirements ISight as the framework was chosen. From this survey, Mareschi (2014)recommends the Dakota framework as an integration design environment to develop this methodology at low-cost. In Dakota framework, the integration of the selected design tool into one single design process is based mainly on bash commands/scripts that can be easily integrated with Dakota environment, but that, at the same time can be used in any other frameworks if required; for instance it could be implemented in ISight framework as well. Thus, for this thesis Dakota framework was chosen considering the cost, experience, user-friendliness and time required to develop the methodology using this framework. The framework is further described in Chapter 2.

### 1-1-2 Mission Heritage

#### Re-entry Vehicles Derived from the Baseline Configurations

There is a significant advancement in the propulsion system, landing mechanism, avionics, and interior of the spacecraft, but the aerodynamic shape are scaled or modified with respect to heritage design as shown in Figures 1-4 and 1-5. These vehicle configurations are designed for different mission objectives and system requirements. This thesis revolves around deriving an optimized re-entry vehicle using the baseline configuration as a starting point. These optimized configurations are derived for different mission scenarios using a concurrent engineering approach, such that they can satisfy the mission and system requirements. Thus, this section highlight few re-entry vehicles, that are developed from the benchmark designs and furthermore describes the design process used for multidisciplinary optimization of a re-entry vehicle.

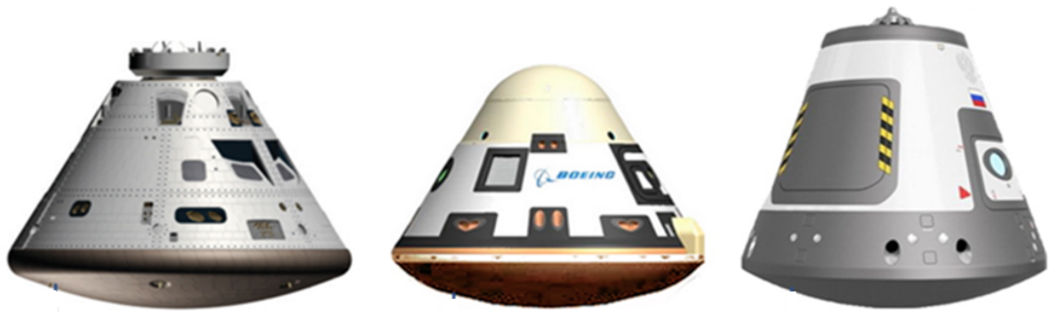


**Figure 1-4:** Orion, CST-100, Federatsiya, Image courtesy of NASA and Roscosmos

SpaceflightInsider (2016) describe these vehicles, Orion is the Apollo-driven capsule, where Orion is 5 meters in diameter and 3.3 meters in height, whereas Apollo Command Module was 3.9 meters in diameter and 3.47 meters in height. Orion is designed to send four to six astronauts atop the heavy-lift Space Launch System (SLS) booster to destinations, such as an asteroid or Mars. Thus, have different requirements regarding mass, radiation protection, avionics and the spacecraft's heat shield. There is a big difference between the entry velocities, for coming back from the interplanetary orbits. It is higher than 11 km/sec to return from the interplanetary orbits, whereas about 7 km/sec while coming back from the Low Earth Orbit (LEO). There is a significant difference when the vehicle gets into the higher velocities, the radiated heat from the shock layer is the dominating driver for the heat loads. When the speed is greater, it also affects the ablation rates and thereby also influences the thickness of the heat shield. This thickness of heat shield indirectly affects the total mass and the internal volume of spacecraft and to achieve the mass efficiency, one needs to optimize the vehicle.

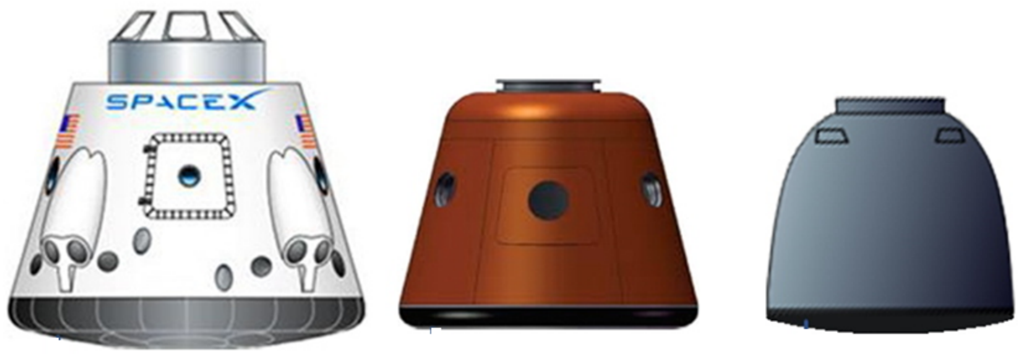



**Figure 1-5:** Space-X Dragon, ISRO Orbital Vehicle, Chinese MCV, Image courtesy of SpaceX, ISRO and CNSA

Furthermore, the Crew Space Transportation (CST)-100 Starliner crew capsule is a spacecraft design under construction by Boeing in collaboration with Bigelow Aerospace as their entry for NASA's Commercial Crew Development (CCDev) program. It is similar to the Orion, where it has a diameter of 4.56 meters which is slightly larger than the Apollo command module and smaller than the Orion capsule. This vehicle is compatible with multiple launch vehicles. Thus, the design criteria satisfy the outer envelope constraint of most of the launch vehicle. It can carry seven crew and cargo to the ISS and back to the Earth. The Orion capsule can splashdown in water during landing whereas CST-100 use airbags to land on the surface, thus the landing requirements such as accuracy, landing impact sustainability are different. Similarly, Federatsiya is a new Russian conceptual design similar to Orion capsule as shown in Figure 1-4; landing accuracy is around 10 kilometers and performs descent phase using parachutes, thus it's landing requirements are different than others. Furthermore, it is unmanned cargo version of the vehicle, that would carry about 2,000 kg to the Earth orbit and return about 500 kg back to the Earth. It is designed to conduct the fully automated and manual docking. It will be a reusable spacecraft and could fly up to 10 missions during a 15-year lifespan. Other similar shapes of re-entry vehicle are Chinese Multi-purpose Crew Vehicle (MCV) is also very similar to Orion, Indian Space Research Organisation (ISRO) Orbital Vehicle, which is smaller than Apollo capsule and SpaceX Dragon, as shown in Figure 1-5. SpaceX Dragon is designed to transport the cargo and crew to the ISS. SpaceX Dragon is having the upmass capacity of 6000 kg, whereas the total return payload mass of 3000 kg. Since these designs are developed for different mission scenario, the mission and systems requirements are completely different, but all these vehicles can be derived from a baseline configuration such as Apollo.


**Figure 1-6:** Space Shuttle ,Image courtesy of NASA

Similarly, the American Space Shuttle is considered as baseline configuration, that has been the primary spacecraft for ISS construction. The Space Shuttle orbiter as shown in Figure 1-6 was the reusable space-plane component of the Space Shuttle program operated by NASA. Six orbiters were built for flight: Enterprise (OV-101), Columbia (OV-102), Challenger (OV-099), Discovery (OV-103), Atlantis (OV-104), and Endeavour (OV-105). The lifting re-entry configurations are interesting because of their cross- range and downrange capability and low-speed handling qualities. Space shuttle program took the advantage of numerous experimental lifting vehicles such as ASSET, X-15, PRIME (X-23A) and X-24. With X-15 knowledge regarding the metallic thermal protection, terminal area energy management, and nonpropelled landing for a vehicle with a poor L/D ratio was gained. Through the ASSET and PRIME (X-23A) orbital and suborbital re-entry flights, the Flight Control System (FCS) and Reaction Control System (RCS) efficiency, thermal protection system (metallic), aero-thermodynamics measurement, flight worthiness and guidance accuracy have been explored. X-24 landing training vehicles with poor aerodynamic characteristics and transonic control ability have been investigated.

Krevor et al. (2011) describe the Dream Chaser as an American reusable automated crew and cargo carrying lifting-body spacecraft developed by Sierra Nevada Corporation (SNC) Space Systems

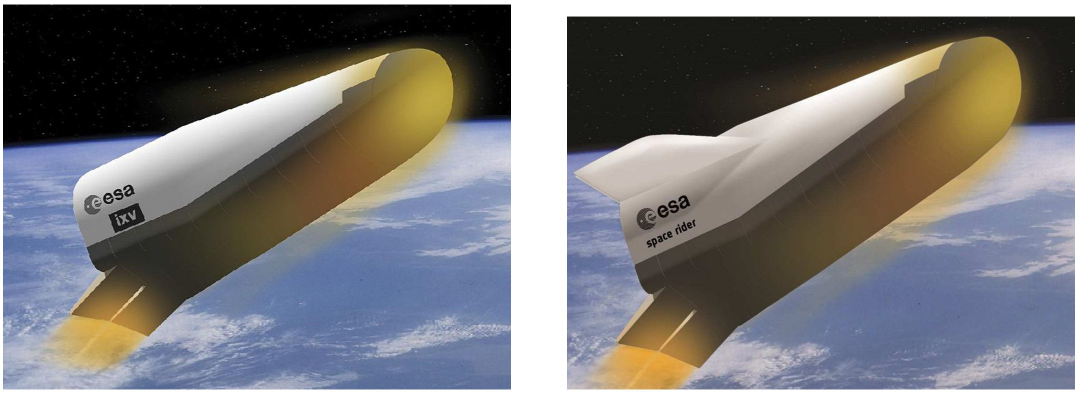
**Figure 1-7:** Dream Chaser, Image courtesy of Sierra Nevada Corporation

as shown in Figure 1-7. The Dream Chaser is designed to resupply the ISS. It can carry the pressurized as well as the unpressurized cargo. It is based on NASA Langley's Horizontal Lander HL-20 lifting body design concept. It will be launched vertically and land horizontally on the conventional runways. It will have the capability to re-enter with low-g of about 1.5 g thus, protecting the crew. It is having mass of 11,300 kg and length of 7 meters.

X-38 program was canceled by NASA in 2002, but its shape was similar to X-23. This design introduced the nose and body flap technology in Europe as well as in the field of aerodynamic characterization of the high speeds shape designs. In the frame of Buran program, Russian in-flight experimentation developed BOR-4 and BOR-5 which were capable of flying up to Mach 25 and Mach 18, respectively. Japan developed experimental vehicles started a program called HOPE-X and developed HYFLEX and HSFD under this program, these vehicles concerned about the flight guidance, aerodynamics and post-flight analysis. In Europe, Hermes program was started with a focus on reusable vehicles and Shuttle/Buran like technologies. After Hermes continued with FESTIP, X-38, FLTP, ANGEL, PHOENICS and ARD programs for re-entry studies.

ESA's Intermediate eXperimental Vehicle (IXV) was an atmospheric re-entry demonstrator as shown in Figure 1-8a. The further advancement in the IXV configuration is in the preliminary design phase at ESA called as Space Rider as shown in Figure 1-8b. Its shape is still the subject of debate and under preliminary design phase, but the vehicle should be sized to fit the cap of Vega C launcher, an evolved version of the current Vega. Another feature is Space Rider may land on a runway or use parafoil to reduce the extent of the landing site compared to a conventional parachute. This configuration is also compared with the X37- B configuration. Rufolo (2016) describes that Space Rider will be similar to the IXV configuration, but with a payload capacity of 450 kg.

As described previously, this thesis involves multidisciplinary optimization of a re-entry vehicle, where an optimized configuration is derived from a baseline configuration for different mission scenarios. Advancement in the IXV for payload carrying capacity is an on-going research project at TASI in collaboration with ESA, thus deriving a new vehicle considering IXV as a baseline configuration for the mission and system requirements of Space Rider is most appealing. Furthermore, this thesis is conducted at TASI, thus all the necessary information about the reference vehicles, such as IXV is easily available. Based on this criteria, IXV is considered as a reference vehicle for this thesis and using the multidisciplinary optimization design process developed within the




**(a)** IXV **(b)** Space Rider

**Figure 1-8:** IXV and Space Rider, Image courtesy of ESA

Dakota integration framework, design case such as IXV with payload carrying capacity is derived for the some of the mission and systems requirements derived from the Space Rider re-entry vehicle. The details of these reference configuration will be further discussed in the relevant chapters of this thesis.

#### Design Process for MDO of a Re-entry Vehicle

Hammond (2001) describes that the MDO branch of NASA Langley Research Center (LaRC) defined MDO as follows, **"A methodology for the design of complex engineering systems and subsystems that coherently exploits the synergism of mutually interacting phenomena."** In the complex and multidisciplinary design problems such as re-entry vehicles, there is more than one objective (cost) function to be optimized. MDO techniques have been developed to allow multi-disciplinary systems to reach a global optimum. In the 1960s to 1990s, there were efforts towards MDO mainly by NASA and its collaborating universities and researchers, such as Sobieski, Alexandrov, Haftka, Lewis, Braun, Kroo, Olds and others. Even though the MDO has been widely used in all domains of complex engineering applications, still it's efficient use in space systems optimization is a new research topic and is currently being considered for spacecraft and launch vehicle designing processes.

Use of MDO techniques in the preliminary phase of the design process is more beneficial as there is higher design freedom and different configurations can be considered according to the objectives and constraints. Once the knowledge is gained about the product or the design gets matured, then MDO techniques might not be used as it will over constrain the design process but can be used to optimize the financial aspects of the design. Thus, MDO techniques can be used to derive the optimal design solutions for the different objectives and satisfying the constraints. Developing an optimal configuration at the preliminary design phase reduces the time required for the higher design process due to a stronger preliminary stage design.

Figure 1-9 shows the general multidisciplinary optimization process, where design vector is given as input to the simulator code. The simulator code computes solutions from toolchain and generates the objective vector for the given inputs. The optimization algorithm is then applied to these objective functions and re-calculate the design vector. The loop continues till an optimal solution is derived from the computation.

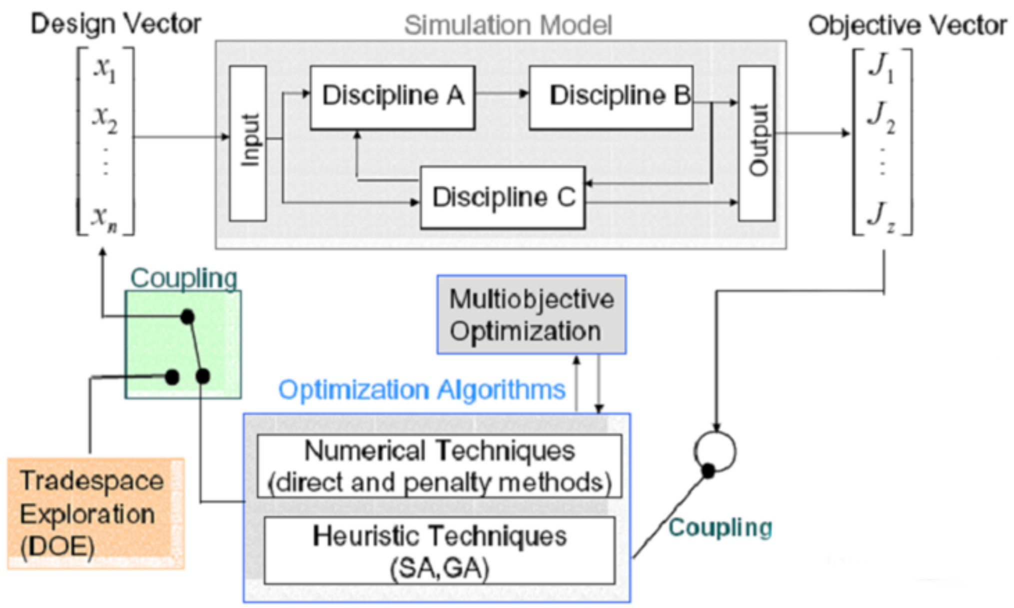



**Figure 1-9:** Multidisciplinary design optimization, (Riccardi, 2012)

Riccardi (2012) gives the general MDO problem formulation as follows:

$$\textbf{Minimize: } \mathbf{f}(\mathrm{x,y})$$
$$\textbf{With respect to: } y$$
$$\textbf{Subject to: } \mathbf{g}(\mathrm{x,y}) \leq 0,\ \mathbf{h}(\mathrm{x,y}) = 0,\ \mathbf{c}(\mathrm{x,z}) = 0$$

where,
**f** is the objective function,
**g** is the inequality constraint,
**h** is the equality constraint,
**c** is the coupling constraint,
$y$ is the design variable,
$x$ is the coupling variable.

The objective function describes the functions that are required to be minimized or maximized. The equality and inequality constraints represent the disciplinary restrictions. Coupling constraints symbolize the consistency of the coupling between the different subsystems. Design variables can be used in one or several subsystems. Design variables are either specific to one subsystem called as local variables or shared with various subsystems and called as global variables. Coupling variables are used to build a consistency between the different subsystems.

Schmit (1960) published an innovative paper on a structural design by systematic synthesis, which marked a revolution in the structural design procedures. Later, Schmit and Thornton (1965) applied this method for the synthesis of an airfoil at supersonic Mach number, where they used the gradient steep descent alternate step synthesis method. Haftka et al. (1975) extended their experience in structural optimization to include other disciplines and used different structural optimization procedures and compared the results obtained from the mathematical programming and optimality criteria procedures. Grossman et al. (1988) published a paper on integrated aerodynamic, structural design of a sailplane wing and transport wing, where they studied the integrated design approach and compared with the results from sequential optimization approach. Antoine and Kroo (2005) developed, a preliminary design tool using multi-disciplinary, multi-objective genetic algorithm to determine the optimal aircraft configuration and to derive the sensitivities between the conflicting objectives. Henderson et al. (2012) used an augmented Lagrangian particle swarm Optimizer and a genetic algorithm to solve single objective and multi-objective optimization problems of aircraft conceptual design. Tröltzsch et al. (2014) from German Aerospace Center (DLR) actively worked on the Spaceliner and developed the conceptual ultra fast aircraft. They implemented a MDO framework to analyse the DLR Spaceliner design concept involving different engineering disciplines, Figure 1-10. In this figure, it can be observed that design disciplines are integrated on a single platform to perform the feasibility study of the hypersonic aircraft.

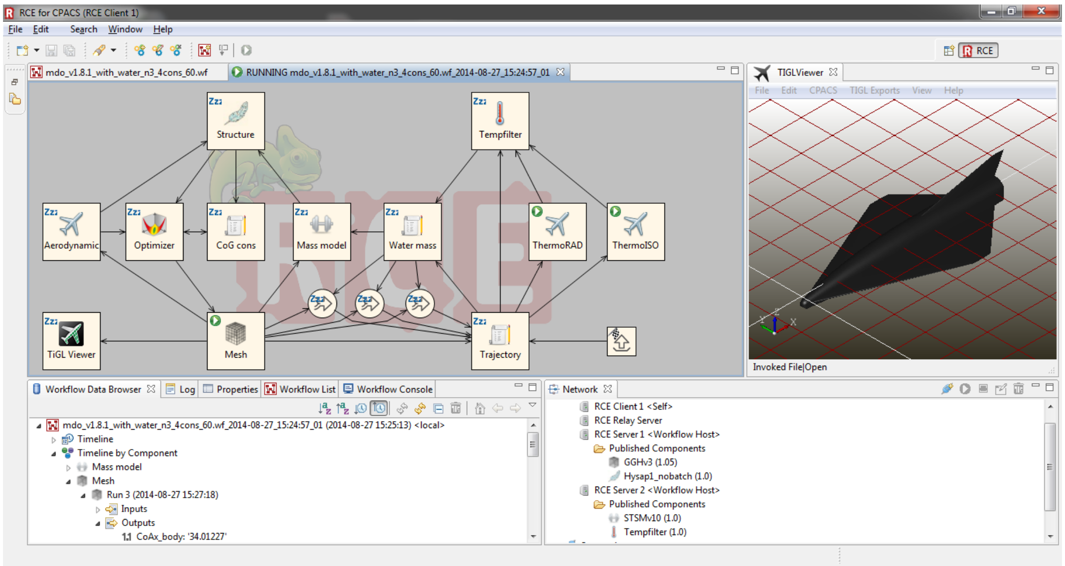



**Figure 1-10:** Spaceliner using RCE framework, Image courtesy of DLR

If we look at the past, present projects there are not many research projects over the MDO application for re-entry vehicle design. Gang et al. (2005) published a paper on MDO application on re-entry trajectory. In this article, a conventional re-entry trajectory optimization problem is built on given aerodynamic configuration and total mass, where they investigated bi-conic configuration with objectives to minimize the mass of the Thermal Protection System (TPS) and maximize cross-range. They used two kinds of MDO algorithms, All In Once (AIO) and Collaborative Optimization (CO), to solve the re-entry optimization problem including aerodynamic configuration, flight dynamics and TPS. The results show that a MDO algorithm is an important tool for the preliminary configuration design of re-entry vehicle.

A similar approach is carried out in space plane design optimization, Yokoyama et al. (2001) used the MDO for space plane design and trajectory optimization of a space plane ascending to the ISS, where the Block Diagonal Hessian (BDH) method is applied for the trajectory optimization. In this project, during the analysis they considered the configuration and the propulsion system as fixed, however, the L/D parameter of the configuration influence the take-off window, thus they recommended to perform a simultaneous optimization of trajectory and configuration of the vehicle for future studies.

Later, Yokoyama (2002) presented a paper on trajectory optimization of space plane using the Genetic Algorithm (GA)combined with gradient method. Here the GA is applied to obtain an appropriate guess of the optimal solution, and subsequently, gradient-based method is used to refine the solution. Generally, for the trajectory optimization a point mass is assumed, and thus rigid body dynamics is not covered. Thus, in this project trajectory optimization of a space plane with rigid body assumption was covered using effective optimization methods. Further, the same author, Yokoyama (2004) presented a paper, where a conceptual design of single-stage-to-orbit (SSTO) space plane as shown in Figure 1-12, is designed using the MDO techniques, covering the rigid body characteristics. Figure 1-11 gives the schematic view of the design process using the MDO techniques. The results from this project indicated that MDO technique successfully worked to design the vehicle, that has the trim capability throughout the flight and stability during the flight with high dynamic pressure. Additionally, meta-models are used to reduce the computational cost for the analysis of the aerodynamic characteristics and the propulsion performance.

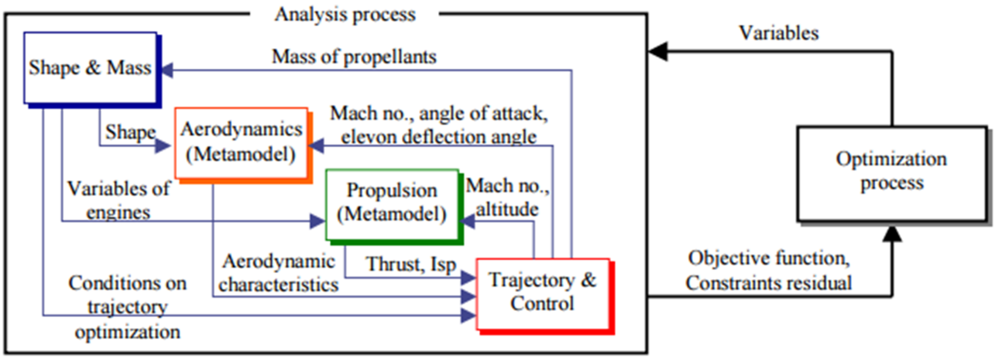



**Figure 1-11:** Schematic view of MDO, (Yokoyama, 2004)

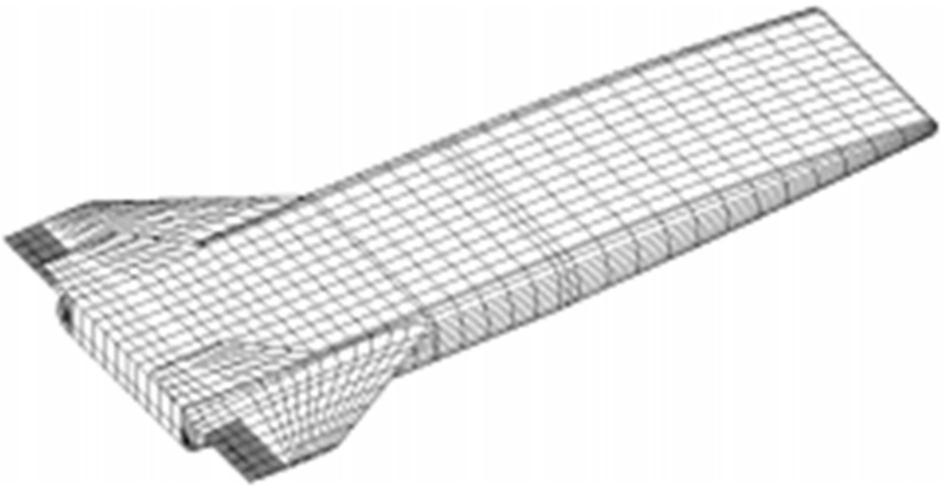


**Figure 1-12:** Shape of the vehicle (nominal case), (Yokoyama, 2004)

Rasmussen (2006) developed a surrogate model based on gradient enhanced Gaussian process methodologies, but there was difficulty in fitting the aerodynamic data over the Mach number range of space vehicles. Later research was conducted and partitioning of design space and implementation of mapping function for the covariance was introduced into the process. Yokoyama continued the work in the MDO of space planes along with Tsushiya and Takeshi. Later, Yokoyama et al. (2007) once again, presented a paper on MDO of a space plane considering rigid body characteristics. This time, All-At-Once (AAO) based multidisciplinary optimization approach incorporating sparse nonlinear programming and meta-modeling were used to optimize the flight trajectory. Figure 1-13 gives the schematic view of AAO-based MDO framework. Furthermore, Tsuchiya et al. (2007) continued the work described by Yokoyama et al. (2007), where a hypersonic experimental vehicle is designed using MDO techniques to demonstrate the performance of a hypersonic engine.

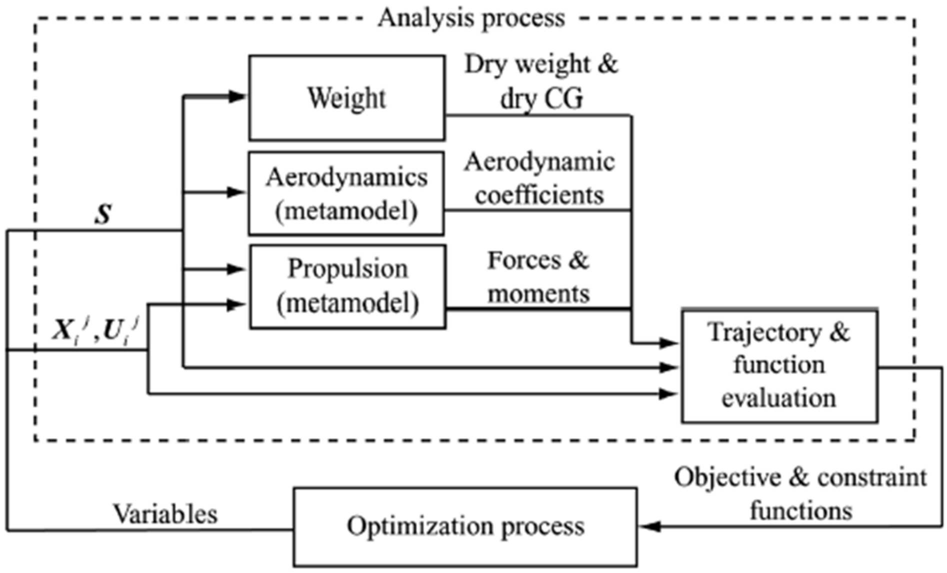



**Figure 1-13:** Schematic view of MDO, (Yokoyama et al., 2007)

Furthermore, Nosratollahi et al. (2010) describe the optimal design of the re-entry vehicle configuration. It is designed in such a way, that it will minimize the cost of the mission. Minimizing the lateral area for minimum structural mass, minimizing the total heat absorbed for minimum heat shield mass, decreasing the mission cost and maximizing the drag coefficient for decreasing the speed in the last phase of flight were the objectives considered for this optimization. Here the axis-symmetric capsule is considered as a re-entry vehicle, where the configuration can be varied to mono-conic to the tri-conic axis-symmetric capsule. In this project, the single-objective GA and multi-objective GA are employed for optimization. The conclusion of the paper highlights the differences in the optimized solutions for different combinations of objectives (single or multiple) as shown in Figure 1-14. In this figure, the first configuration is derived for multiple-objectives, whereas others derived by using single objective for optimization. The second configuration is derived using the objective to maximize the drag coefficient. The third configuration is derived using the objective to minimize the total heat absorbed whereas the fourth configuration is derived using the objective to minimize the structural mass.

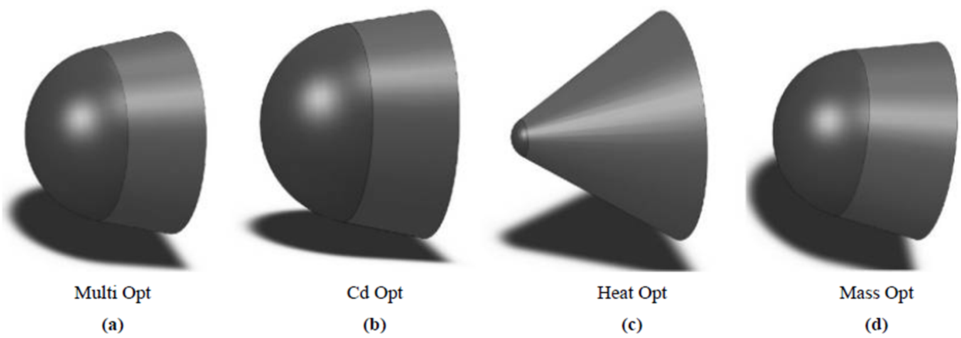



**Figure 1-14:** Multi-objective and single objective optimization results, (Nosratollahi et al., 2010)

Again Adami et al. (2011), the same group of researchers, published a paper on MDO application on a manned re-entry mission where trajectory and aerodynamics disciplines are considered for optimization of re-entry bi-conic configuration. They developed a framework using AAO MDO strategy and utilized GA with nonlinear constraints to minimize the mission cost by reducing re-entry module mass. The recent publication by Adami et al. (2015) on multidisciplinary design optimization of a de-orbit maneuver considering propulsion, TPS and trajectory depict the optimal deorbit parameters to minimize the thermal protection system mass and de-orbit propellant mass using MDO technique. This project was based on the AAO framework, where GA was used to perform the optimization. Figure 1-15, gives the geometry and TPS parameters of the considered bi-conic configuration. GA allows to find the global optimum and one can use the population of guesses, which are randomly spread throughout the design space. GA provides powerful operators for crossover, mutation, and selection of the population, which helps the optimizer to direct the members of each population towards the desired goal of the problem. From the results, it is concluded that simultaneous optimization considering multiple disciplines is more efficient that the single discipline optimization. Furthermore, it is also recommended to use hybrid optimizer such as combination of a GA and a gradient-based method such as the Sequential Quadratic Programming (SQP) to refine the results.

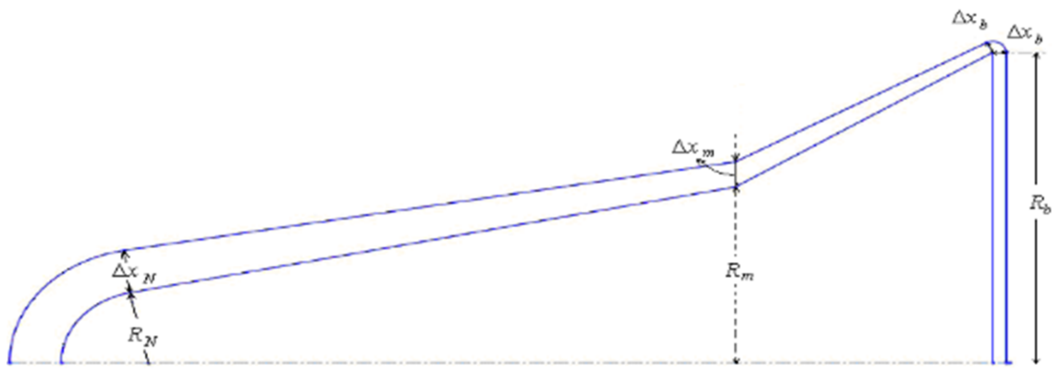



**Figure 1-15:** The geometry and TPS parameters, (Adami et al., 2015)

Dufour et al. (2015) presented a paper on trajectory driven MDO of a sub-orbital space plane, where a non-stationary Gaussian process was used for optimization. Priyadarshi and Mittal (2010) published a paper on multi-objective multi-disciplinary design optimization of a semi-ballistic re-entry module. In this paper, optimization of system level design objectives, cross-range, and total mass, has been attempted in a multi-objective, multi-disciplinary framework. The characteristics of the configurations have been analyzed in the Pareto set and performed a comparative analysis.

### 1-1-3 MDO Architecture

The MDO architecture concerns about the structure of the multidisciplinary analysis and optimization. Each architecture is a combination of both problem formulation method and problem decomposition method. The problem formulation methods are Multi Discipline Feasible (MDF), Individual Discipline Feasible (IDF) and AAO and the problem decomposition methods are No-Decomposition (ND), Hierarchical Decomposition (HD) and Non-Hierarchical Decomposition (NHD). Thus, it is required to know the basic architectures formed with the combination of problem formulation methods and problem decomposition methods.

To perform the optimization, one has to consider that each subsystem may or may not have an optimizer but the system optimizer should take care of the optimization process. No matter which programming language or subsystem MDO architecture is used, one should be able to derive converged optimal solutions. The MDO architecture can be executed at a single level or a multi-level for both single objective or multi-objective problems. The tools available for each discipline, for this thesis are without optimizer thus, single level optimization architecture are considered over the multi-level optimization. There is more than one objective function for each discipline in our research goal thus we also need multi-objective problem-solving strategies. These factors were considered to select the appropriate methods and techniques. This section describes the methods to decompose the MDO problem and its formulation methods, respectively, as engineering practice of optimization applied to large problems. Furthermore, it describes the methods considered for this thesis for multi-objective optimization techniques and sampling, respectively.

### Methods for Problem Decomposition

Problem decomposition methods improve the efficiency and reduce the complexities of the optimization process. This is done by subdividing the multidisciplinary model into smaller blocks and grouping them together to a highly coupled design variables, constraints, and governing equations. Riccardi (2012) described three methods to decompose the problem, in general, such as ND, HD and NHD approach.

#### Trade-off of Problem Decomposition Methods

**Table 1-1:** Trade-off MDO problem decomposition methods

| Features | Hierarchical | Non-Hierarchical |
|---|---|---|
| Interaction between levels of subsystems | Yes | Yes |
| Interaction between subsystems at same level | Yes | Yes |
| Interaction between subsystems at different levels | No | Yes |
| One way or both ways interaction | Both ways | Both ways |
| Number of coupling variables | Less | More |
| Ease of implementation | complex | complex |

To design complex systems, ND has been one of the first and most straightforward methods. In this process, a monolithic model where all the governing equations of the elements and disciplines are managed together. This model is just a black-box function that takes input and constraints, providing output to the user(s). ND is easier to implement and use for designing. It does not allow efficient code re-utilization, modularity, maintainability and scalability is the disadvantages of this method. This approach has the advantage of the high potential of direct linking of the monolithic model with an optimization algorithm, which may use the model as if it were an objective function(s) to obtain the feasibility/optimality of the design. Ridolfi (2013) states that the ND kind of approach becomes inadequate for applications involving a large number of design variables and constraints or fully coupled problems, which require several iterations to converge to a consistent design. Thus, this approach is not considered for the system study of a complex system like re-entry vehicle design, which includes a large number of design variables and complexity. For this thesis, we need to decompose our system into subsystems, thus ND can be discarded from our selection process, giving us two choices for problem decomposition methods, i.e., HD and NHD approach. The option between these methods is based on the coupling characteristics of the system study of mission selected.

The HD method allows managing the complexity, providing ease of maintainability of the code, modularity, and scalability. The HD method treats the system models by dividing into several independent sub-models. The element models in which the system model is divided are independent of the local parameters. It exploits the concept of modularity, giving the user the possibility to treat the models separately. The important feature of the hierarchic decomposition approach is that the flow of the input and output information is only in the vertical direction and no data is transferred between pairs of boxes located at the same decomposition level. This method gives a benefit of optimizations at the same level and execution of the process in parallel. N-square diagrams are utilized in case the subdivisions are not straight forward and the feed-forward and feed-backward information paths are represented above and below the diagonal respectively. It also has a disadvantage that the coupling between the blocks or the elements involves global parameters. This method is used for more complex optimization problems and thus, one can consider it for re-entry vehicle design optimization.

NHD is used when the fully coupled multidisciplinary system cannot be decomposed into a pure hierarchy of blocks or modules. The NHD is better for complex systems if identifying the hierarchy in complex systems becomes difficult. Different individuals develop the mathematical models of the elements of a complex system and subsequently linked together through input/output software architecture. Therefore, independence between the elements cannot be assessed prior the primary goal of designing using all the element models concurrently is to understand and explore the interactions. Thus, the NHD and HD are both promising problem decomposition methods that can be implemented in this research goal, as we are looking for parallel computation. NHD is preferred over the HD as the defining the pure hierarchy in a complex problem is difficult. Additionally, the NHD approach is flexible enough to allow for a plug and play management of the mathematical models of the elements and it naturally behaves as the hierarchically decomposed model in the case of completely uncoupled elements.

**Selected Method for Problem Decomposition**

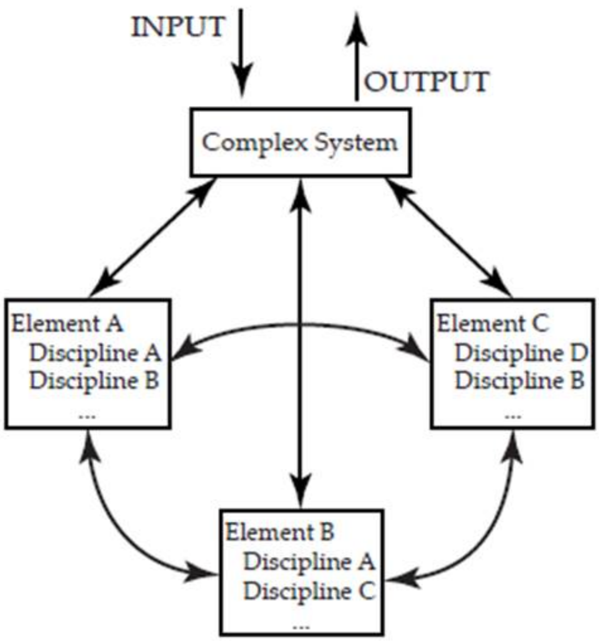



**Figure 1-16:** Non-Hierarchical Decomposition (NHD) method, (Ridolfi, 2013)

From the above trade-off, it is preferable to use NHD over the HD. In the case of NHD, the system is not required to be decomposed in purely hierarchic of blocks or modules. There is no priority of inserting any module at the top level in the hierarchical structure as shown in Figure 1-16. The flow of information is in vertical as well as in the lateral direction. The information flow is more complex and includes more variables. This method is used for more sophisticated handling of optimization problems, thus can be considered for space vehicle design optimization.

In the case of re-entry vehicle optimization, the design disciplines are inter-linked and the hierarchic decomposition is may not be easy and thus, NHD will be useful. If the priority of design disciplines are known, it can be easily converted to the HD method. The decomposition method can be implemented within the simulator script of the Dakota framework, where the interfaces of each discipline can be mentioned.

#### Methods for MDO Problem Formulation

The problem formulation methods are classified as single-level methods and multi-level methods. In single-level methods, all the cases the architecture is conceived so that the analysis, or the optimization, of the system, has to be performed in one single location. Thus, analysis is performed at the model optimizer interface. Whereas, in the multi-level methods or collaborative environment the team experts can participate in the process from different locations. The few methods of multi-level optimization are Blackbox optimization, Nested Optimization Loop, Linear Decomposition, Concurrent Sub-Space optimization, Bi-Level Integrated System Synthesis (BLISS) and Collaborative optimization. For this thesis, it does not matter if the modules in simulator have an individual optimizer or not, the Dakota optimizer takes care of the complete optimization procedure. Considering the time frame for the thesis and the requirements it is preferred to use single-level optimization strategy. In single-level strategy, only one system optimizer is present, which is responsible for the design variables and there is a coordinator that is responsible for the state variables. Thus, this section only highlights the single-level problem-formulation methods.

The three most important single-level methods of describing MDO problems are namely: MDF, IDF and AAO. The different MDO strategies are distinguished themselves by the distribution of the design and co-ordination tasks throughout the system architecture as described by Steenhuizen (2015). MDF method contains two Disciplinary Analysis (DA) process (DA1, DA2), which are coupled via a System Coordinator (SC). For each disciplinary analysis, there is a Disciplinary Evaluator (DE), and a Disciplinary Coordinator (DC). The System Optimizer (SO) determines and controls the value of the system design variables and disciplinary design variable based on some system objective and constraints. DE computes output state, disciplinary residual, objective function, constraint values, the system output state and disciplinary coordinator helps to determine the disciplinary input state. The disciplinary analyzer is the combination of disciplinary evaluator and disciplinary coordinator. Together with the two disciplinary analyzer, comprises the system evaluator. The SC handles the interdisciplinary relations to keep the solutions consistent at every step. Thus, a fully coupled multidisciplinary analysis is performed at every optimization step. Furthermore, the IDF method contains two DA process (DA1, DA2). For each disciplinary analysis, there is a DE, and a DC. In this method, there is no SC. Thus, the systems state variable is also controlled by the SO. The internal coupling variable is controlled by the DC. The SO determines and controls the design variables along with the system state variable to ensure the multidisciplinary consistency. The AAO method contains two DE and a SO. For each disciplinary analysis, there is a DE. The SO determines and controls all the design variables to ensure an interdisciplinary level as well as at the disciplinary level consistency. The SO determines and controls the value of the system design variables and disciplinary design variable, based on some system objective and constraints. Along with the above variables, the SO controls the system state variable and internal coupling variable.

#### Trade-off of Problem Formulation Methods

All the methods give multidisciplinary feasibility at convergence, but if we consider the feasibility at each iteration, MDF is better where it meets our requirements of an inter-disciplinary feasibility, whereas the IDF provide individual discipline feasibility and AAO provides no feasibility. In the case of optimization efficiency and effectiveness, the AAO is leading, where no design solution lost in the optimization procedure, whereas in the case of IDF the efficiency and effectiveness are

**Table 1-2:** Trade-off MDO problem formulation methods

| Features | MDF | IDF | AAO |
|---|---|---|---|
| Feasibility at each iteration | Multidisciplinary feasibility | Individual discipline feasibility | None |
| Optimization efficiency | Low | Depend on coupling level | High |
| Optimization effectiveness | Low | Medium | High |
| Maintainability and re-usability of the codes | Medium | High | Low |
| Parallel computation | Low | Medium | High |
| Optimization variables | Design variables | Design and coupling variables | All design and behavior variables |
| Optimization problem type | Small and dense | Average | Large and sparse |

both medium and depends on the coupling, whereas the efficiency is low for MDF but have high effectiveness.

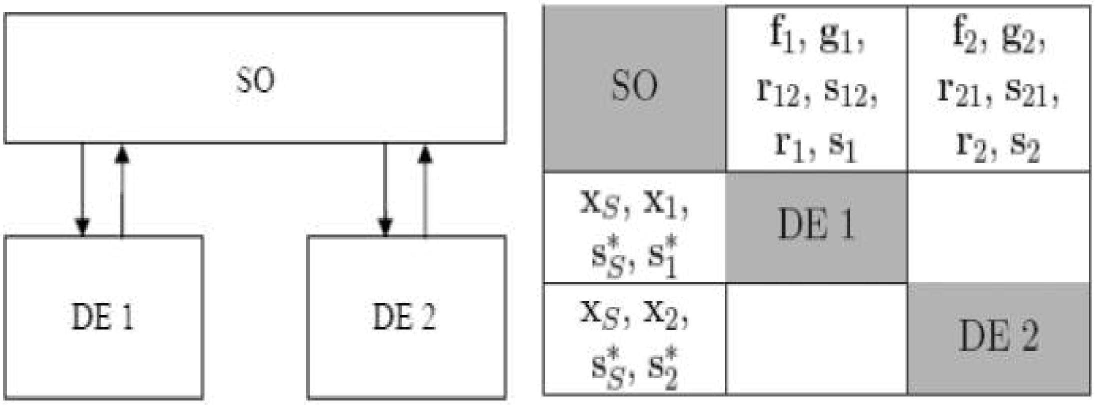
The maintainability and re-usability of the codes are highest for the IDF and is as low for the AAO and medium for the MDF. The parallelization is high in case of the AAO and low in the MDF but still, the IDF maintains it to medium level. The AAO method considers all design variable and behavior variables for the optimization process, whereas the IDF considers the design and coupling variables and the MDF just consider the design variables. The AAO can be applied on the large and sparse optimization problem type whereas IDF for medium and MDF for only small and dense problems.

The choice between the various formulation approaches mainly depends on the objectives of the analysis, on the complexity of the problem and on the design environment in which they will be adopted. From a practical point of view, the MDF approach can be considered the easiest and most straightforward. It inherits its working principle from the classical engineering design approach: multiple iterations until convergence, for every given design variable set. On the other hand, the IDF and AAO seem promising, concerning parallelization of the analysis and reduced computational effort required for complex systems models. The potential to parallelization of IDF is related to the coupling relations between disciplines.

Thus, the trade-off shows that there is a hard choice among them. MDF, AAO and IDF are in competition with each other, whereas the AAO has the only disadvantage of maintainability and reuse. For this thesis, the tools are mostly without an optimizer. Thus, a system optimizer is the only optimizer to derive a converged solution. Additional optimizer or coordinator cannot be introduced to these tools as they will act as a black-box for the Dakota optimizer. In such case, one can use the AAO only. Specific boundary conditions are given for the initial set of the design parameters, which will result into at least one feasible or set of possible optimal solutions. Thus, the AAO is considered over the MDF and the IDF for its efficiency-effectiveness and parallelization qualities. Additionally, the work by Yokoyama et al. (2007) and Adami et al. (2015), where they have successfully implemented AAO as a problem formulation method for space plane and re-entry capsule, gives the confidence to use this method for this thesis.

**Selected Method for Problem Formulation**

The AAO method is also known as Single SAND-SAND (SSS) and is the most elementary MDO method. In this case, not only the multidisciplinary optimization coordinator is removed, but also the disciplinary coordinators are also removed compared to MDF. Thus, the SO is responsible for ensuring the consistency at an interdisciplinary level as well as at the disciplinary level. In this method, the optimization problem and the equations of the different subsystems are solved




**(a)** All-At-Once (AAO) approach

**(b)** AAO design structure matrix

**Figure 1-17:** AAO approach and its design structure matrix, (Steenhuizen, 2015)

simultaneously. The system level optimizer aims to optimize a global objective and calls the subsystem evaluations. The design variables, coupling variable and state variables are all handled by the optimizer. The design and the evaluation of the subsystem level are performed at the same time. The equations are not satisfied at each iteration of the optimization process, but they have to be at the convergence. The schematic representation of a basic AAO architecture for two disciplines and AAO design structure matrix is depicted in Figure 1-17a and Figure 1-17b respectively.

Figure 1-17a, shows that the AAO method contains two DE and a SO. For each disciplinary analysis, there is a DE. The flow of variables is in the clockwise direction. As shown in Figure 1-17b, the system optimizer determines and controls all the design variables to ensure an interdisciplinary level as well as at the disciplinary level consistency. It determines and controls the value of the system design variables $x_s$ and disciplinary design variable $x_i$ for discipline i, based on some system objective, constraints, and disciplinary residual $r_i$. The system objective is the combination of disciplinary objectives $f_i$ and disciplinary constraints $g_i$. Along with the above variables, the system optimizer controls the system state variable $s_s$ and internal coupling variable $s_i$.

#### Methods for Multi-objective Optimization

Chiandussi et al. (2012) describe the methods for the multi-objective optimization used for engineering applications. For this thesis, it is required to consider multiple objectives to derive an optimized re-entry vehicle. Thus, multi-objective techniques are considered over the single objective techniques and thus not taken into account for any further selection process. In the case of multi-objective optimization, one has to identify the better or optimal solutions with respect to defined performance criteria. In multi-objective optimization problems, the set of optimal compromise solutions (Pareto front) has to be identified by an efficient and complete search procedure to let the designer carry out the best choice. The research goal involves global multi-objective optimization and this study will focus on the methods available for the same as studying all the optimization techniques is beyond the scope of this study. The methods are classified as a priori articulation of preferences, posteriori articulation of preferences and no articulation of preference.

**Trade-off of Optimization techniques**

In the priori articulation of preference, the user indicates the relative importance of the objective functions or desired goals before running the optimization algorithm. In the posteriori articulation of preference, the user selects a single solution from the set of mathematically equivalent solutions. The no articulation methods are applied when the user is unable to make a concrete decision of what he or she prefers. The methods that provide both necessary and sufficient conditions for Pareto-optimality are preferable. The following paragraphs describes these categories.

The priori articulation of preference methods allows the user to choose the important objective function based on the mission requirements. The preference of the decision maker is indicated by the use of parameters such as coefficients, exponents and constraints limits. Weighted global criteria method, weighted sum method, lexicographic method, weighted min-max method, exponentially weighted criterion, weighted product method, goal programming methods, bounded objective function method and the physical program are some of the methods classified under this category. Introducing weights is a converting multi-objectives to single objective function where the preference objective is considered and others are converted into constraints. These category methods can be considered for multi-objective optimization, but the results depend on the accuracy of the selected weights. It is difficult to set the weights or preference to obtain a Pareto-optimal solution in the desired region in the objective space. Consistent variation in this setting parameters can theoretically yield the complete Pareto-optimal set, but difficulties may be encountered in selecting parameters that provide a feasible solution. Thus, these methods are not preferred for the multi-objective optimization of the complex system like re-entry vehicle design.

The no articulation methods are applied when the user is unable to make a concrete decision of what he or she prefers. These methods do not need any articulation of preference, but are the simplification of the method with the priori articulation of preference. Global criterion method and Nash arbitration and objective product method are some of the methods of this classification. Global criteria method that uses the weighted exponential method by setting all the weights to one single objective function. The global criterion method is particularly suitable when the ideal value of the objective functions is known and the user can define the target values of the optimization. When the objective function bounds are known and it is particularly efficient with convex and linear Pareto fronts. This method is efficient to identify the non-convex or discontinuous Pareto fronts, but fails in the identification of concave Pareto fronts. Nash arbitration scheme is derived from game theory and is based on the predetermined axiom of fairness. The Nash arbitration scheme provides an approach that alleviates the need for function transformation, but introduce non-linearity and also have computational difficulties. Thus, these methods are not preferred for re-entry vehicle optimization.

The posteriori articulation preference methods are used when it is not possible to express the preferences and thus need to choose from a palette of solutions. These methods allow the user to explore the options before making a decision. From the most appealing solution, the decision is made by the user rather than analyzing the important objective functions. Physical programming, Normal Boundary Intersection (NBI) and Normal Constraint (NC) are some of the methods of this classification. To select the appealing solution from the set of solutions, one must eventually present the solutions to the decision maker in graphical or tabular form, and choosing a single solution can be an intimidating task with a relatively large number of objectives, variables, or solution points. Consequently, these methods are suitable only for the problems with less number of objective functions. If more than one solution is considered then a clear Pareto optimal set is possible at the cost of computational time. Considering these methods will add a restriction on the number of objective functions selected for re-entry vehicle design optimization. These category methods are also not the preferred option. Thus, neither of the above classes are suitable for the re-entry vehicle multi-objective multidisciplinary optimization. Furthermore, Marler (2004), describe genetic multi-objective algorithm as an approach for a posteriori articulation of preferences, which is intended for depicting the complete Pareto optimal set. MOGA provides an alternative to the methods described in the posteriori articulation of preferences classification. In those methods, one Pareto point is determined at a time. Additionally, each point requires the solution of a single-objective optimization problem, whereas the MOGA does not need solving a sequence of single-objective problems.

Chiandussi et al. (2012) describe other Evolutionary Algorithms (EA) for general engineering application. Thus, EA for searching the near-optimal solutions for the given problem is preferred by the researchers. Evolutionary algorithms are suitable to solve multi-objective optimization problems as they deal simultaneously with a set of possible solutions. In case of the traditional techniques, the user has to run a series of separate simulations to derive the members of the Pareto optimal set, whereas by using EA these members can be found by a single run. There are many EA but some of the major EA based algorithms that can be relevant to complex engineering optimization are GA, Memetic Algorithm (MA), Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO) and Shuffled Frog Leaping algorithm (SFL). The GA is inspired by the biological systems improved fitness through evolution. MA is similar to the GA with a difference of considering the memes and not genes. MA is also different than the GA as they incorporate the local search process to refine the solutions before they get involved in the next evolutionary process. The PSO is based on the behavior of the flock of migrating birds trying to search an unknown destination. Further, the ACO is an agent-based system based on the biological ants and their social behavior of choosing the path to reach the destination. The SFL algorithm combines the benefits of MA and the PSO algorithm, where the population consists of a set of frogs that is partitioned into subsets referred to as memeplexes.

MA, ACO and SFL methods are generally available as a single objective optimization method and have not been used for a multi-disciplinary optimization of a re-entry vehicle. If these single objective optimization methods are chosen, they are prone to failing in the local optimal solution because they update the solution based on their current best path. Out of these, GA and PSO are considered as better options for multi-objective optimization as they are available for multi-objective optimization. This leaves us with PSO and GA as an option for multi-objective optimization of a re-entry vehicle.

As discussed in Section 1-1-2, Nosratollahi et al. (2010) describe multidisciplinary design optimization of a re-entry vehicle using GA. In their study, they performed a comparative analysis of utilizing single-objective genetic algorithm (SOGA) and MOGA for re-entry vehicle design and also derived a single run genetic algorithm to reduce the time of the computation. For n objectives, SOGA requires (n+1) GA runs, whereas for the multi-objective genetic algorithm can generate the Pareto optimal solution in a single simulation run. During this study, they also performed an analysis where every configuration respected to objective optimization individually and objectives optimization simultaneously. Using the single objective optimization they derived solutions which were optimal separately with respect to Cd, Heat Flux and Mass and one case which was optimal considering all objectives. Such comparative study is interesting for a feasibility study of the re-entry vehicle and understands the influence of each objective on the re-entry vehicle shape. Priyadarshi and Mittal (2010), have performed the multi objective multi-disciplinary design optimization of semi-ballistic re-entry vehicle, where they utilized multi-objective genetic algorithm. The results indicated that MOGA is successfully applicable complex problems like optimization of re-entry vehicle. The recent publication by Adami et al. (2015) performed multidisciplinary design optimization of a de-orbit maneuver considering propulsion, TPS and trajectory using GA as MDO technique and AAO as MDO problem formulation method. Lavagna et al. (2012) describes the PSO application for the planetary atmosphere entry vehicle multidisciplinary guidance design, where it has been successfully implemented for a multi-objective optimization in the distributive optimization framework. These research papers give the confidence that EA, can be utilized for the multi-disciplinary re-entry vehicle. Thus, GA and PSO are used as MDO techniques for re-entry vehicle application.

Hassan et al. (2005) describe a comparison between PSO and GA. PSO is a relatively recent heuristic search method. PSO is similar to the GA as they are both population-based search approaches and that they both depend on information sharing among their population members to enhance their search processes using a combination of deterministic and probabilistic rules. Conversely, the GA is a well-established algorithm with many versions and many applications. The PSO and the GA on an average yield the same effectiveness (solution quality) and PSO are more computationally efficient (uses less number of function evaluations) than the GA.

Additionally, from Section 1-1-2, it is observed that GA is more often used successfully for re-entry vehicles, which included shape as well as trajectory optimization in a combination of AAO as MDO strategy. For this thesis, the tools involved are low-medium fidelity tools which do not need high computational time and to solve an industrial problem within the thesis work, GA is preferred over the PSO as both of them yield a same quality solution. Additionally, the Dakota framework provides three multi-objective optimization methods, Weighted sum approach, Pareto-set optimization and MOGA. Here, MOGA is the GA specifically for a problem involving the multiple-objectives. Weighted approach and Pareto-set optimization both converts the multi-objective functions to a single objective which are not preferred methods as discussed above. Whereas the MOGA supports the multi-objective optimization, it is useful and efficient, when the search space is large, complex and poorly known, or no mathematical analysis is available. It is also used for re-entry vehicle optimization, thus can solve complex design optimization because it can handle both discrete and continuous variables and non-linear objective functions without requiring gradient information. MOGA satisfy all the necessary requirements and has been widely used in many complex optimization problems as well as recently been used for the multi-disciplinary optimization of re-entry vehicle. Thus, MOGA is selected method for the optimization. Furthermore, this selected method is described in Section 2-3.

#### Sampling the Design Space

Ridolfi (2013) describes that when one has to study the mathematical model of a system, sampling of the design space is considered as the first step. A sample is a set of points in the design space, where the coordinates of these points are the values of the design variable taken from their variability. The model is executed using each sample point as input and the corresponding output gives the performance of the model. From this output, one can study the details to draw conclusions on the correlation between the input and output and the set of the design factors. One can also explore the design space and get a better understanding of the range in the output, for a particular domain. Additionally, these results are also used to determine which inputs have the most influence on the output, or how changes in the inputs vary the output. These sample inputs and corresponding output can be also used to create the response surface, which is used when running a computational model is extremely expensive.

The Design of Experiments (DoE) are computer experiments performed on the mathematical model in correspondences of the sample points. Adams (2009) describes the sampling methods for the DoE, where it includes the classical DoE methods and stochastic sampling methods. The classical methods include the Central Composite Design (CCD) sampling, Box-Behnken Design (BB) sampling and grid-based sampling method. Furthermore, the stochastic sampling methods include Monte Carlo sampling and Latin Hypercube Sampling (LHS).

The CCD contains an embedded factorial or fractional design with center points that is augmented with a group of star points, that allow an estimate of curvature. If the distance from the center of the design space to a factorial point is $\pm 1$ unit for each factor, the distance from the center of design space to a star point is $\pm\alpha$, where the absolute value of $\alpha \geq 1$. The value of $\alpha$ depends on the properties of the design parameters and number of factors involved. BB is similar to CCD, where BB is a quadratic design and do not contain an embedded factorial. In this method, design treatment combinations are at the midpoints of edges of the process space and at the center, whereas in case of CCD extra points are placed at the star points. BB are considered as rotatable and require 3 level of each factor. BB method is considered better than the CCD as it requires fewer runs than CCD, but as the number of factors increases BB requires more runs as well as BB have limited capability of orthogonal blocking. In the grid-based method, a grid is placed over the input variable space, where the samples are taken over a set of partitions on each variable. This method has disadvantage if a purely structured grid is used, where the samples are exactly on the grid points and unable to capture the important features. Another disadvantage is that number of samples required depends exponentially on the input dimension. Thus, these methods are not considered to be suitable for a complex problem where the input dimension is large.

Adams (2009) describes the Monte Carlo sampling method. The random sampling method or the pseudo-random method used to generate the samples. Monte Carlo technique is also a random sampling method and is used a widely used sampling method. In this method, initially, the input domain is defined. Once the domain is defined, the inputs are randomly generated from the probability distribution over the domain. Then a deterministic computation on the inputs is performed and the results are aggregated. A large number of samples are required for to analyze the results. Furthermore, LHS is a stratified sampling technique for which the range of each uncertain variable is divided into $N_s$ segments of equal probability, where $N_s$ is the number of samples requested. The relative lengths of the segments are determined by the nature of the specified probability distribution. For each of the uncertain variables, a sample is selected randomly for each of these equal probability segments. These $N_s$ values for each of the individual parameters are then combined in a shuffling operation to create a set of $N_s$ parameter vectors with a specified correlation structure. A feature of the resulting sample set is that every row and column in the hypercube of partitions has exactly one sample. Since the total number of samples is exactly equal to the number of partitions used for each uncertain variable, an arbitrary number of desired samples is easily accommodated.

Adams (2009) compares the Monte Carlo method with the LHS method. LHS technique in general, require fewer samples than traditional Monte Carlo for the same accuracy in statistics. Additionally, LHS allows more uniform coverage of the design region of interest, compared to a random sampling method. The LHS provides a more accurate estimate of the mean value than does random sampling. Adams (2009) describes the Design and Analysis of Computer Experiments (DACE) methods, where LHS method generates the samples and furthermore using the Spearman's rank correlation formulation, it computes the rank correlations that are used to analyse the sampling data. Thus, LHS is the method considered as the sampling method and is further described in Section 2-4.

## 1-2 Research Goal

The re-entry vehicles are designed with similar aerodynamic shapes but need to be improved and optimized for a different mission and system requirements. MDO techniques will allow one to perform the feasibility study of such vehicles and derive optimal configuration designed for different mission requirements. This project involves the MDO of re-entry vehicle for cargo retrieval of a scientific experiment or payload from the ISS. The main focus is on the re-entry phase of the mission for the system study, as it is the most critical mission phase and deciding factor to design the Earth return-service spacecraft. Design disciplines like geometry, mass, aero-thermodynamics and trajectory are considered for this study to understand the interaction between them. Other design disciplines are like avionics, propulsion and Guidance Navigation Control (GNC) are not considered to reduce the complexity of the problem. Since the design criteria for a crew-carrying mission is complex and require more design requirements, this study focuses only on the transportation of pressurized cargo from the ISS to the Earth.

The initial design process of IXV followed a traditional approach, where engineers worked sequentially passing the design from engineer to engineer and then followed by iterations. This process eventually resulted to be an expensive methodology in-terms of cost, time and resources. TASI modified partially the design methodology, coupling the three main system disciplines: Mission Analysis, GNC and Aerothermodynamics. In this way, the design process related to trajectory optimization, GNC validation, and the aero-thermal loads definition was drastically accelerated, and the robustness of the approach was significantly improved. Thus, to perform the feasibility study of the re-entry vehicle a Concurrent Engineering (CE) approach, extended to all the involved disciplines, is required from the preliminary design phases.

Even though TASI started its own CE design facility in 2005, it is mostly used for satellite design while the necessary tools and interfaces for its utilization for the design re-entry vehicles are still missing. The mathematical models are used separately to simulate each discipline considering the requirements of the other disciplines as an input. Such methodology eventually is not an efficient way to consider the tight coupling between the complex systems like re-entry vehicles. Thus, there is an industrial requirement to develop a methodology that will integrate all the tools on one single platform and automate the process. There is an on-going research project at TASI considering IXV as a baseline configuration to derive a payload carrying vehicle (Space Rider). Developing a design process to derive the optimized configuration of a re-entry vehicle for the mission and system requirements, considering IXV as a baseline configuration is most appealing. To solve this industrial problem within the work of thesis, the main research question is framed as follows:

*“To what extent can an optimal re-entry vehicle similar to the IXV configuration be developed in the conceptual design phase using an integrated design process, where the vehicle is designed for a specific mission scenario and system requirements?"*

This is the main research question and can only be answered, if the integrated design process to perform the feasibility study of a re-entry vehicle is developed. This design methodology developed within the Dakota framework, will allow one to integrate all the tools on one single platform and automate the design process. Thus, the following two sub-goals are achieved prior to the main research question.

*“To perform the feasibility study of the re-entry vehicles using the MDO techniques, to what extent is it possible to integrate the design disciplines on a single platform and automate the design process within the Dakota framework?"*

It is also important to make the best use of the developed integrated design process within this framework. Currently, even though there are good mathematical models available for each design discipline it is required to consider the time and resources limitation during the conceptual design phase. Therefore, the third research question is formulated as follows:

*“How and to what extent can the design techniques available within the framework, assist the engineering team during the conceptual design of complex systems to obtain better, faster and eventually low-cost design process?"*

If the above research questions are answered, then an industrial problem within the work of a thesis, can be solved, which is also the motivation of the project. Even though it is challenging, this project is considered as a feasible project as a similar approach has been successfully implemented in the aircraft design sector. The CDF of an industry can use this methodology for the analysis of the re-entry vehicle at the conceptual design phase. Solving these research questions will provide a design process that will be efficient, faster, cheaper and more robust than the traditional approach, to investigate a wide range of conceptual re-entry vehicle designs and derive an optimized design for different mission scenarios.

## 1-3 Outline of Thesis

To answer the above research questions, the following chapters are considered. Chapter 2 describes the integration methodology developed within the Dakota framework. Furthermore, Chapter 3 describes all the tools considered for the thesis and highlights their unit test verification. Once all the required tools are verified, Chapter 4 describes the workflow created for the feasibility study of the axis-symmetric Raduga-like capsule. This chapter also highlight the system level verification, where it discuss the results of the sensitivity analysis and the optimization of Raduga-like configuration. This verified design process is then extended for the lifting body re-entry vehicle. Chapter 5 describes the trajectory optimization of the IXV configuration using the developed methodology. Later on these verified workflows for the geometry and the trajectory optimization are clubbed together to answer the main research question. Additionally, a new tool for the shape morphing of a IXV configuration and its mass estimation is introduced in the workflow, thus Chapter 6 describes the workflow to answer the main research question and furthermore highlights the results obtained. Chapter 7 describes the conclusions drawn from the results of this thesis and the recommendations that will assist in future work related to this thesis.

Chapter 2

# Integrated Design Process

The first step in any design process is to define the need and the mission statement; then identify mission and systems requirements. Thus, a system engineering approach is followed to derive the design criteria of the project.

- **Need Statement:** There is a need of frequent logistics supply chain to maintain the ISS as well as to supply and return the experimental and scientific data from the ISS to the Earth.
- **Stakeholders**: Student, Delft University of Technology (DUT) and TASI.
- **Mission Statement:** Developing a re-entry vehicle that can meet the need of transporting cargo from the ISS to the Earth.

To satisfy the mission statement, we need to design a spacecraft with the Earth-return capabilities. The main objective of the thesis is to derive an optimized re-entry vehicle similar to the IXV configuration during the conceptual design phase, where the vehicle is designed for a specific mission scenario and system requirements. To perform the multidisciplinary optimization of a re-entry vehicle, it is important to integrate all the design disciplines on a common platform to automate the design process. Here, the Dakota framework is used as an integration design environment (framework) to develop the design process that enables the communication and interaction between each of the design discipline. Thus, the mission objective is defined as follows:

- **Mission Objective:** Derive an optimized design configuration of a re-entry vehicle using the design process developed within the Dakota framework for the multidisciplinary design optimization of a complex engineering system like a re-entry vehicle.

Section 2-1 describes the architecture of the Dakota framework. Furthermore, this chapter highlights the system requirements related to integration methodology in Section 2-2-1. From these requirements, the design process for the multidisciplinary optimization of a re-entry vehicle is developed within the given framework. Section 2-2 describes the design methodology for the integration of all the design disciplines on a common platform. This section also highlights the generalized interface template created for the flow of information within the workflow. Furthermore, to answer the main research question, it is required to derive an optimized design configuration of a re-entry vehicle, using the MDO techniques. As described in Chapter 1, MOGA is considered as an optimization technique. Section 2-3 describes the MOGA and furthermore highlights the verification of this optimization technique. Furthermore, prior to performing the MDO of the re-entry vehicle, it is important to perform the Design of Experiments (DOE), such that the user can derive the possible tendency of the data. This tendency of the data should be derived using the less number of sample points. As described in Chapter 1, LHS method and the Spearman's rank correlation is considered for the sensitivity analysis and is described further in Section 2-4.

## 2-1 Software Architecture

To derive a top level software architecture, one must initially look into the architecture of the chosen integrated design environment, i.e., the Dakota framework. Adams (2009) describes the user manual of the Dakota framework. There is a requirement of computational models in all engineering designs and scientific activities. The computational models are used for the simulation of a complex physical system. This simulator used as a virtual prototype, where the system parameters are tuned to improve the performance of a system. The system is defined by one or more system performance cost functions. During the optimization, this virtual prototype runs the simulator to evaluate the cost function(s) and select the appropriate system parameters in an iterative way, such that the process is automated and direct.

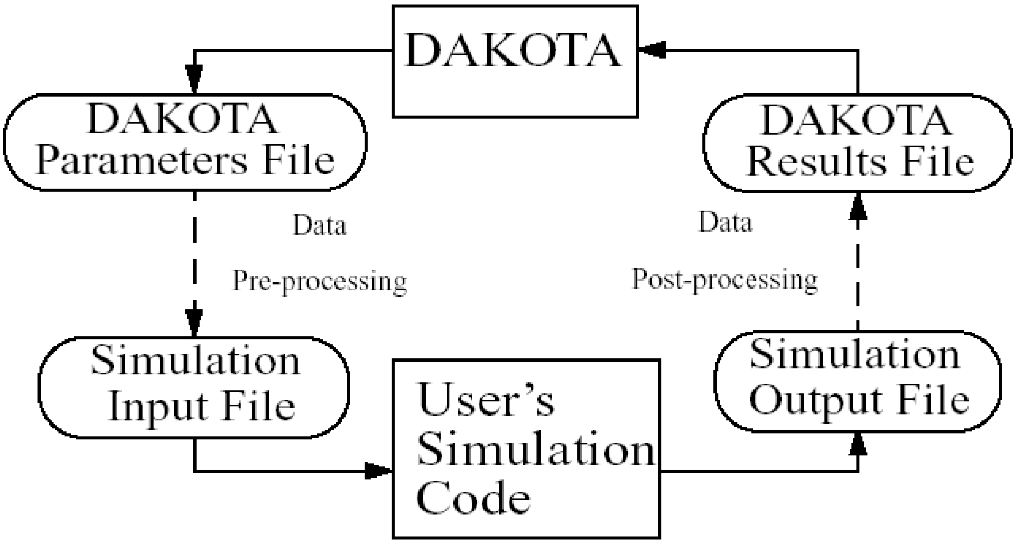



**Figure 2-1:** The loosely-coupled relationship between Dakota and a user-supplied simulation code, (Adams, 2009)

Figure 2-1 describes a typical loosely-coupled relationship between Dakota and the simulation code(s). For this coupling, Dakota does not require any internal details of the computational models. This eliminates the need for the source code, and such coupling can be also called as a black-box. Such coupling provides a simple interface approach and allows the user to use the Dakota framework for different applications. The Dakota block as shown in Figure 2-1, provide engineers standard algorithms for the parameter study, design of experiments, optimization, uncertainty quantification and calibration of a system.

The exchange of the data between the Dakota and the simulation code is performed by reading and writing short data files. The commands provided in the Dakota input file are used to execute the Dakota simulator. This input file includes the information of the type of analysis to be performed, along with the file names associated with the user's simulation code. The user's simulation code is executed automatically through a separate process. This process is external to Dakota. In Figure 2-1, the solid line describes the file input/output operations, which are standard for all the coupled relationship between Dakota and the user's simulation code. The input/output files format of Dakota and the simulation code is standard. It is important that the simulation code can read the output from Dakota and vice-verse to generate a closed loop. The dotted lines indicate the conversion of the information from one format to another, such that Dakota and the simulator can interact with each other. A parameter file is generated by Dakota, which is passed to the simulator to perform the simulations and generate a results file. This results file is supplied back to Dakota to continue the iteration process. This computation continues till a converged solution is obtained. For any type of analysis to be performed, within the Dakota framework, these operations are considered as standard.

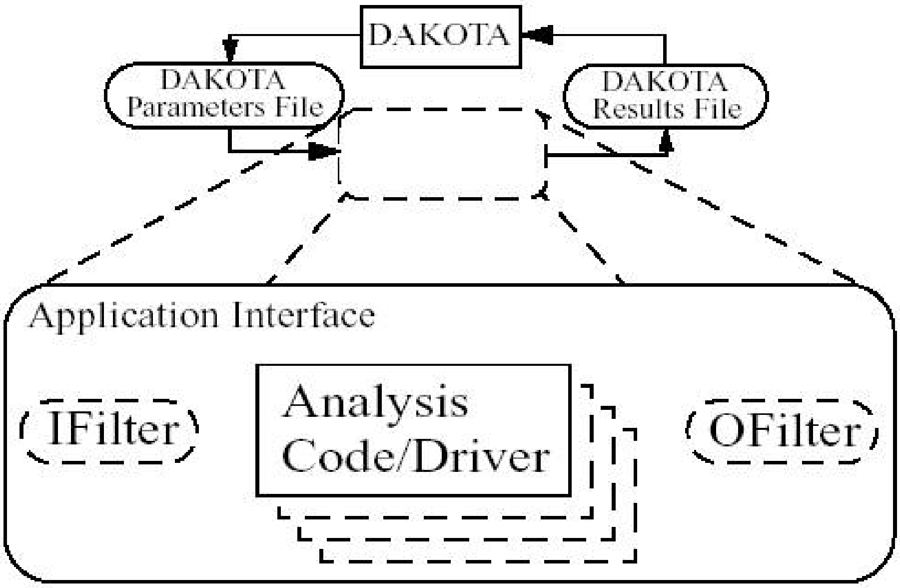



**Figure 2-2:** Components of the simulation interface,(Adams, 2009)

Figure 2-2 is similar to Figure 2-1, where the details of the components that make up each of the simulation interfaces are added. Here the simulation interfaces indicate the type of interface between Dakota and simulator code. The system calls, forks or via direct linkage are used to call the simulation code. In case of system call and fork, simulation and the interface between Dakota and the simulation code occurs through parameter and responses file. This process is separate from Dakota and thus, the user can implement their own computational models in the simulation code, whereas for the direct linkage case, a separate process is not created and flow of information between Dakota and the simulator occurs through Application Programming Interface (API). For the re-entry vehicle application, one needs the external simulation code thus, the direct interface cannot be used. The fork interface supports additional capabilities than system interface and Adams (2009) has encouraged the users to use the fork simulation interface over the system call, when managing asynchronous simulation code execution. For the re-entry vehicle application, there are multiple analysis drivers (tools), thus fork simulation interface is used.

In Figure 2-2 the additional components include an input filter (IFilter), analysis driver/code and output filter (OFilter). There can be single or multiple analysis codes and corresponding input and output filter. The input filter is used to extract the parameters required by the input file of the similar program. In the case of a re-entry vehicle application, where more than one tool will be included in the simulator code, the input filter will be responsible for extracting the necessary data for each tool from the Dakota parameter file and provide it, to the input files of each tool included in the simulator code. Similarly, the output filter can be used to extract the output data each tool within the simulator code and compute the desired response data set.

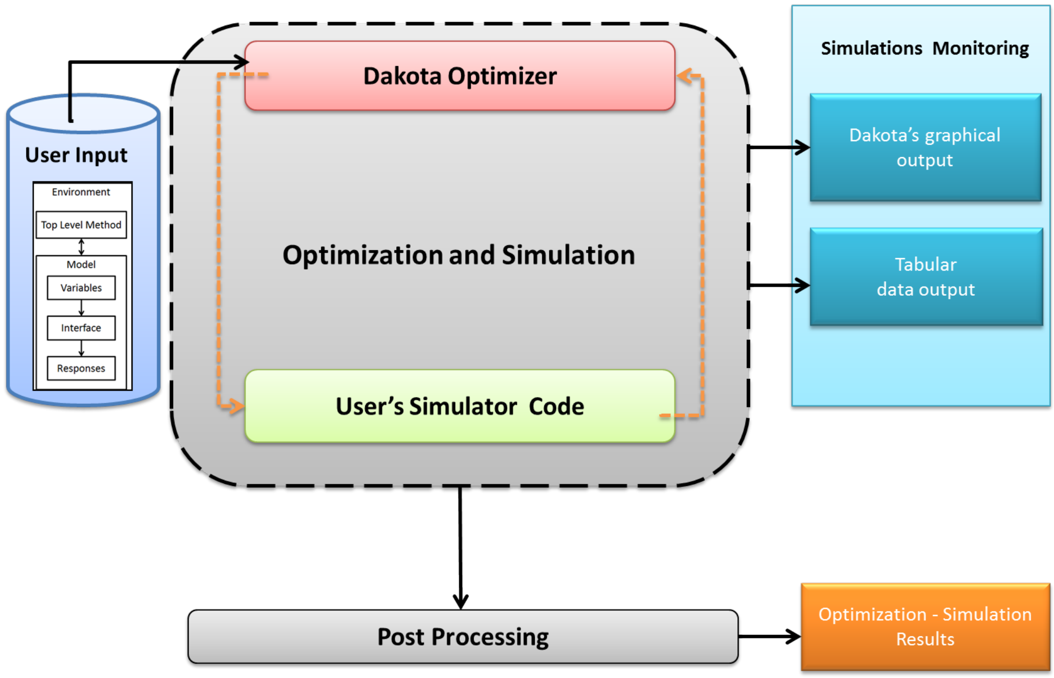



**Figure 2-3:** Top level architecture of the main program

Figure 2-3 gives the top level architecture of the main program. This architecture includes blocks like user input, optimization and simulation, simulation monitoring, post-processing and optimization simulation results. The user input block is composed of input block, environment block, method block and response block. The user can define the design parameters and constraints in the Dakota input file. The environment block, specify the setting for the graphical and the tabular data output. In the method block of the input file, the user can mention the iterative method that Dakota should use to perform the feasibility study of a system. The model block of the input file specifies if single or multiple models will be used during the process. In the variable block, the user can define the characteristics of the parameters that defines the problem or the system. In the interface block defines, the user can define the approach to be used to map variables into responses. This also indicates the details on how the information will flow from the Dakota to simulator and vice-verse. In the responses block, the user defines the data that the interface will give back to Dakota to continue the iteration process.

The Dakota input file and the shell script are required to execute simulations. Dakota can also execute the simulation code, which is external to the Dakota and developed by the user(s) using the fork simulation interface as discussed previously. Thus, one can adapt this framework for re-entry vehicle application. The Dakota input file provides flexibility to change the methods or approach to be followed for the feasibility study. Thus, to perform the different type of analysis such as parameter study, optimization, sensitivity analysis, uncertainty quantification requires changes in the input file. This input file is also used to define the characteristics of design parameters, objective functions, and constraints considered for the optimization process. Using this input file the Dakota creates the Dakota parameter file called 'params.in'.

Furthermore, the Dakota shell script is used to define the task performed within the simulator. This file indicates the steps taken to copy the necessary files in the working directories, creating an input file for each tool in the workflow, executing the tools and writing their output files. It also indicates the task to write the overall simulation output file. The Dakota framework does not need any knowledge regarding the software inside the simulator. The shell script only requires the executable file that can run in the loop for the given design parameters. Thus, developers can provide only their executable file without any internal information of the tool, which makes this coupling as a black-box.

In the case of the complex engineering problems, more than one tool/software are considered in the workflow. Thus, it is important to automate the design process by creating necessary interfaces within the simulator as well as between Dakota and the simulator code. The Ifilters are responsible for converting the format of the Dakota parameter file into the required format for each tool. Similarly, the Ofilters are responsible for converting the data format of the output file from each tool to the data format of the Dakota results file. For each parameter file generated from Dakota, the simulations are performed and the response in the form of the results file is returned back to Dakota to generate new parameter file. This is required to create a closed loop where all the tools and Dakota can communicate with each other.

Dakota can perform the optimization for the user supplied simulator code. During the simulations graphical as well as tabular results are given as an output to monitor the optimization process. For the implemented MDO strategy, Dakota provides the best optimal design solutions in its output file. Dakota does not have any features of post processing of the obtained optimal solutions, but they can be extracted into Matlab to perform the post processing. From the derived results, the best-optimized solution is chosen that meets the mission and system requirements.

## 2-2 Integration Methodology

Using the traditional engineering approach to design a complex system the different disciplines can autonomously affect the system performance. In this approach, there is a sequential design cycle, where the interaction between the design activities and the results are confined within each discipline. The single discipline transfers its verified results to the other discipline that require them. Iterations are required, whenever the results of one discipline affect the inputs of a previously run discipline. The iterations are performed until the coupling variables converge to within acceptable tolerance. Being, such an iteration accomplished by each disciplinary expert physically waiting and receiving his appropriate inputs, performing his analysis and then physically giving the results to the other experts that require them. Using this approach for deriving a single converged design, can take a great deal of time with consequently unacceptable penalization in cost.

In the case of a multidisciplinary design process, the tools/modules for each discipline is developed by various industries and research institutes. Traditional design approach allowed these developers to pass on their results to each other, but in case of concurrent design approach, all the design disciplines are required to be integrated on a common platform. The concurrent approach makes all those issues less critical, which involves manufacturing, process, planning, assembly and cost when addressed early in the design process. During the in-process design process, the changes made can propagate and are accessible to various cross-functional team for subsequent corrective actions.

Considering the time and resources available, if the process fails it is difficult for a user to debug the error without getting the access to each tool. The user can contact the developer, but its a time-consuming process and if the user has access to the tools he/she has to face the issues of complexities, as the tools are developed by different developers. In this case there is a requirement of a generalized method where the user can access the necessary inputs given to the tool as well as the output generated by the tool, without going into the details of the code. Such a methodology will allow the user to check with the inputs and outputs to/from tool and will be useful to debug the errors within the design process. For the re-entry vehicle application, where multiple disciples are required to be integrated on a common platform, the developed integration methodology should satisfy the requirements described in Subsection 2-2-1.

### 2-2-1 System Requirements of the Integration Methodology

To optimize the complex engineering systems, a methodology is developed within the Dakota framework. The system requirements for methodology developed within the framework are described below:

1. **The integrated tool requirement**
    - There shall be a freedom to utilize specific software for each discipline.
    - There shall be a flexibility to choose the level of analysis depending on the accuracy and the cost.
    - There shall be a data management and visualization infrastructure necessary to handle a large amount of data that is generated during the multidisciplinary design process.
    - It shall have dedicated interfaces with discipline databases for easy access during the process.
    - There shall be communication among the disciplines for data exchange and design variables.
    - There shall be a capability of the automatic data transfer.
    - There shall be a flexibility to define a strategy optimization based on the selection of problem formulation, algorithms, and computational power resources.
    - Human intervention for decision making should be possible in the process.
2. **Architectural design requirement**
    - Architecture should be object-oriented approach, where plug and play of all the involved discipline tools should be possible.
    - It should be extensible, such that it can allow the addition of any new features to the tool.
    - The tool should be able to manage a large number of design variables and constraints.
3. **Problem construction requirement**
    - High-level programming that is complex branching and iterations should be possible.
    - There should be fast and efficient debugging capabilities.
4. **Information access requirements**
    - Database management should be efficient as there is a great amount of data generated during the simulation.
    - It should be capable of producing a visualization of the results with any appropriate graphical tools.
    - It should be possible to monitor the MDO process at any phase of the execution.

### 2-2-2 Description of the Integration Methodology

This subsection describes the integration methodology that can satisfy the above requirements. The design process for the feasibility study of a system requires various numerical models to be integrated in the workflow. The results from one numerical model influence the results of the sequential tool's inputs and the response function. The design process to be developed within the Dakota framework should be able to manage these inputs and the response function. Based on this theoretical background a generalized interface template is created to develop the design process. Dakota includes a function called dprepro, which is used by the Dakota optimizer to write its design parameter file and read from the Dakota results file. To utilize this function for the complete workflow, all the output files from each tool should be in the same format as the Dakota parameter file. Figure 2-4 gives the generalized interface template created to use the dprepro function.

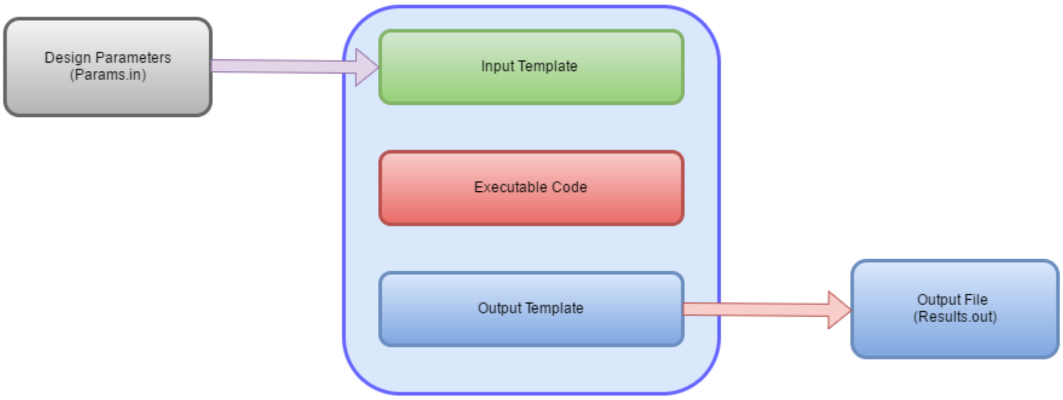



**Figure 2-4:** Interface template

Thus, to develop a generalized approach, dprepro is used to extract the data from the parameter file and write the required values in the input template of the respective tool. In Figure 2-4, it can be seen that the inputs required by the tool are extracted from the Dakota design parameter file (params.in) using this dprepro function. The Dakota shell script is written to inform Dakota about the steps to be taken during the complete process. This shell script is responsible for automating the design process.

Once the inputs values are written in the template, it is saved with a different name and then this new file is executed using the shell script. Once the output are generated from the tool, these are written back in the same format in the output template. To keep dprepro as a common function to read and write the input and output files, it is ensured that the output produced from the tool is written in format which dprepro can read. Thus, dprepro function is used for the transfer of information from one tool to other. This final output file of each tool is saved which describes the inputs considered for the simulation and output values. This gives the flexibility to the user to check and debug the errors. Thus, this methodology becomes independent of the programming languages used to design a tool, as well as if the only the executable file is provided by the developer this same tool becomes a black-box for the user.

This interface template is used as reference for all the tools. Figure 2-5, shows the workflow that describes the integration of more than one tool. This workflow describes how the flow of information within the complete closed loop takes place. The flow of information takes place from the Dakota through the complete tool chain within the simulator and then the required information is passed back to the Dakota. The output generated from each tool is also written in a format such that dprepro can be read it and the user can use the same function to fill the templates of the other tools in the tool chain.

The deprepro function can be used to extract the data from multiple files, thus if any tool requires input from more than one tool this same function can be used to fill the templates of the respective tool. Thus, the design process is automated. The generalized work flow is developed, which is user friendly and allows the user to perform the feasibility study, even if the tools or software provided by the developers are a black box. Thus, it satisfy the requirement of communication among the disciplines for the data exchange and design variables. Additionally, the use of input and output interface templates for each tool, gives the user easy access to the inputs used for each tool and output generated by them, which allows the human intervention to easily debug the problem in the design process.

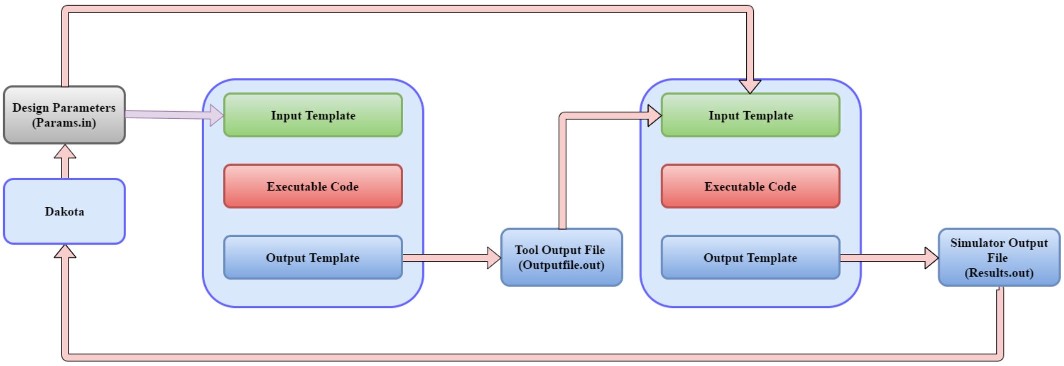



**Figure 2-5:** Workflow

As the interface templates are generalized, there is freedom to plug and play different tools in the workflow based on the requirements. Similar input and output interface files can be created for any tool and thus gives a freedom to utilize specific software for each discipline based on the level of analysis, accuracy and the cost.

The Dakota framework provides data management and visualization infrastructure at any phase of the execution of the MDO process. Thus, allows the user to monitor the process. Additionally, the data is stored in the tabular data that can be easily imported in the Matlab for further analysis. The input file of Dakota allows the user to select the optimization strategy based on the selection of the problem formulation, algorithms and computational power resources. Furthermore, once this design process is developed within the framework to perform the multi-objective optimization a MDO technique called MOGA is selected as discussed in Chapter 1. The following section describes this optimization technique and furthermore gives the verification of this optimization technique.

## 2-3 Optimization Technique

EA is suitable to solve multi-objective optimization problems. The methods from EA category deal simultaneously with a set of possible solutions as discussed in Chapter 1. These methods allows the user to derive several members of the Pareto optimal set in a single run, whereas one has to perform a series of separate runs as in the case of the traditional mathematical programming techniques. MOGA satisfies the necessary requirements and has been widely used in many complex optimization problems as well as recently been used for the multi-disciplinary optimization of re-entry vehicle. As described in Chapter 1, the MOGA is considered for this thesis.

### 2-3-1 Description of the MOGA

Adams (2009), describes MOGA algorithm available within the Dakota framework. The MOGA method can give a correct and accurate identification of the whole Pareto front. This method is based on the idea that as the population evolves in a GA, solutions that are non-dominated are chosen to remain in the population. This approach works effectively on multi-objective problems. In this method the user can avoid the problem of converting a multi-objective problem to a single objective and thus avoid the problems with aggregating and scaling objective function values. This method provides a fitness assessor, which works by ranking population members. These members are ranked such that their resulting fitness is a function of a number of other designs that dominate them. Considering the fitness of the members, the selector chooses the designs. If a design is dominated by more than a specified number (user-defined limit) of other designs, then it is discarded by the selector, otherwise, it is selected for the next generation. This algorithm also gives the flexibility to determine the minimum number of the selections that will be take place if enough design variables are available. That is, one can chose the percentage of the population size that must go on to the subsequent generation. If there are less possibilities of population that should actually go to the next population, then it automatically relaxes this limit and choose the designs from the remaining set.

Figure 2-6 describes the tuning for the MOGA considered in the input file. The method for the feasibility study and the options within the method are selected by stating their Dakota keyword in the input file, as described by Adams (2009). The MOGA method is chosen by indicating its keyword in the method block of the input file.

```
moga
output debug
  seed = 20000
  final_solutions = 30
max_function_evaluations = 2500
initialization_type unique_random
crossover_type shuffle_random
  num_offspring = 2 num_parents = 2
  crossover_rate = 0.8
mutation_type offset_uniform
  mutation_scale = 0.15
  mutation_rate = 0.08
fitness_type domination_count
replacement_type elitist
convergence_type metric_tracker
  percent_change = 0.1 num_generations = 5
```

**Figure 2-6:** Tuning within the MOGA, (Adams, 2009)

The output option controls the amount of information presented to the user. The output file includes the final solutions derived after optimization. The number of final solutions is defined by the user in the option final solutions. In case, the user needs to see the selected design parameters and the corresponding values for the objective functions and constraints involved in the optimization for each simulation, one needs define this in the output option. This amount of information is classified in the 5 levels: silent, quiet, normal, verbose and debug. The debug, gives the maximum information to the user, whereas the silent gives the minimum information. With respect to the normal, quiet reduces parameter and response set reporting, whereas the silent further suppresses function evaluation headers and scheduling output. The verbose adds the details of file management, approximation evaluation, and global approximation coefficient with respect the normal. Furthermore, debug adds the diagnostics from non-blocking schedulers. In case the user does not select any option, normal is considered as the default option.

The random seed control the mechanism for making a stochastic method repeatable is provided by the seed option. The maximum function evaluation option describes the stopping criteria of the simulations. If convergence criteria is not mentioned, the simulations stops at maximum evaluations, but if it is mentioned it stops when solutions converge. The maximum function evaluation is defined by the user. This method works by using the following steps:

- **Initialize the population:** MOGA starts with the initialization step, where the user can specify how to initiate the population. Simple random, unique random and flat file are the three different types of initialization. Simple random creates the initial solutions

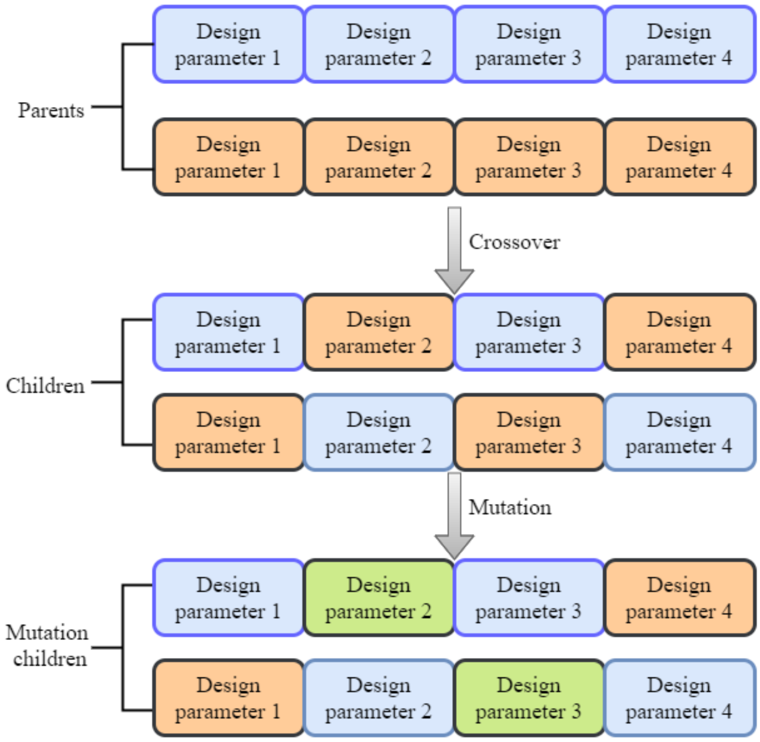



**Figure 2-7:** Initilization, crossover and mutation of the population in MOGA

with random variable values according to a uniform random number distribution and gives no consideration to any previously generated designs. Furthermore, in the unique random initialization method, the new solution is checked against the rest of the solutions, to avoid duplication of a solution. Furthermore, the flat file allows the initial population to be read from a flat file, which can be used for a hybrid optimization. Here, the hybrid optimization includes more than one method used in a sequential order such that the results from the first method are considered as input for the second method. Here, the unique random is considered as shown in Figure 2-6.

- **Evaluate the population:** Once the initial population is available, this is evaluated by computing the corresponding value of the objective functions and constraints.
- **Loop until converged, or stopping criteria reached:** Using this initial population, a new population is derived through crossover and mutation. Figure 2-7 describes the crossover and mutation of the design parameters. During the crossover, the bits of the parent's population are swapped with each other. This new population formed after the crossover is called as children. These children are then used for mutation. During the mutation, the bits of the children are varied to form a new mutated children population. The following steps are followed in the loop:
  - **Perform crossover:** Multi-point binary, multi-point parametrized binary, multi-point real and shuffle random are the four types of crossover method available for MOGA within the Dakota framework. Multi-point binary performs a bit switching crossover at N crossover points, multi-point parametrized binary performs a bit switching crossover routine at N crossover points but performs crossover on each design variable individually. Furthermore, multi-point real crossover performs a variable switching crossover routing at N crossover points in the real-real value genome of two designs. The final crossover type is the shuffle random. This crossover type performs the crossover by

choosing the design variables at random from a specified number of parents enough times that the requested number of children are produced. This type is useful when there are more design parameters and the user is unable to define which parameters are crossed. In the crossover setting the user is allowed to define the number of parents and child considered for the algorithm. The crossover rate is also defined by the user, which is used to calculate the number of crossover operation that takes place. The number of the crossover is equal to the rate times the population size.

– **Perform mutation:** Once the crossover is performed the mutation process is carried out. The Bit random, replace uniform, offset normal, offset cauchy, offset uniform are the five types of the mutation available for MOGA within the Dakota framework. The bit random mutator introduces random variation. The randomly chosen variable is first converted into a binary string and is then flipped in the string from 1 to 0 or vice-verse. In this type of mutation, there is more probability of being similar to the original value. Furthermore, the replace uniform type of mutation introduce random variation by first randomly choosing a design variable of a randomly selected design and re-assigning it to a random valid value for that variable. No consideration of the current value is given when determining the new value. Furthermore, the offset normal mutator introduces random variation by adding a Gaussian random amount to a variable value. Furthermore, the offset Cauchy mutator introduces the random variable by adding a Cauchy random amount to a variable value. In case of offset normal and offset Cauchy, the random amount has a standard deviation dependent on the mutation scale. The mutation scale is a fraction in the range of 0 to 1, and it controls the amount of variation that takes place when a variable is mutated. The final mutator type is offset uniform. This mutator, introduces random variation by adding a uniform random amount to a variable value. The random amount depends on the mutation scale. The mutation rate controls the number of mutation performed.

– **Combine population, evaluate the new population:** Furthermore, the parents and the mutation children are combined to form a new population.

– **Assess the fitness of each member in the population:** This new population is evaluated by the fitness assessor. Layer rank, below limit and domination count are the three fitness assessor available for MOGA within the Dakota framework. The layer rank fitness assessor works by assigning all non-dominated designs a layer of 0, then from what remains, assigning all the non-dominated a layer of 1, and so on, until all designs have been assigned a layer. Additionally, the below-limit fitness assessor has the effect of keeping all those designs whose layer is below a certain threshold again subject to the shrinkage limit. Furthermore, the domination count fitness assessor works by ordering population members by the negative of the number of designs that dominate them. The higher the fitness number, the better is population.

– **Replace the population with members selected to continue in the next generation:** Once, the fitness of the population is assessed, it is required to replace the population with members selected to continue in the next generation. The pool of potential members is the current population and the current set of offspring. Elitist and below limit selector are the two selectors available for MOGA within the Dakota framework. The Elitist selector simply chooses the required number of designs taking the fittest, whereas below-limit selector attempts to keep all designs for which the negated fitness is below a certain limit.

– **Test for convergence:** This process of crossover, mutation, assessing the new population and selection of the new population for next generation continues until the convergence criteria are met. Metric tracker is used for the convergence, where the percentage change in the generation and number of generations considered for the convergence. The tracker observes the variations from generation to generation in the non-dominated frontier. As these changes fall below a user specified threshold, for a user specified number of generations, the algorithm stops and the solution is considered as converged solution.

- **Perform post processing**

## 2-3-2 Verification of the Optimization Methodology

To successfully implement the optimization methodology in the workflow, it is important to verify this method within the Dakota framework. To verify this MOGA, the solution derived for a given problem from the Dakota framework is compared results mentioned in the Dakota manual by (Adams, 2009). The problem is defined with 2 objective functions $f1$ and $f2$ which are the function of $x1$ and $x2$, variables. These input variables are bounded by -20 $\leq$ $x_i$ $\leq$ 20, where [i=1,2]. The following are the objective functions of the given problem:

$$f_1(x) = (x_1 - 2)^2 + (x_2 - 1)^2 + 2 \tag{2-1}$$

$$f_2(x) = 9x_1 - (x_2 - 1)^2 \tag{2-2}$$

The following are the constraints of the given problem:

$$0 \leq x_1^2 + x_2^2 - 225 \tag{2-3}$$

$$0 \leq x_1 - 3x_2 + 10 \tag{2-4}$$

The input file defines the number of optimized solutions required for the multi-objective problem. In this case 3 optimized solutions are derived. Figure 2-8 gives the solutions along with the three optimized solutions derived using the MOGA.

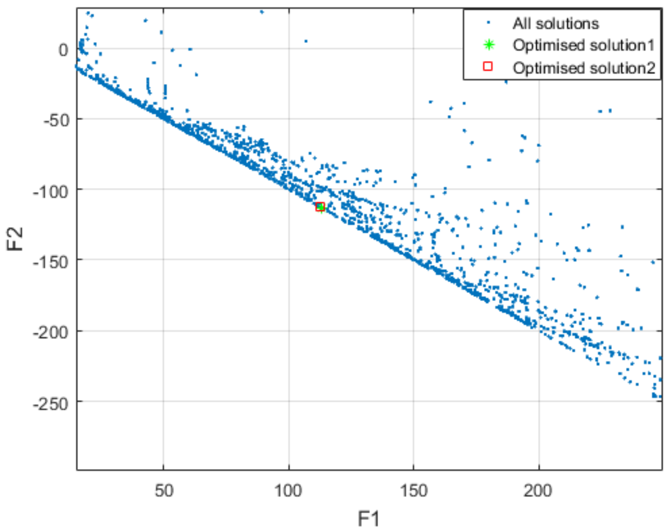



**Figure 2-8:** Pareto front showing tradeoffs between function f1 and function f2

**Table 2-1:** Results obtained from MOGA, (Adams, 2009)

| $x_1$ | $x_2$ | f1 | f2 | C1 | C2 |
|---|---|---|---|---|---|
| -2.62 | 10.46 | 112.92 | -113.16 | -108.65 | -24.01 |
| -1.99 | 10.72 | 112.48 | -112.48 | -106.06 | -24.16 |
| -3.09 | 10.37 | 115.80 | -115.70 | -107.80 | -24.21 |

**Table 2-2:** Results obtained from MOGA

| $x_1$ | $x_2$ | f1 | f2 | C1 | C2 |
|---|---|---|---|---|---|
| -2.62 | 10.46 | 112.92 | -113.16 | -108.65 | -24.01 |
| -1.99 | 10.72 | 112.48 | -112.48 | -106.06 | -24.16 |
| -3.09 | 10.37 | 115.80 | -115.70 | -107.80 | -24.21 |

Table 2-2 describes the results derived from the Dakota framework which is furthermore compared with the reference results given in Table 2-1. The reference results are taken from the Dakota manual given by Adams (2009). It is seen from the above tables, that the results match with the results mentioned in the Dakota manual, Adams (2009). Furthermore, Chapter 4 and Chapter 5 describe the verification of design process using the MOGA as optimization technique for the axis-symmetric Raduga configuration and the trajectory optimization of the IXV configuration, where the tuning of the MOGA is discussed for the re-entry vehicle application.

## 2-4 Sensitivity Analysis

Adams (2009) describes the more modern DACE methods, which seek to extract as much as trend data from a parameter space as possible using a limited number of sample points. Using these data the user is then required to analyse these data to answer the following questions:

- Which factors contribute to the results and how much?
- What is the optimum condition with respect to quality?
- What will be the expected result at the optimum condition?

There are many goals of running a computer experiment: one may want to explore the input domain or the design space and get a better understanding of the range in the output for a particular domain. Another objective is to determine which inputs have the most influence on the output, or how changes in the inputs change the output. This is usually called sensitivity analysis. These sample input and corresponding output can be also used to create the response surface, which is used when running a computational model is extremely expensive.

As described in Chapter 1, LHS is considered for the design of experiments that can accurately extract the trend information. It is a stratified sampling technique for which the range of each uncertain variable is divided into $N_s$ segments of equal probability, where $N_s$ is the number of samples requested. The relative lengths of the segments are determined by the nature of the specified probability distribution. For each of the uncertain variables, a sample is selected randomly for each of these equal probability segments. These $N_s$ values for each of the individual parameters are then combined in a shuffling operation to create a set of $N_s$ parameter vectors with a specified correlation structure. A feature of the resulting sample set is that every row and column in the hypercube of partitions has exactly one sample. Since the total number of samples is exactly equal to the number of partitions used for each uncertain variable, an arbitrary number of desired samples is easily accommodated.

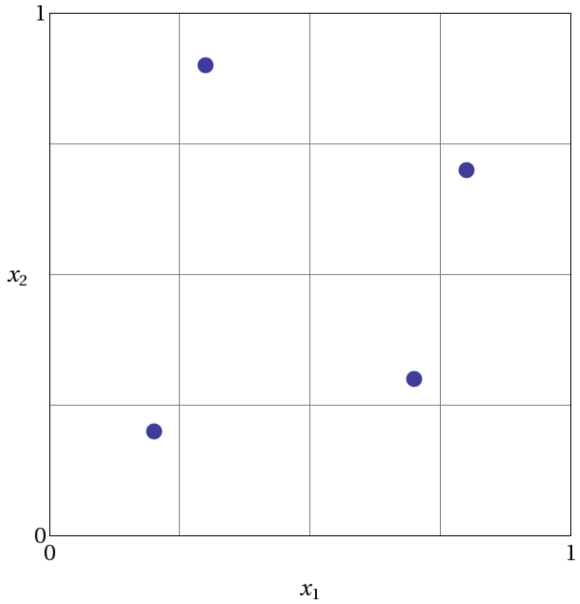



**Figure 2-9:** An example of Latin hypercube sampling with four bins in design parameters x1 and x2. The dots are the sample sites, (Adams, 2009)

Figure 2-9 demonstrates the LHS on a two-variable parameter space. Here x1 and x2 varies in the range between [0,1] and have uniform statistical distributions. The range of each parameter is divided into p bins of equal probability. For the uniform distribution parameters, these partitions are of equal size. Thus for n design parameters, this partitioning gives a total of $p^n$ bins in the parameter space. Next, p samples are randomly selected in the parameter space, with the following restrictions:

- Each sample is randomly placed inside a bin.
- For all one-dimensional projections of the p samples and bins, there will be one and only one sample in each bin.

In this method, only one bin can be selected in each row and column. Thus, for example if p=4, there are four partitions in both x1 and x2, which gives total of 16 bins. The 4 samples are chosen according to the criteria described above. There is more than one possible arrangement of bins that meet the LHS criteria. Figure 2-9 gives the example, where the dots represents the four sample sites, where each sample is randomly located in its bin. All the parameters must have the same number of bins, while there is no restriction on the number of bins in the range of each parameter. The analysis and post-processing of the results is further done by the following ways:

- **Data Mining:** Better understanding of results through trade-off and by viewing the design space from multiple points of view.
- **Correlation Map:** Understand and plot the graphs for the correlation between the parameter sets.
- **Visual Design Driver:** Visualize approximations by interactively surfing the design space.

The LHS method within the Dakota framework also gives the simple and partial raw correlations; and simple and partial rank correlations. Adams (2009) describes that the raw correlations and partial rank correlations. Here, the raw correlations refer to correlations performed on the actual input and output data, while the rank correlations refer to correlations performed on the ranks of the data. Ranks are obtained by replacing the actual data by the ranked values, which are obtained by ordering the data in ascending order. Adams (2009) describes the following correlation coefficients:

**Pearson's correlation coefficient** is a measure of the linear association between two variables. This coefficient is denoted by r. It accounts for each variables measure, that is the interval or the ratio scale in which variables are expressed. When a non-linear relationship exists between the variables this correlation coefficient can be misleadingly small. The procedures, based on r, for making inferences about the population correlation coefficient, make the implicit assumption that the two variables are jointly normally distributed. Other non-parametric measures such as the Spearman rank correlation coefficient should be used if this assumption is not possible.

$$r_{xy} = \frac{\sum_{k=1}^{N}\left(X_k - \bar{X}\right)\left(Y_k - \bar{Y}\right)}{\sqrt{\sum_{k=1}^{N}\left(X_k - \bar{X}\right)^2}\sqrt{\sum_{k=1}^{N}\left(Y_k - \bar{Y}\right)^2}} \tag{2-5}$$

Where, $\bar{X}$, $\bar{Y}$ are means values of X and Y respectively.

**Spearman rank correlation coefficient** is used when it is not convenient, or even possible to account for actual values of variables. It is denoted by s. It is in fact defined by assigning a rank order to each evaluation of the variables. The rank array of a variable is the array of integers that specifies the numerical orders of the values of the variable. The Spearman rank correlation coefficient may also be useful as a better indicator of the existence of non-linear relationships between two variables and when the use of a parametric measure is inappropriate.

$$s_{xy} = \frac{\sum_{k=1}^{N}\left(R_k - \bar{R}\right)\left(S_k - \bar{S}\right)}{\sqrt{\sum_{k=1}^{N}\left(R_k - \bar{R}\right)^2}\sqrt{\sum_{k=1}^{N}\left(S_k - \bar{S}\right)^2}} \tag{2-6}$$

Where, $\bar{R}$, $\bar{S}$ are rank arrays of X and Y respectively.

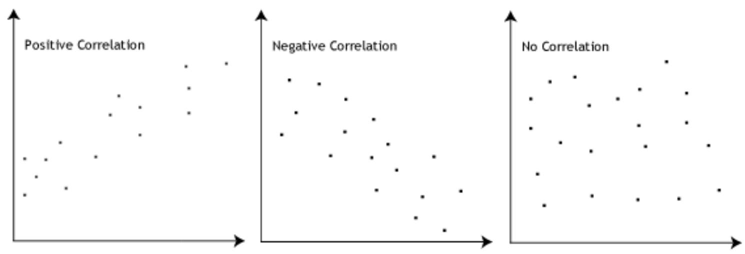



**Figure 2-10:** Example of correlation between two variables, (Adams, 2009)

As shown in Figure 2-10, a correlation coefficient is used to measures the degree of linear correlation between two variables. It ranges between -1 and 1. It is equal to 1, if there is a perfect positive linear relationship between the two variables, while it is -1, if there is perfect negative linear relationship between them. A correlation coefficient of 0 denotes no linear relationship between the variables. In case of a positive relationship between two variables, if one variable increases then the value of the other variable also increases, whereas if there is a negative relationship, then if value of one variable increases, the value of the other variable decreases.

Chapter 3

# Software Verification and Validation

As described in Chapter 2, to answer the main research question an integrated design process is developed within the Dakota framework. Deriving an optimized configuration of a lifting body re-entry vehicle involves the shape and the trajectory optimization simultaneously. Performing the shape and trajectory optimization simultaneously of re-entry vehicle is challenging by itself; and performing this optimization using an open source Dakota framework, which is not yet used for a re-entry vehicle applications is more challenging. Considering these problems involved in this project, to answer the main research question, the problem is breakdown into a small set of tasks as given below:

- Develop the design process within the Dakota framework to perform the feasibility study of a system.
- Extend the design process for re-entry vehicle application.
  1. Perform optimization of axis-symmetric ballistic re-entry vehicle (Raduga configuration).
  2. Perform trajectory optimization of the given lifting body re-entry vehicle (IXV).
  3. Perform the geometry and the trajectory optimization simultaneously of a complex re-entry vehicle similar to IXV to answer the main research question.

The first task includes the development of a design process within the Dakota framework, that can enable one to integrate the different tools and software on a single platform and automate the design process. This design process was described in the previous chapter. Once the design methodology is created within the Dakota framework, it is then extended for the re-entry vehicle applications.

As the task to adapt the design process for the complex system like re-entry vehicle, initially only an axis-symmetric configuration of re-entry vehicle is chosen to establish the methodology for the re-entry vehicle application. The choice of a simple axis-symmetric configured re-entry vehicle (Raduga configuration) allows one to improve the design process and gives a better understanding of requirements for a generalized interface between each tool as well the possibilities to plug and play with the various tools or commercial software that are developed using different programming languages.

Once the design process is verified to derive an optimized configuration of the Raduga capsule, the design process and the optimization techniques are fixed. The design process is then extended for a lifting body vehicle by initially performing the trajectory optimization of a given configuration and its aerodynamic database. The IXV configuration is considered to verify the design process for the trajectory optimization. The main research question, involves shape as well as trajectory optimization of the IXV-like vehicle, thus a new tool is created to perform the shape morphing and the mass estimation of the re-entry vehicle. This tool is introduced in the work flow for the design case. This chapter describes all the tools considered for the thesis and furthermore highlights their unit test verification.

## 3-1 Software and Tools

Once the design process is developed within the Dakota framework as described in Chapter 2. This work flow is furthermore extended for the re-entry vehicle application by implementing the necessary tools and software within the design process. This section describes the tools and software considered for the complete thesis work. As described above, the initial work flow includes optimization of the Raduga capsule. For this optimization, there is a requirement of tools that can generate geometry, compute mass, generate mesh, derive aerodynamic data-set, perform trajectory simulations, compute aero-thermodynamic database and for the thermal analysis.

For the trajectory optimization of the IXV, the aero-thermodynamics database computation is required. The trajectory optimization is performed for a given aerodynamic database of the IXV. Once the design process for the trajectory optimization of lifting body and the optimization of an axis-symmetric configuration is performed, it is then clubbed together to answer the main research question. Additionally, to answer the main research question, it is required to perform the shape and trajectory optimization of a re-entry vehicle. Thus, an additional tool Shape Morphing and Mass Estimation Tool (SMMET), for the shape morphing and mass estimation is required to answer the main research question. The primary tools/software required for the geometry optimization of Raduga configuration, trajectory optimization of IXV and for the design case are given in Table 3-1.

**Table 3-1:** Tools and software included in the workflow for the feasibility study of a re-entry vehicle

| Tools and software | Description or name of tool/ software |
|---|---|
| GGMET | Geometry generation and mass estimation tool for the Raduga like capsule (Matlab) |
| SMMET | Shape morphing and mass estimation tool for IXV like vehicle (Matlab) |
| Commercial tool | |
| Gridgen | Mesh generation software |
| Tools developed by TASI | |
| CoRADS | Aerodynamic database computation (Matlab) |
| FMST | Trajectory simulator (Simulink) |
| Modified CoRADS | Aero-thermodynamics database computation (Matlab) |
| Thermal analysis tool | TPS thermal analysis tool (Fortran) |

The Geometry Generation and Mass Estimation Tool (GGMET) is developed to generate the geometry of a Raduga-like capsule using analytical expressions. Raduga is an axis-symmetric capsule. Additionally, this tool also gives the mass estimation of the derived configuration. SMMET is developed to morph the original mesh of the IXV to derive new configuration of re-entry vehicle. This tool is also responsible for the mass estimation of the derived configuration. Subsection 3-1-1 describes tools that are developed along with their unit test verification.

Furthermore, the above tools and software for aerodynamic database computation, trajectory simulation, aero-thermodynamics database computation and thermal analysis are provided by

TASI. For implementing these tools in the design process an acceptance test is performed for these tools. Furthermore, Gridgen is a commercial software for the mesh generation and thus there is no requirement of the verification of this tool. Subsection 3-1-2 describes the available tools and their acceptance test.

## 3-1-1 Verification of Developed Tools

This subsection describes the tools that are developed to complete the workflow for the multidisciplinary optimization of the re-entry vehicle. For the Raduga configuration optimization, there is a requirement to create a tool that describes the geometry and furthermore computes the payload mass capacity of the vehicle. This section initially describes the tool dedicated for the geometry generation and mass estimation (GGMET) of the Raduga-like configuration and its verification. Furthermore, this section describes the tool created for the shape morphing of the baseline IXV configuration and mass estimation of the lifting body re-entry vehicle.

### Geometry Generation and Mass Estimation Tool (GGMET)

Figure 3-1 gives the top level architecture of the GGMET. The user can define the design parameters as inputs and compute the dimensions of the re-entry vehicle, furthermore the geometric parameters are utilized to compute the volume and mass of each sections.

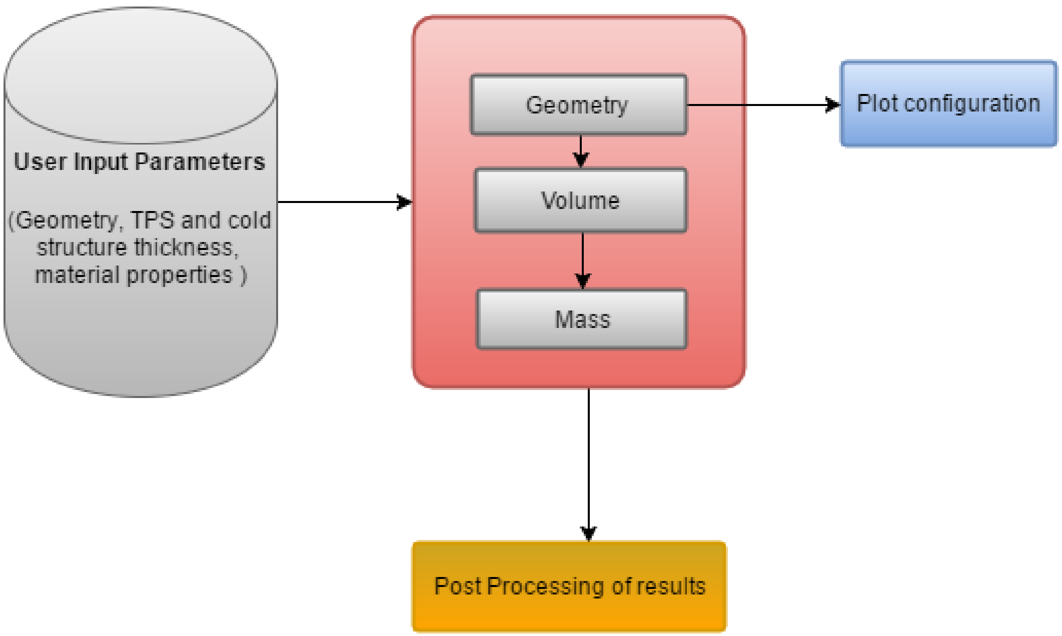



**Figure 3-1:** Top-level architecture of geometry and mass tool

Analytical expressions are derived to define the geometry of an axis-symmetric configuration similar to Raduga capsule. These expressions are defined as a function of the input parameters. Figure 3-2 gives the parametrization of the re-entry vehicle.

Adami et al. (2015) gives the parametrization of bi-conic capsule. Considering this as a reference, the parametrization of the shape is defined by a set of analytical expressions. These expressions are derived by enforcing the continuity and the tangency at the boundary of each conical and spherical segment to obtain a consistent design for variation in the input parameters. The following are the analytical expressions:

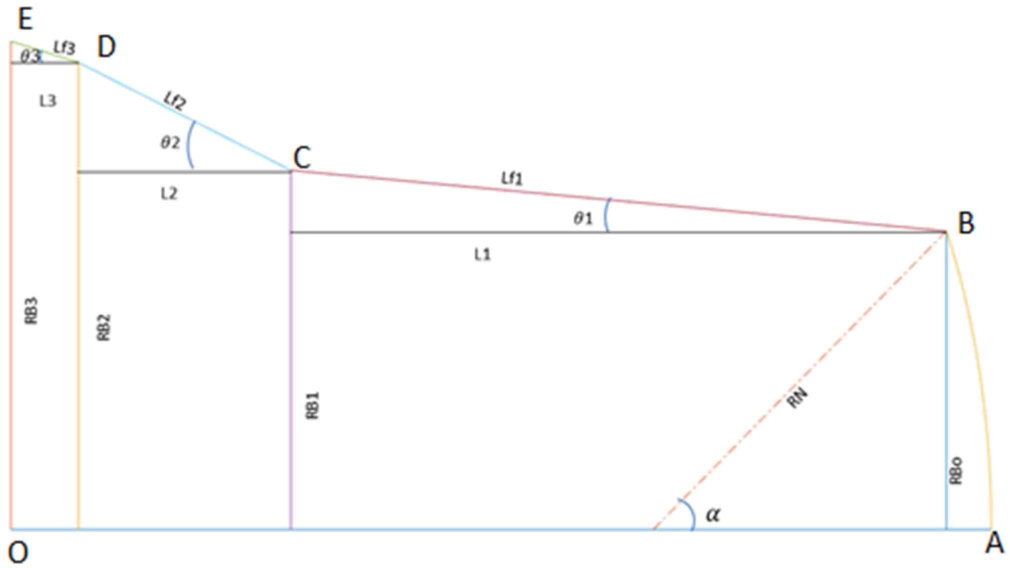



**Figure 3-2:** Parameterisation of the curve generating the solid of revolution

$$R(x)_i = R(x)_{i-1} + L_i \tan\theta_i \tag{3-1}$$

Here, i=[1,2,3] and R(x)$_0$=$RBo$. R(x) gives the base radius of the configuration at the given distance from origin. $L1$, $L2$ and $L3$ are the lengths of all sections of the capsule respectively. Similarly, $RB1$, $RB2$ and $RB3$ are the base radius of all sections of the configuration. At x=$L3$, the base radius is $RB3$, at x=$L3$+$L2$, the base radius is $RB2$ and for x=$L3$+$L2$+$L1$ the base radius is $RB1$. The $\alpha$, $RN$ and $RBo$ are related by Equation 3-2.

$$\sin\alpha = \frac{RBo}{RN} \tag{3-2}$$

During the unit test of the software, it is checked that if the relation $0 < \sin\alpha \leq 1$ is not satisfied, then a dummy value is generated, such that the solution is discarded during the optimization. This is also done to make sure that the workflow do not break if $RN \leq RBo$ during the feasibility studies.

The following equations defines the base radius of the vehicle. The maximum of these base radius will be considered for the outer envelope constraint defined by the selected launcher. The sections of the geometry are nose section, first section, second section and the third section. Here i = [1,2,3] where, the number defines the first section, second section and the third section respectively. The base radius of each section is defined by the following relation:

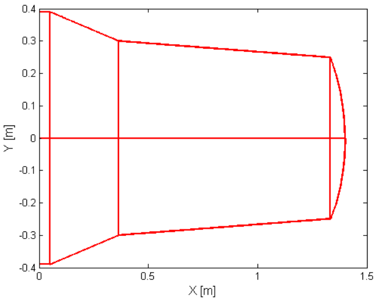
$$RB_i = L_i \tan\theta_i + RB_{i-1} \tag{3-3}$$

When i=1, R$_0$ is the nose base radius (RBo). Furthermore, the slant lengths of the configuration and the total length is defined by Equation 3-4 and Equation 3-5 respectively.

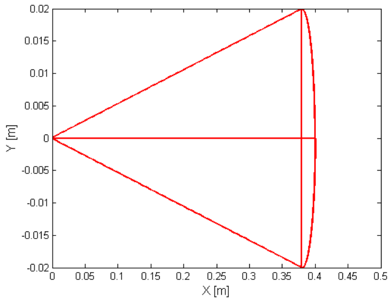
$$Lf_i = \frac{L_i}{\cos\theta_i} \tag{3-4}$$

$$L_{total} = L1 + L2 + L3 + RN(1 - \cos\alpha) \tag{3-5}$$

Here $Lf1$, $Lf2$ and $Lf3$ are the slant lengths of all sections respectively and $\theta1$, $\theta2$, $\theta3$ gives the cone angle of each section. Equation 3-4 gives the relation of the slant lengths of each section and their respective cone angle. $L_{total}$ is the total length of the capsule.

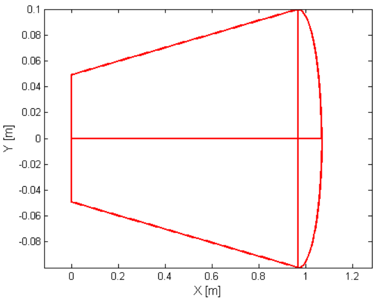



**(a)** Configuration 1

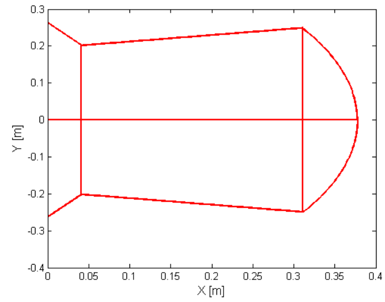



**(b)** Configuration 2

**Figure 3-3:** Axis-symmetric configurations generated from GGMET




**(a)** Configuration 3




**(b)** Configuration 4

**Figure 3-4:** Axis-symmetric configurations generated from GGMET

Table 3-2 gives the input parameters used to compute the dimensions, volume and payload mass of the re-entry vehicle using the above equations. The GGMET developed for the Raduga configuration optimization is a tool that can create different axis-symmetric configurations by the variation in the inputs described in Table 3-2. Figures 3-3 and 3-4 give example of configurations generated from this tool. Starting from a 3 section axis-symmetric capsule to a Apollo-like capsule can be derived from this tool.

Using the inputs given in Table 3-2, the other dimensions of the vehicle are computed. Table 3-3 gives the derived dimensions of the results from GGMET and their comparison with the reference dimensions of the Raduga capsule, as given by Legostaev and Minenko (1994). Once the dimensions of a re-entry vehicle are derived from the above equations, the external volume is computed by using the following analytical expressions. The external volume of the nose section of the vehicle is given by Equation 3-6 and the external volume of the first, second and third section is given by Equation 3-7.

$$Volume_{E,nose} = \frac{4\alpha}{3}RN^3 - \frac{1}{3}\pi RBo^2 RN \cos\alpha \tag{3-6}$$

**Table 3-2:** Inputs for GGMET, Legostaev and Minenko (1994)

| Input parameters | | Units |
|---|---|---|
| Front cone nose radius (RBo) | 0.24 | [m] |
| Nose radius (RN) | 0.50 | [m] |
| Length of section1 (L1) | 0.96 | [m] |
| Length of section2 (L2) | 0.31 | [m] |
| Length of section3 (L3) | 0.05 | [m] |
| Cone angle of section1 ($\theta 1$) | 3.00 | [deg] |
| Cone angle of section2 ($\theta 2$) | 16.00 | [deg] |
| Cone angle of section3 ($\theta 3$) | 0.00 | [deg] |
| TPS thickness of nose section($TPSthk$) | 0.07 | [m] |
| TPS thickness of section1 ($TPSthk1$) | 0.06 | [m] |
| TPS thickness of section2 ($TPSthk2$) | 0.05 | [m] |
| TPS thickness of section3 ($TPSthk3$) | 0.05 | [m] |

**Table 3-3:** Geometry tool outputs

| Parameters | GGMET | Reference | Units |
|---|---|---|---|
| Slant length of section1 (Lf1) | 0.97 | 0.97 | [m] |
| Slant length of section2 (Lf2) | 0.32 | 0.32 | [m] |
| Slant length of section3 (Lf3) | 0.05 | 0.05 | [m] |
| Base radius of section1 (RB1) | 0.30 | 0.30 | [m] |
| Base radius of section2 (RB2) | 0.39 | 0.39 | [m] |
| Base radius of section3 (RB3) | 0.39 | 0.39 | [m] |
| Nose cone angle ($\alpha$) | 29.89 | 29.89 | [deg] |
| Total length of vehicle ($L_{total}$) | 1.40 | 1.40 | [m] |

$$Volume_{E,section_i} = \left(\frac{1}{3}\pi L_i (Lf_i \sin\theta_i)^2\right)^2 + \pi RB_{i-1}^2 L_i \tag{3-7}$$

Here i = [1,2,3] where, the number defines the first section, second section and the third section respectively. Once, the external volume of each section is computed, the total external volume of the capsule is evaluated using the by the following equation.

$$Volume_{E,total} = Volume_{E,nose} + \sum_{i=1}^{3} Volume_{E,section_i} \tag{3-8}$$

**Table 3-4:** Geometry tool outputs: External volume of sections

| External volume | From GGMET | Reference | Units |
|---|---|---|---|
| External volume of nose section | 0.06 | - | [m$^3$] |
| External volume of section1 | 0.19 | - | [m$^3$] |
| External volume of section2 | 0.09 | - | [m$^3$] |
| External volume of section3 | 0.02 | - | [m$^3$] |
| Total external volume | 0.37 | 0.37 | [m$^3$] |

Table 3-4 gives the external volume of each section of the re-entry vehicle computed by GGMET and the total external volume is compared with the reference volume of Raduga capsule. The total external volume is required, while selecting the launcher for the re-entry vehicle or inserting the constraint over the external volume, while performing the optimization to chose a specific launcher. Furthermore, the internal volume is required to compute the maximum payload volume capacity. Analytical expressions are derived to compute the internal volume of the capsule considering the given value of TPS thickness and cold structure and marginal thickness for other reinforcements. To derive these expressions, the cold structure thickness and the reinforcement thickness are together considered as structural thickness.

**Table 3-5:** Geometry tool outputs: Volume of sections

| Volume of TPS | Values | Units |
|---|---|---|
| Volume of TPS nose section | 0.05 | [$m^3$] |
| Volume of TPS section1 | 0.08 | [$m^3$] |
| Volume of TPS section2 | 0.03 | [$m^3$] |
| Volume of TPS section3 | 0.01 | [$m^3$] |
| Total TPS volume | 0.17 | [$m^3$] |

It is assumed to be uniform all over the surface of the capsule, whereas the TPS thickness is considered to be different for all the four sections as well as for the afterbody of the vehicle. From the literature study the TPS thickness of the afterbody is assumed as 20 % of the TPS thickness at the nose. Since the thickness of the TPS, and the structural thickness are know one can compute the volume of the TPS and the structure using the similar relations as mentioned above. Table 3-5 and Table 3-6 gives the volume of the TPS and the cold structure computed using the GGMET respectively. Furthermore, Table 3-7 gives the internal volume of each section evaluated by the GGMET.

**Table 3-6:** Geometry tool outputs: Volume of cold structure

| Volume of cold structure | Values | Units |
|---|---|---|
| Volume of structure nose section | 0.002 | [$m^3$] |
| Volume of structure section1 | 0.009 | [$m^3$] |
| Volume of structure section2 | 0.004 | [$m^3$] |
| Volume of structure section3 | 0.001 | [$m^3$] |
| Total structure volume | 0.018 | [$m^3$] |

**Table 3-7:** Geometry tool outputs: Internal volume of sections

| Internal volume | Values | Units |
|---|---|---|
| Internal volume of nose section | 0.004 | [$m^3$] |
| Internal volume of section1 | 0.100 | [$m^3$] |
| Internal volume of section2 | 0.060 | [$m^3$] |
| Internal volume of section3 | 0.017 | [$m^3$] |
| Total internal volume | 0.182 | [$m^3$] |

Legostaev and Minenko (1994) describes the total payload volume of the Raduga configuration as 0.12 $m^3$. The vehicle is designed for the cargo transfer and thus 65 % of this maximum available internal volume is assumed as the maximum payload volume capacity of the vehicle. Table 3-8 gives the total payload volume capacity of the Raduga configuration, where it is compared with the reference data of Raduga capsule, Legostaev and Minenko (1994).

Once the dimensions and the volume of the configuration is verified, they are further used to compute the payload mass capacity of the vehicle. The mass of TPS and the cold structure of the re-entry vehicle is determined from the the volume of TPS and cold structure. The density

**Table 3-8:** Geometry tool outputs: Volume of sections

| | GGMET | Reference | Units |
|---|---|---|---|
| Payload volume | 0.12 | 0.12 | [$m^3$] |

of the material used for the TPS and the structure is considered as given input, thus the mass of TPS and structure is computed by the product of the density of the corresponding material and the volume. For the feasibility study of the Raduga configuration, the total mass of the re-entry vehicle is considered as 350 kg, as described by Legostaev and Minenko (1994).

**Table 3-9:** Mass distribution, Larson and Wertz (1992)

| Subsystem | Mass estimate [%] |
|---|---|
| Structure | 40 |
| Power system | 10 |
| Flight instrumentation | 8 |
| GNC | 16 |
| TPS | 8 |
| Recovery system | 14 |
| Launcher I/F | 2 |

The dry mass and payload mass can be computed from the thumb rules as described by Larson and Wertz (1992). As the total mass of the vehicle is considered as constant, the lower is the dry mass of the vehicle, the higher is the payload capacity. The dry mass includes mass of the subsystem, which is assumed to remain constant irrespective of the variation in the configuration. The payload mass is derived from the following equation.

$$Mass_{payload} = Total\,Mass - (Mass_{constant} + Mass_{TPS} + Mass_{structure}) \tag{3-9}$$

where, $Mass_{constant}$ is the total mass of the subsystem which remains constant irrespective of the variation the configuration during optimization. The nose section is occupied by the TPS and cold structure and thus, the avionics part is distributed in the first section. The power system, flight instrument, GNC, launcher IF is uniformly distributed in the first Section. The recovery system is uniformly distributed in the third section. The mass of payload is uniformly distributed in the first and the second section.

$$Payload\,Mass_{section\,1} = Mass_{payload}\left(\frac{Area\,of\,section\,1}{Area\,of\,section\,1 + Area\,of\,section\,2}\right) \tag{3-10}$$

$$PayloadMass_{Section\,2} = Mass_{payload}\left(\frac{Area\,of\,section\,2}{Area\,of\,section\,1 + Area\,of\,section\,2}\right) \tag{3-11}$$

Furthermore, the total mass of each section is computed. Since the nose section is occupied by the TPS and the cold structure, the total mass of nose section is summation of the TPS and structure mass of the nose section. Furthermore, the mass of the section 1 is given by the summation of the TPS and structure mass of section 1 along with the power system, flight instrument, GNC, launcher IF system mass. Additionally the section also includes a part of payload mass distributed in section 1 as given by Equation 3-10. Furthermore, the total mass of the section includes the mass of the TPS and structure of second section along with the part of the payload mass distributed in section 2 as given by Equation 3-11. Finally the mass of the third section includes the mass of the TPS and structure of this section along with recovery system mass. Table 3-10 gives the

**Table 3-10:** Geometry tool outputs: Mass budget

| Mass | Value | Units |
|---|---|---|
| Mass of Payload | 150 | [kg] |
| Dry Mass | 200 | [kg] |
| Subsystem mass | 102 | [kg] |
| Mass of TPS | 48 | [kg] |
| Mass of structure | 50 | [kg] |
| Total mass | 350 | [kg] |

mass distribution of the Raduga configuration derived by GGMET. Furthermore, the location of the center of mass of each section is computed by considering the uniform mass distribution and thus, the center of mass is located at the geometric center of each section. The Center of Mass (COM) is computed by the following relationship:

$$COM_x = \frac{Mass_{nose}\, COM_{nose} + \sum_{i=1}^{3}(Mass_{section(i)}\, COM_{section(i)})}{Mass_{nose} + \sum_{i=1}^{3} Mass_{section(i)}} \tag{3-12}$$

Here, Mass$_{nose}$ and COM$_{nose}$ is the total mass and location of COM of the nose section, respectively. Mass$_{section(i)}$ and COM$_{section(i)}$ is the total mass and location of COM of each section (first, second, third) respectively.

**Table 3-11:** Geometry tool outputs: Center of mass location

| Section | Mass [kg] | COM [m] | Reference [m] |
|---|---|---|---|
| Nose section | 21 | 0.02 | - |
| First section | 227 | 0.55 | - |
| Second section | 62 | 1.19 | - |
| Third section | 40 | 1.37 | - |
| Configuration | 350 | 0.72 | 0.72 |

### Shape Morphing and Mass Estimation Tool (SMMET)

To answer the main research question, it is required to design a re-entry vehicle with the payload carrying capacity similar to IXV. Thus, the IXV is considered as a reference geometry and the mesh of this configuration is morphed to derive new design configurations. Figure 3-5 gives the software architecture of the SMMET.

This software is developed using Matlab. The user need to provide the initial mesh of the IXV and the scaling factors to morph the mesh. Figure 3-6 and Figure 3-7 describes the sections of the configuration. The configuration is divided into 3 sections namely; nose section, main body section and the flap section. Each section is scaled along the length with 3 different scaling factors namely; $X_{nose}$, $X_{body}$ and $X_{flap}$. Furthermore, to morph the design along the width, $Y_{start}$, $Y_{end}$ are the scaling factors introduced as shown in Figure 3-6. Similarly, to morph the design along the height, $Z_{start}$, $Z_{end}$ are the scaling factors introduced as shown in Figure 3-7. Using these scaling factors, the initial mesh of IXV is morphed along the length, width and height. Thus, the user provides the initial mesh of IXV and these scaling factors as input to the tool.

Once the inputs are given to the tool, as described in Figure 3-5 the first block is responsible to define the elements of the mesh. The given mesh of IXV is a structured and raw triangular format. This block defines the co-ordinates of each vertex and the centroid of each triangular element of

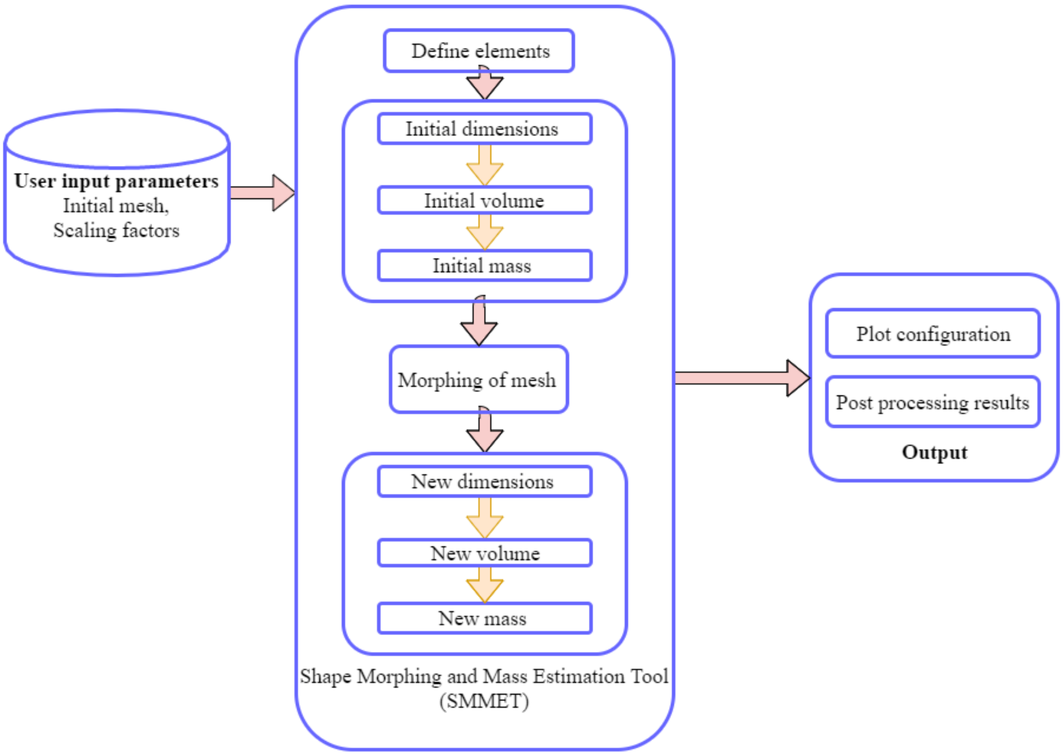



**Figure 3-5:** Software architecture of SMMET

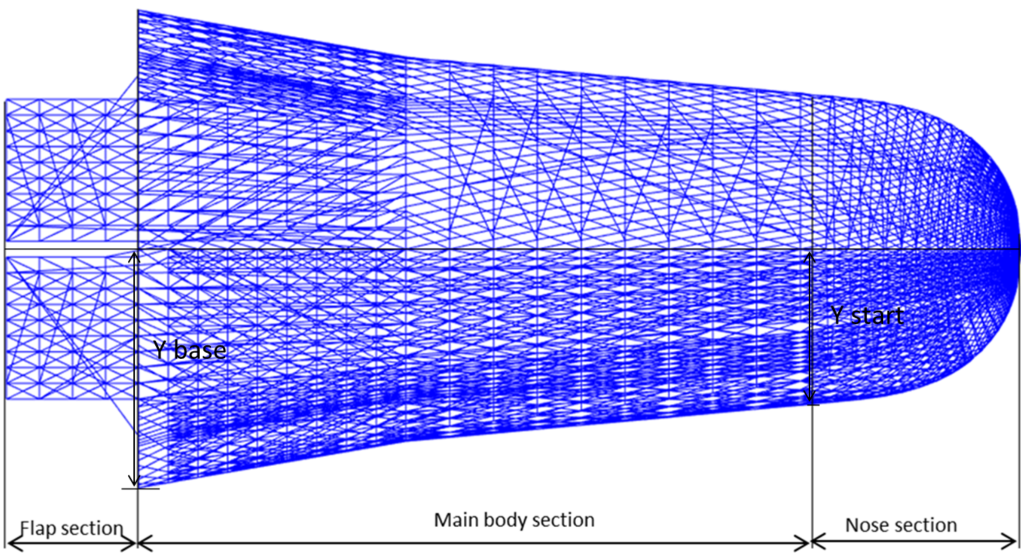



**Figure 3-6:** Sections along the length of configuration

the mesh. Furthermore, the surface area of each triangular element is also computed within this block, which is further used for the mass estimation of the configuration. The elements defined in this block are used to divide the configuration in different section. By morphing the mesh, the number of elements in the respective section and their corresponding index remain same. Thus allows the user to define as many sections as required to compute the necessary dimensions and thereby the volume. The nose section, main body section, flap section and payload section are the

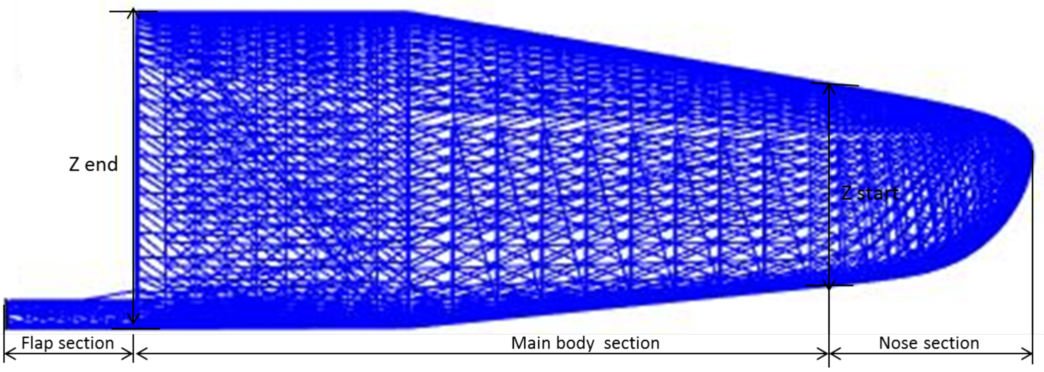



**Figure 3-7:** Sections along the length of configuration

required sections.

Here, the origin is considered at the tip of the nose, and the elements which are between the $x$=0 to $x$=1.1 are considered as nose elements, the elements which lie between the $x$=1.1 to $x$=4.4 are considered as main body elements and the elements, which are after $x$=4.4 are considered as flap elements. Furthermore, the payload section elements are the elements which are between $x$= 1.5 to $x$=3.0.

The second block as described in Figure 3-5 is responsible to compute the dimensions of the initial mesh. Once the elements within each sections are known, the dimensions of each section is computed by the difference between the co-ordinates of the centroid of the elements. The difference between the maximum and minimum value of the centroid of the triangular mesh elements along the length, width, height gives the total length, total width and the total height of the vehicle respectively. Similarly, the dimensions of the payload bay is computed by the difference between the maximum and minimum value of the centroid of the triangular mesh elements within the payload section.

The third block as described in Figure 3-5, is responsible to compute the payload volume of the initial mesh. The payload section is a trapezoidal section. Since the elements of the payload section are know, one can similarly compute the necessary dimension of the payload section to compute the volume of the trapezoid. Furthermore, considering the space required for the TPS, cold structure as well as for the other margins 50% of this payload section volume is assumed as payload volume capacity of the configuration.

The fourth block is responsible to compute the mass of the configuration as described in Figure 3-5. The mass budget of the reference configuration IXV is described by Chiarelli (2014). Using this mass budget, the mass of the configuration is found. It is also considered that the tool is developed for a design case, which is payload carrying vehicle, whereas the mass budget is of IXV is without payload. The total mass of the design case is considered as 2150 kg, whereas the IXV total mass is 1841.3 kg. It is considered that the total mass of IXV was higher to balance the Center of Gravity (COG) and thus the ballast mass can be removed from the mass budget along with some unwanted mass. The total mass of the design case is considered as constant and the mass budget is derived from the IXV configuration mass budget. Here, it is assumed that the fixed mass does not change, with the morphing of the vehicle similar to IXV configuration. The fixed mass for the original IXV is 726.76 kg and for the design case 142 kg mass is removed from this fixed mass. This gives the payload mass capacity of 450 kg for the original design case with scaling factor of 1.

The fifth block includes morphing of the configuration. Prior to this block, the elements are defined for each section. The elements within the section remains constant, even after the morphing of the vehicle. To morph the vehicle the scaling factors described previously are used. The nose section, main body section and the flap section are scaled separately based on the input scaling factors. To morph the configuration along the width and the height scaling factors are introduced for the main body section, where $Y_{start}$, $Z_{start}$ and $Y_{end}$, $Z_{end}$ are the scaling factor for the elements at the intersection of the nose-main body section and main body-flap section. The morphing of the configuration is carried out using the linear interpolation between the co-ordinates of these elements.

Once the original mesh is morphed, the new configuration is saved in the working directory. As the elements remains same within each section, even after morphing the mesh, the dimensions and the volume of the new configuration is computed using the similar approach. These new dimensions and the payload volume is computed in the sixth and the seventh block of the SMMET. Furthermore, in eighth block, the mass of the new configuration is computed. It is assumed that the same material as IXV are used for the design case and thus mass is dependent only on the ratio of the surface area of new and the original configuration. Thus, this block computes the ratio between the surface area of the required sections of the original configuration (scaling factor =1) and new configuration. The mass TPS and the structure is computed by the product of this ratio of surface area with the original mass budget. Thus, one have the fixed mass, TPS and structure mass of the new configuration. The total mass is considered to be constant and thus the payload mass can be computed by subtracting fixed mass, TPS mass and the structure mass from the total mass. Figure 3-8, gives the example of the morphed designs (red), which are compared with the original configuration (blue). Furthermore, Section 6-3-1 describes additional morphed designs derived from the variation of these scaling factors.

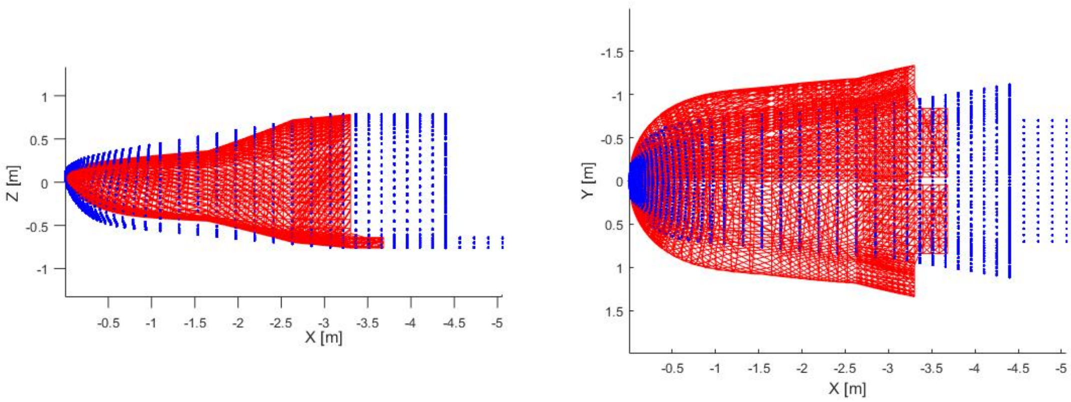



**Figure 3-8:** Morphing of the IXV mesh

### Verification of SMMET for IXV

To verify the SMMET, it is important to check if the geometry for a scaling factor of 1 produces the same configuration as IXV and to derive so, what can be the correction factors to match with reference dimensions of IXV.

Figure 3-9 shows the comparison of the morphed design (red) with the reference IXV configuration (blue). The scaling factors are considered to be 1 to morph the original mesh. From the figure, it is seen that both the designs superimpose each other.

Furthermore, the dimensions of the given configuration computed by the SMMET are compared with the reference data to find the accuracy of the tool and implement the correction factors. Furthermore, the morphed design derived with scaling factor of 1 are compared with the reference

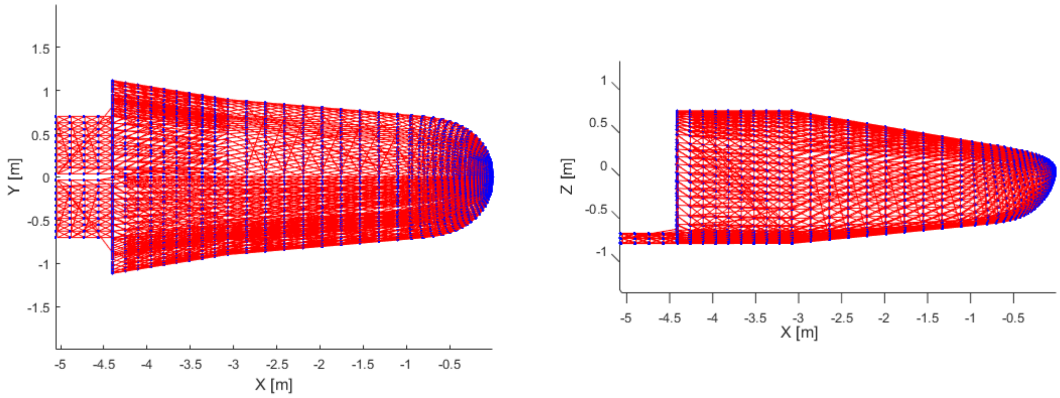



**Figure 3-9:** Comparison of the morphed design with scaling factor of 1 with the reference design

data to verify if the design produced after the morphing is same to as the given input and implement the additional correction factors to match the results.

Table 3-12 gives the dimensions of the given mesh computed by SMMET and the correction factor to match the results with the reference data available of IXV, as shown in Figure 3-10.

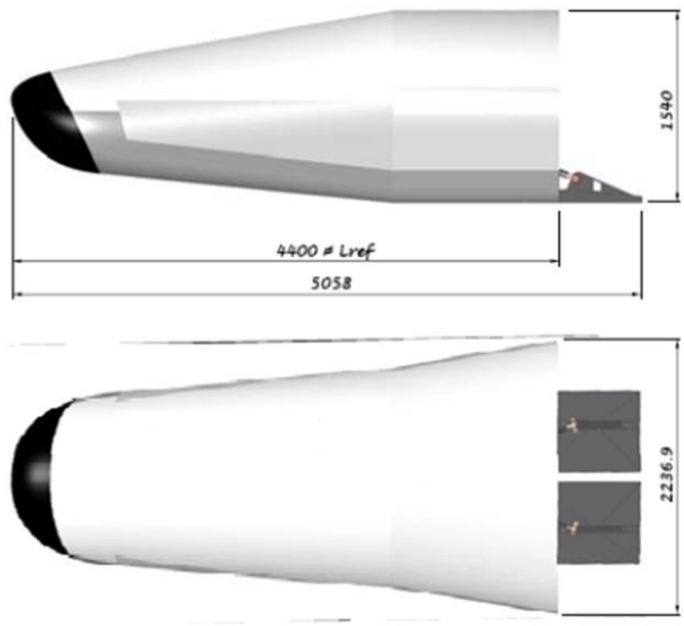



**Figure 3-10:** Intermediate Experimental Vehicle (IXV) shape and dimension, (Rodrigo et al., 2016)

**Table 3-12:** Geometry tool outputs: Dimensions

| Dimensions | IXV (Reference) | Output from SMMET (Before morphing) | Correction factor |
|---|---|---|---|
| Total length | 5.058 [m] | 5.057 [m] | 1.0001 |
| Maximum base width | 2.236 [m] | 2.216 [m] | 1.0093 |
| Maximum base height | 1.540 [m] | 1.539 [m] | 1.0006 |

Table 3-13 gives the dimensions of the new configuration generated after morphing computed by SMMET and the correction factor to match the results with the reference data available of IXV. The correction factors are within 1% and thus it is verified that SMMET computes accurate dimensions of the original configuration as well as of the new configuration generated after morphing by adding this small correction factors.

**Table 3-13:** Geometry tool outputs: Dimensions

| Dimensions | IXV (Reference) | Output from SMMET (After morphing) | Correction factor |
|---|---|---|---|
| Total length | 5.058 [m] | 5.036 [m] | 1.0042 |
| Maximum base width | 2.236 [m] | 2.276 [m] | 1.0093 |
| Maximum base height | 1.540 [m] | 1.539 [m] | 1.0006 |

Table 3-14 gives the payload volume and the dimensions of the payload section. Furthermore, the payload bay is the addition made to the reference IXV configuration, thus the volume of the payload bay is compared with the Space-rider configuration as described by Rufolo (2016). The payload section is considered to start from the 35% of the total length to the 55% of the total length, with respect to the nose. Additionally, the complete section volume cannot be used for the payload as most of the space will be consumed by the coating of thermal projection and cold structure. Thus, 50% of this section volume is considered as the payload bay volume. The length of the payload bay is equal to the length of this section whereas, the width and the height is reduced by 50% to derive the width and the height of the section.

**Table 3-14:** Geometry tool outputs: Payload bay dimensions

| Dimensions | Spacerider (Reference) | Output from SMMET | Units |
|---|---|---|---|
| Payload volume | 0.56 | 0.56 | [$m^3$] |
| Payload bay length | 0.73 | 0.73 | [m] |
| Payload bay width 1 | - | 0.80 | [m] |
| Payload bay width 2 | - | 0.87 | [m] |
| Payload bay height 1 | - | 0.61 | [m] |
| Payload bay height 2 | - | 0.73 | [m] |

The mass budget of the IXV configuration is available, which is considered as reference to compute the mass of the design case. It is assumed that the same material as used for IXV is utilized for the design configuration. To compute the mass of the different sections, the ratio of the new surface area with respect to the reference design is multiplied with the reference mass. The surface area of each triangle of the mesh element and the integral of these surface area for the respective elements in the section gives the surface area of the section. From these mass distribution the payload mass is computed for the given total mass of the vehicle. To verify the SMMET the surface area compute for each section should be equal before the scaling and after scaling with the factor of 1. To match the results a correction factor is implemented. Table 3-15 gives the surface area of the configuration computed by SMMET before morphing and after morphing along with the correction factors implemented.

From the correction factors described in Table 3-15, it is seen that the correction factors are very low and thus tool can be accepted by implementing these correction factors. Table 3-16 gives the mass budget of the derived configuration using SMMET and the comparison with the reference values as described by Chiarelli (2014).

**Table 3-15:** Geometry tool outputs: surface area

| Surface area | Obtained from SMMET (Before scaling) | Obtained from SMMET (After scaling) | Units | Correction factor |
|---|---|---|---|---|
| Total surface area | 26.13 | 26.07 | [$m^2$] | 1.0020 |
| Nose surface area | 3.89 | 3.91 | [$m^2$] | 0.9940 |
| Windward surface area | 7.13 | 7.14 | [$m^2$] | 0.9999 |
| Ablative surface area | 14.21 | 14.20 | [$m^2$] | 1.0008 |
| Flaps surface area | 2.19 | 2.12 | [$m^2$] | 1.0330 |

**Table 3-16:** Geometry tool outputs: mass budget

| Mass budget | Reference values | SMMET | Units |
|---|---|---|---|
| Mass of TPS | 449 | 449 | [kg] |
| Mass of structure | 665 | 665 | [kg] |
| Fixed mass | 585 | 585 | [kg] |
| Payload mass | 450 | 450 | [kg] |

## 3-1-2 Verification of Available Tools

The aerodynamic database computation tool Code for Rapid Aerodynamic Database Synthesis (CoRADS), trajectory simulator (Flight Mechanics Simulation Tool (FMST)), the modified CoRADS for the aero-thermodynamic database computation and the thermal analysis tool are provided by TASI and thus an acceptance verification test is performed to implement these tools in the design process developed within the Dakota framework.

#### Verification of Thermal Analysis Tool

The thermal analysis tool is required in the workflow for the TPS sizing of the Raduga-like capsule during the optimization. Van Eekelen et al. (2013) describe Theoretical Ablative Composite for Open Testing (TACOT) an theoretical material used for testing the thermal analysis tools. It is a low-density carbon/phenolic ablative composite. Bianco et al. (2015) describe this thermal analysis tool. This 2-D fully implicit numerical simulation tool, that is capable of evaluating the behavior of an ablative charring thermal protection system during the atmospheric entry. The heat flux inside the porous material and the decomposition of the latter, pyrolysis gas density, pressure and speed distribution and surface recession can be modeled using this tool. Using a time-implicit scheme the fully coupled governing equations are integrated. To simulate the recession phenomenon and compute the recession rate of different ablative models, the grid can be contracted based on the available data and problem requirements.

For the workflow, TACOT material properties are used as TPS and one-dimension analyse is performed using this tool, where the TPS is coupled with the sandwich cold structure with internal adiabatic conditions. The analyse result is henceforth the minimum TPS thickness, as function of the incoming heat flux, needed to meet the cold structure temperature requirement of $\leq$ 480 K. For the verification of the tool, two analysis cases were performed, the first one considered the 480 K requirement to met during the re-entry phase only (440 seconds) and the second one considers to not exceed the 480 K upto 1000 seconds from the Entry Interface Point (EIP). This provided information on the heat transfer occurring within the material even after the convective heat phase is ended.

This provides information on the heat transfer occurring within the material even after the convective heat phase is ended. It is worthy to notice that the transient last strongly depends on the TPS thickness. Furthermore, the analyzed material TACOT thickness seems to have a small dependance on heat fluxes in correspondence of the upper limit considered; in other words, if higher heat fluxes are considered, the thickness would not increase dramatically, but for the lower-flux regions one cannot obtain dramatically lower thickness either. This would suggest that a future optimization in terms of TPS materials (i.e. selecting a different ablative material for the less heated regions, having optimum performances in lower-flux regime) could be useful for mass reduction purposes.

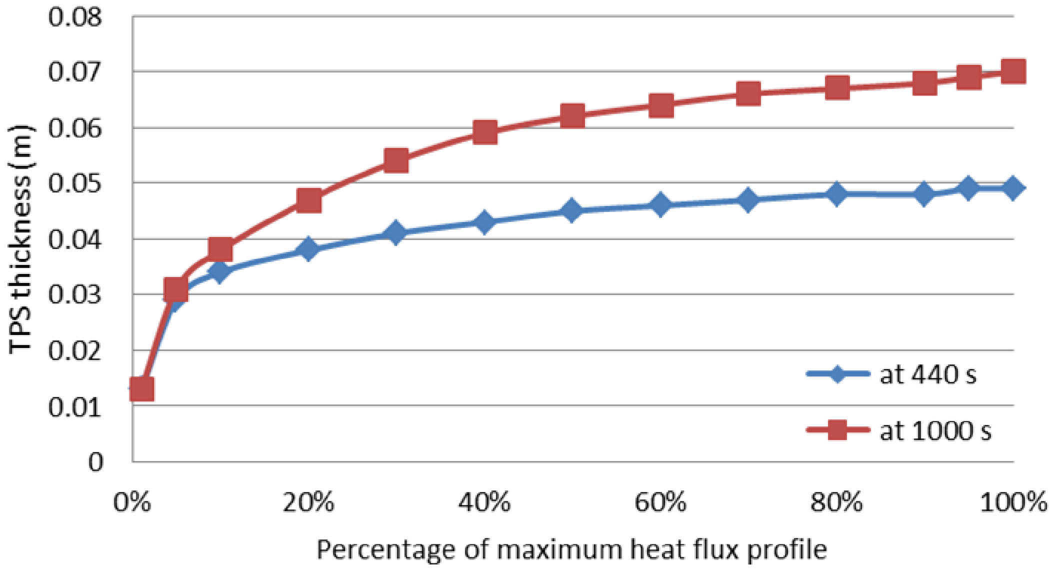



**Figure 3-11:** TPS thickness for given heat flux

**Table 3-17:** Output from thermal analysis tool

| Heat flux | TPS thickness | Temperature |
|---|---|---|
| 3.780 [MW/m$^2$] | 0.07 [m] | 478.67 [K] |
| 1.036 [MW/m$^2$] | 0.06 [m] | 477.08 [K] |
| 0.227 [MW/m$^2$] | 0.05 [m] | 478.56 [K] |
| 0.184 [MW/m$^2$] | 0.05 [m] | 478.56 [K] |

Table 3-17 gives the output from the thermal analysis tool, where for the given heatflux and the TPS thickness the maximum temperature at the cold structure is computed. 3.780 MW/m$^2$ is the maximum heat flux at the stagnation point, thus by comparing with heat flux and the TPS thickness as shown in Figure 3-11, it can be seen that for the 100% heat flux the corresponding TPS thickness is 7 cm as well as within the constraint of 480 K.

#### Verification of CoRADS

#### Newtonian Method

Fluid dynamics theory by Issac Newton derived a law that "force on an inclined plane in a moving fluid varies as the square of the sine of the deflection angle" (Anderson, 2000).

$$C_p = 2\sin^2\theta \tag{3-13}$$

The above equation gives the relation for the Newtonian method, where $C_p$ is the coefficient of pressure. $\theta$ is the local deflection angle, that is the angle between the tangent to the surface and freestream velocity. This method assumes that, upon hitting a surface, the flow loses its component normal to the surface while retaining all of its tangential motion. This theory does not explicitly depend on Mach number as it assumes that $M_\infty$ is high enough to enter the calculations and thus the equation becomes independent of M number. Thus, Modified Newtonian method is introduced, which takes Mach number into account. A modification of this method that is typically used involves an additional physical consideration of supersonic flow, namely the loss of total pressure over a shock wave.

$$C_p = C_{p,max} \sin^2 \theta \tag{3-14}$$

Equation 3-14 gives the relation for the modified Newtonian method, where $C_{p,max}$ is the maximum value of the pressure coefficient, evaluated at the stagnation point behind the normal shock wave. Anderson (2000) gives the modified Newtonian method interms of M and $\gamma$ as given in Equation 3-15.

$$C_{p,max} = \frac{2}{\gamma M_\infty^2} \left\{ \left[ \frac{(\gamma+1)^2 M_\infty^2}{4\gamma M_\infty^2 - 2(\gamma-1)} \right]^{\frac{\gamma}{\gamma-1}} \left[ \frac{1-\gamma+2\gamma M_\infty^2}{\gamma+1} \right] - 1 \right\} \tag{3-15}$$

From this equation, if $\gamma$=1, then $C_{p,max}$ is 2. This derives the Equation 3-13. CoRADS is a tool developed by TASI that computes the aerodynamic database of a given configuration. Sudars, Martin (2016a) describes this tool, where the aerodynamic database is computed by using the modified Newtonian method.

To compute the accuracy of CoRADS the results are compared with the commercial software called SPARTA. SPARTA Development Core Team (2014) describes that SPARTA is a parallel DSMC code for performing simulation of low-density gases in 2d or 3d. The accuracy of the tool is computed by comparing the drag coefficient of IXV at angle-of-attack=45 deg, derived from the SPARTA and CoRADS for continuum regime and for free molecular regime respectively.

Table 3-18 gives the inputs considered for the simulations for continuum regime and Table 3-19 gives the corresponding output derived from SPARTA and CoRADS, respectively. Figure 3-12 gives the pressure and velocity field computed by CoRADS and SPARTA respectively.

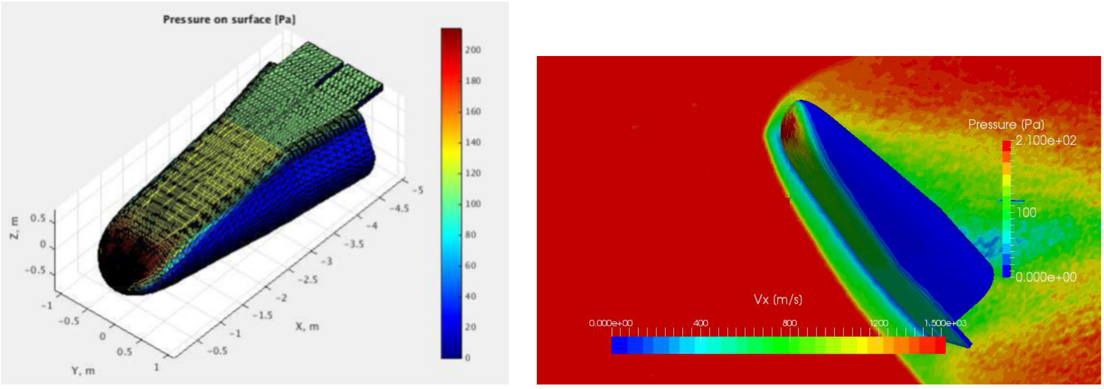
**Table 3-18:** Inputs for CoRADS and SPARTA (Continuum regime)

| | | |
|---|---|---|
| Velocity | 1500 | [m/s] |
| Density | 1.05E-04 | [kg/m$^3$] |

**Table 3-19:** Outputs from CoRADS and SPARTA (Continuum regime)

| | SPARTA | CoRADS |
|---|---|---|
| Cd | 5.920 | 5.952 |
| Stagnation Pressure | 271 [Pa] | 214 [Pa] |
| Computation time | 2.5 hours | 3 sec |
| Iteration | 15000 | - |

From the simulations performed for the continuum filed, it is clear that simulation time required for CoRADS is very less than compared to SPARTA. The difference between the drag co-efficient computed by CoRADS and SPARTA is within 1%. Table 3-20 gives the inputs considered for the simulations for Free Molecular Flow (FMF) regime and Table 3-21 gives the corresponding output




**(a)** Pressure field by CoRADS **(b)** Pressure and velocity field by SPARTA

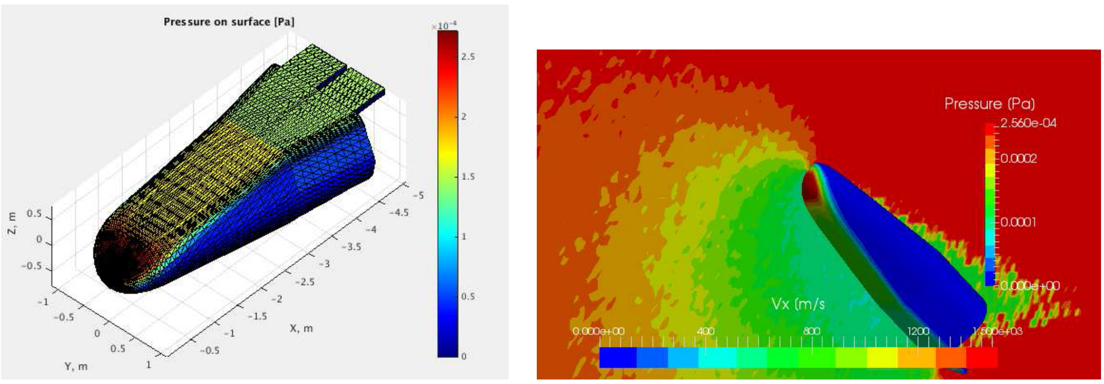
**Figure 3-12:** Pressure and velocity field (Continuum regime), (CoRADS Development team, 2016)

**Table 3-20:** Inputs for CoRADS and SPARTA (FMF regime)

| | | |
|---|---|---|
| Velocity | 1500 | [m/s] |
| Density | 1.05E-10 | [kg/m$^3$] |

**Table 3-21:** Outputs from CoRADS and SPARTA (FMF regime)

| | SPARTA | CoRADS |
|---|---|---|
| Cd | 14.15 | 15.24 |
| Stagnation pressure | 2.91E-04 [Pa] | 2.72E-04 [Pa] |
| Computation time | 2.7 days | 3 sec |
| Iteration | 2500 | - |




**(a)** Pressure field by CoRADS **(b)** Pressure and velocity field by SPARTA

**Figure 3-13:** Pressure and velocity field (FMF regime), (CoRADS Development team, 2016)

derived from SPARTA and CoRADS, respectively. Figure 3-13 gives the pressure and velocity field computed for FMF by CORADS and SPARTA respectively.

From the results for the FMF regime, the computation time for SPARTA is 2.7 days which is very large as compared to 3 second required for CoRADS. Furthermore, the drag co-efficient computed

**Table 3-22:** Inputs for CoRADS

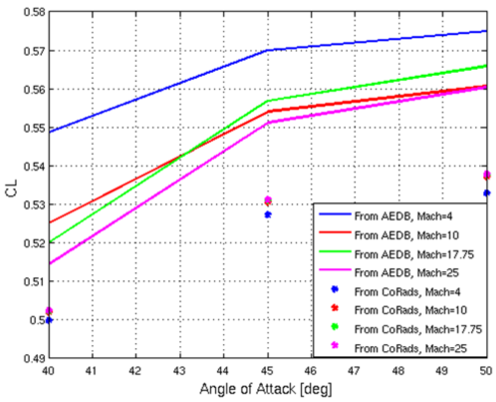
| | | |
|---|---|---|
| Surface area | 7.26 | [$m^2$] |
| Total mass | 1957 | [kg] |
| Radius of nose | 1.05 | [m] |
| Reference length | 5.05 | [m] |
| Altitude | 90000 | [m] |

by CoRADS is 7.1% larger than SPARTA and this value can be considered as the accuracy of the tool with respect to SPARTA for FMF.

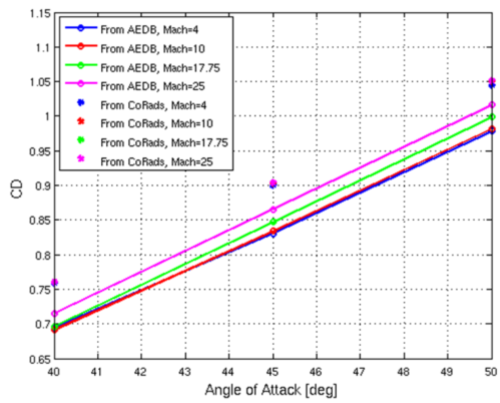
To implement the tool within the work flow, it is necessary to perform the acceptance test and verify if the tool gives the desired output. For the verification of the tool, the aerodynamic data base computed by CoRADS are compared with the reference aerodynamic data base of the IXV as described by Santilli and Sudars (2014). Table 3-22 gives the inputs for the CoRADS tool. Along with these inputs, the mesh of the IXV configuration is provided to the CoRADS as input. The aerodynamic co-efficient are computed about the origin (0,0,0). Using these inputs the aerodynamic data base is generated for the IXV configuration.

**Table 3-23:** Output from CoRADS

| Output from CoRADS | AOA = 40 [deg] | | AOA =45 [deg] | | AOA =50 [deg] | |
|---|---|---|---|---|---|---|
| Mach | CD | CL | CD | CL | CD | CL |
| 4 | 0.759 | 0.499 | 0.900 | 0.527 | 1.049 | 0.533 |
| 10 | 0.760 | 0.520 | 0.903 | 0.531 | 1.050 | 0.537 |
| 17.75 | 0.760 | 0.502 | 0.903 | 0.531 | 1.051 | 0.538 |
| 25 | 0.760 | 0.502 | 0.903 | 0.531 | 1.051 | 0.538 |




**(a)** Lift co-efficient




**(b)** Drag co-efficient

**Figure 3-14:** Comparison of CL and CD computed by CoRADS and the reference aerodynamic database of IXV

Table 3-23 gives the aerodynamic data base generated by CoRADS. Figure 3-14a and Figure 3-14b gives comparison of the lift and drag co-efficient computed by CoRADS and its comparison with the reference aerodynamic database. These results are compared with each other to compute the accuracy of the CoRADS tool for the thesis project. As the CoRADS used modified Newtonian method, the difference between the reference aerodynamic data base is less for higher Mach number than compared to the lower Mach number. It is also observed that the for the given angle-of-attack the value of the drag and the lift co-efficient do not vary a lot with the change in the Mach number. For the higher Mach number the difference between the aerodynamic database computed by CoRADS and the reference data is within 5% deviation. Furthermore for lower Mach the aerodynamic database computed by CoRADS and the reference are within 5-10% deviation.

#### Verification of Flight Mechanics Simulation Tool

This trajectory simulator is developed by TASI. Sudars, Martin (2016b) describes the FMST as a 3/6 Degree of Freedom (DOF)trajectory and attitude dynamics propagation code. It features variety of aerodynamic model formats, environment models, GNC, Monte Carlo and parametric analysis capability, limited trajectory optimization capabilities, data post-processing etc. Sudars, Martin (2016b) describes that, the FMST has been used in some of well-known re-entry vehicle or de-orbit projects such as EXPERT phases C and D, IXV phase C, D and E results crosscheck, safety analysis, ERC 3/6 DOF re-entry analysis Mars sample return, Mars Moon sample return, Phootprint, Phobos sample return, De-orbit decay and re-entry analysis of the satellite assembly, NPSAFE, IRENA, STEPS, Inflatable Aerodynamic Decelerator (IAD).

For this thesis, 3 degrees of freedom entry trajectories are simulated using FMST. Additionally, a standard US76 atmospheric model and WGS84 gravity field is considered. The lateral guidance is based on the dynamic heading-error corridor for bank angle ($\sigma$) maneuvering of the lifting body as described by Liang et al. (2013). The target latitude and longitude co-ordinates are provided by the user, where the heading error determines the required bank angle maneuvering to reach the target landing location.

**Table 3-24:** Inputs for trajectory simulator

| Inputs | | |
|---|---|---|
| Surface area | 7.25 | [m$^2$] |
| Total mass | 1957 | [kg] |
| Radius of nose (RN) | 1.05 | [m] |
| Predicted landing latitude ($\delta_T$) | 3.3 | [deg] |
| Predicted landing longitude ($\tau_T$) | 236.90 | [deg] |
| Entry altitude ($h_E$) | 120.00 | [km] |
| Entry latitude ($\delta_E$) | -4.49 | [deg] |
| Entry longitude($\tau_T$) | 173.20 | [deg] |
| Entry velocity ($V_E$) | 7434.87 | [m/s] |
| Entry heading angle ($\chi_E$) | 86.71 | [deg] |
| Entry flight-path angle ($\gamma_E$) | -1.21 | [deg] |
| Entry angle-of-attack ($\alpha_E$) | 45 | [deg] |

As the trajectory simulator will be used in a tool-chain, where the aerodynamic database will be computed by CoRADS, it is important to verify, if the trajectory results derived from the simulator for the given reference aerodynamic database of IXV match with the nominal trajectory of IXV as well as the trajectory simulator produce the similar results for the aerodynamic database generated by CoRADS. Additionally, for CoRADS only few combinations of M and $\alpha$ are considered to compute the aerodynamic database, whereas the reference aerodynamic database available is a complete data set for the different combinations of M and $\alpha$ (17 $\alpha$ and 15 M numbers). Thus, it is important to verify, if the aerodynamic data produced by CoRADS and reference aerodynamic database corresponding to few selected $\alpha$ and M number (3 $\alpha$ and 4 M numbers) gives similar trajectory.

Table 3-24 gives the input parameters considered for the trajectory simulator. These input parameters are considered same as IXV, to compare the derived trajectories with the nominal trajectory of IXV as described by Rodrigo et al. (2016).

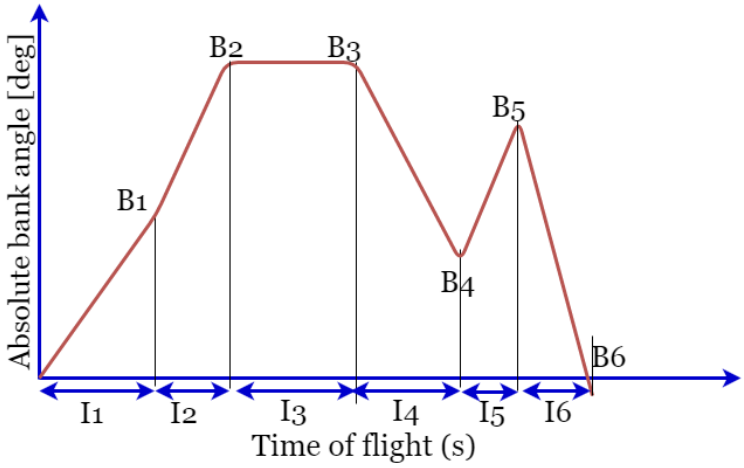



**Figure 3-15:** Bank angle and time interval

Figure 3-15 describes the bank angle and time interval considered as input. Here the B1 and I1 are the initial absolute bank angle and initial time interval respectively. Similarly B2, B3, B4, B5 and B6 are the absolute bank angles at the time interval of I2, I3, I4, I5 and I6 respectively. The time at any n$^{th}$ absolute bank angle is summation of all the time intervals till the corresponding n$^{th}$ absolute bank angles. Thus, time of flight at bank angle B2 is summation of the time intervals corresponding to B1 and B2 i.e, I1+I2. Table 3-25 gives the other 19 absolute value of bank angle and their corresponding time interval considered for the trajectory simulations. These values are derived from the IXV nominal bank profile as described by Rodrigo et al. (2016).

**Table 3-25:** Design parameters for trajectory simulation

| No | Absolute bank angle [deg] | Time interval [sec] | No | Absolute bank angle [deg] | Time interval [sec] |
|---|---|---|---|---|---|
| 1 | 0 | 200 | 11 | 47.98 | 14.00 |
| 2 | 2.89 | 93.20 | 12 | 43.74 | 42.00 |
| 3 | 69.42 | 83.15 | 13 | 46.65 | 50.00 |
| 4 | 38.93 | 314.95 | 14 | 48.92 | 14.00 |
| 5 | 39.13 | 14.00 | 15 | 43.1 | 23.00 |
| 6 | 43.36 | 161.10 | 16 | 66.77 | 29.00 |
| 7 | 52.87 | 10.80 | 17 | 73.11 | 19.00 |
| 8 | 42.32 | 52.70 | 18 | 74.57 | 6.00 |
| 9 | 47.13 | 31.00 | 19 | 72.54 | 8.00 |
| 10 | 48.37 | 66.10 | 20 | 0 | 17.00 |

For the comparing results with the nominal trajectory, it is important that the simulations are performed at the same initial conditions. Along with the inputs described in Table 3-24 the initial bank angle and the time interval for the first bank angle is considered as 0 deg and 200 sec, to match with the nominal trajectory initial conditions. For these inputs the trajectory simulation

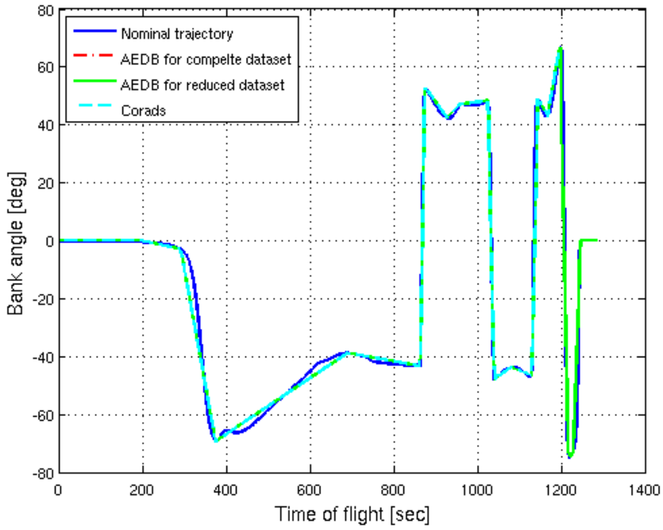



**Figure 3-16:** Comparison of bank-angle profile with nominal trajectory of IXV

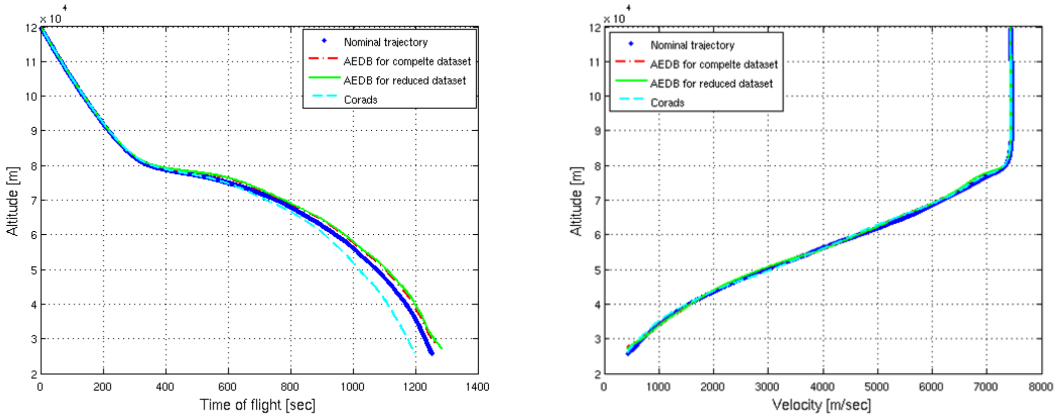
is performed initially for the complete set of aerodynamic data set with 17 angle of attack and 15 Mach numbers. The results derived are compared with the nominal trajectory of IXV.

Furthermore, the aerodynamic database corresponding to Mach number of 4, 10, 17.75 and 25 and angle of attack of 40, 45 and 50 deg are selected and the trajectory simulations are performed for this reference aerodynamic data set as well as for the aerodynamic data set generated from the CoRADS for corresponding Mach number and angle of attack. The following plots gives the comparison of these results. Figure 3-16 gives the bank angle profile for all the three simulations and their comparison with the nominal trajectory. From the figure, it is seen that the bank angle profiles for all the three simulations are matched with the nominal bank angle profile. As the linear interpolation is used within this simulator, there are slight deviation from the nominal bank angle profile, where the profile have more curved profile.

It is also observed that for the trajectory simulation considering the aerodynamic database from CoRADS, the time of flight is less than for the nominal trajectory, whereas the bank angle profile for the both the simulations using the reference aerodynamic database (complete and reduced) of IXV gives time of flight close to the nominal trajectory. This difference can be explained by the less accuracy of CoRADS tool for lower Mach as described in Section 3-1-2. Additionally, in Figure 3-14b, it is observed that as the M decreases, the difference between the drag co-efficient computed by CoRADS and the nominal increases.

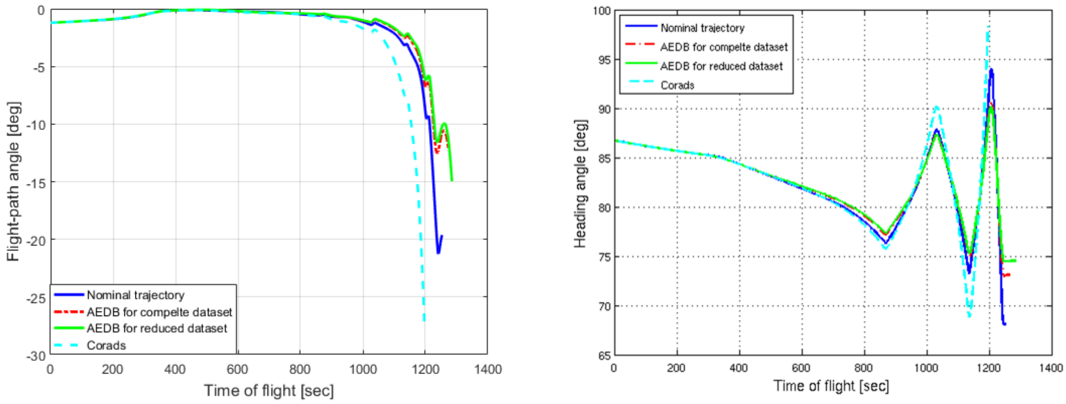
Figure 3-17a and Figure 3-17b gives the comparison of the altitude time profile and altitude-velocity profile. These trajectory profiles are compared with the nominal trajectory of IXV. For both the reference aerodynamic database these profiles are very close to each other as well as match with the nominal trajectory, thus indicates that there is no such effect on the trajectory if the aerodynamic database is considered only for few Mach number and angle-of-attack combinations. Furthermore, for the trajectory simulation considering the aerodynamic database computed from CoRADS gives less time of flight as the CoRADS is based on Newtonian method and the accuracy of the tool is less for the lower Mach number.

Similar deviations of the trajectory simulations using the aerodynamic database from CoRADS tool from the nominal trajectory is observed for the flight-path angle ($\gamma$) and the heading angle ($\chi$) profile as shown in Figure 3-18a and 3-18b respectively. As the Mach number decreases the deviation from the nominal trajectory increases, this can be explained by the accuracy of the CoRADS tool.




**(a)** Altitude profile **(b)** Altitude-velocity profile

**Figure 3-17:** Comparison of the altitude and velocity profile with the nominal trajectory of IXV



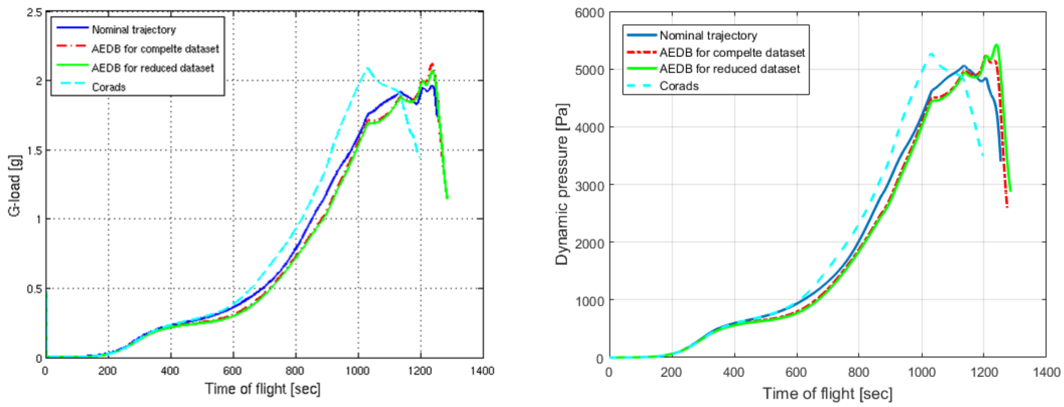

**(a)** Flight-path angle profile **(b)** Heading angle profile

**Figure 3-18:** Comparison of the flight-path angle and the heading angle profile with the nominal trajectory of IXV

Figure 3-19a and Figure 3-19b gives the comparison of the g-load and dynamic pressure profile comparison with the nominal trajectory profile. It is observed that as the time of flight reduces for the trajectory simulation considering the aerodynamic database from CoRADS, the peak of the g-load and dynamic pressure also occurs before the peak occurred in the nominal case.

Figure 3-20a adn Figure 3-20b gives the heat flux and heat load profile. Here the heat flux is computed by the DKR formulation. From the comparison, it is observed that for the trajectory simulation considering the both the reference aerodynamic database gives a similar profile, thus there is no influence of reduced aerodynamic database on heat flux and heat load profile. Furthermore, the peak heat flux also match with the nominal heat flux peak. The trajectory profile generated by using the aerodynamic database from the CoRADS gives a slight deviation and can be explained by the same reason of the accuracy of the CoRADS tool for lower Mach number.

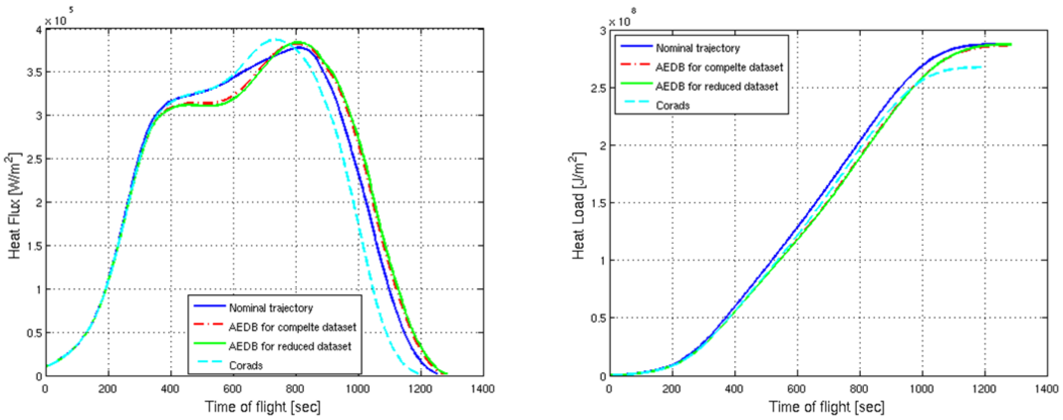
Figure 3-21a and Figure 3-21b gives the comparison of the latitude ($\delta$) and the longitude ($\tau$) profile respectively. From the comparison it is observed that, as the Mach number gets lower than 10




**(a)** G-load profile

**(b)** Dynamic pressure profile

**Figure 3-19:** Comparison of the G-load and the dynamic pressure with the nominal trajectory of IXV



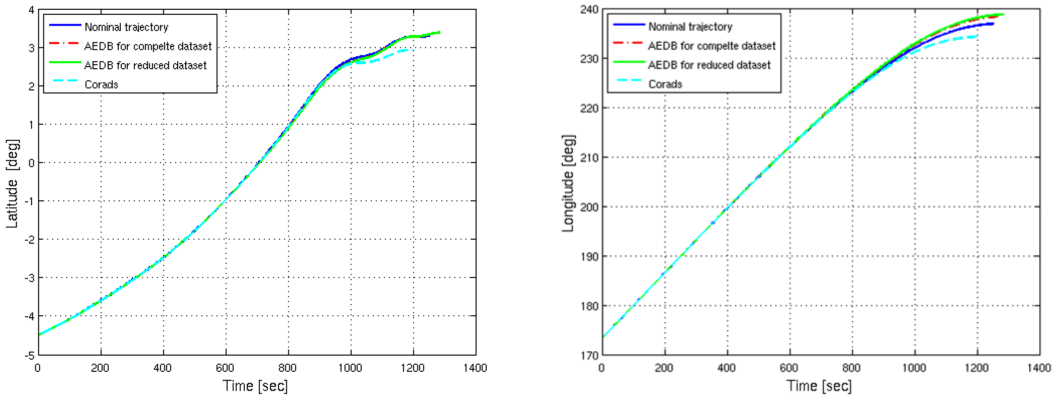

**(a)** Heat flux profile

**(b)** Heat load profile

**Figure 3-20:** Comparison of the heat flux and heat load profile with the nominal trajectory of IXV

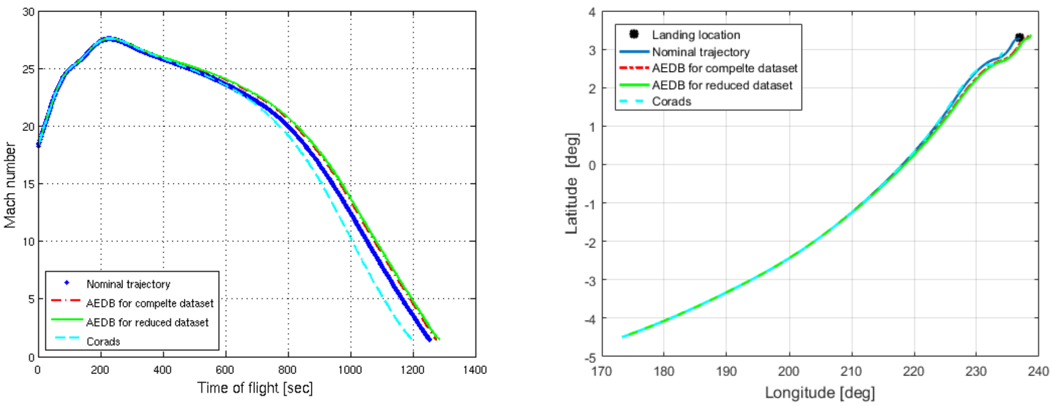
as shown in Figure 3-22a, the accuracy of the CoRADS tool decreases and for the corresponding aerodynamic database there is large deviation observed in the latitude and the longitude profile compared to the trajectory profile obtained by using the reference aerodynamic database.

Figure 3-22b gives the latitude-longitude profile comparison for all the three simulation and their comparison with the nominal trajectory. The trajectory simulations using the reference aerodynamic database reach the target landing location, whereas if the aerodynamic database generated from CoRADS is used it falls short to reach the target. All the trajectory profiles match with the nominal case and thus the trajectory simulator is verified.




**(a)** Latitude profile **(b)** Longitude profile

**Figure 3-21:** Comparison of the latitude and the longitude profile of the optimized trajectories with the nominal trajectory of IXV




**(a)** Mach number profile **(b)** Latitude-logitude profile

**Figure 3-22:** Comparison of the Mach number profile and the latitude-longitude profile with the nominal trajectory of IXV

## Modified CoRADS

The Modified CoRADS is a tool that computes the heat flux using the Fay-Riddel method, which is based on boundary-layer equations and similarity transformation. This tool is developed by TASI as described by CoRADS Development team (2016). The tool requires the mesh of the configuration as well the altitude and Mach number as inputs. For the given mesh of the configuration this tool defines the elements and their corresponding pressure, velocity and temperature to create the initial streamline matrix. Using this streamline matrix, the local stagnation heat flux value is computed using the Fay and Riddle heat relation.

At 69.40 km and 20.95 Mach the maximum heat flux at the stagnation point is computed by this tool for the given IXV mesh. Figure 3-23 gives the heat flux profile at the stagnation point, where it is observed that the peak of the maximum heat flux match with the nominal heat flux of the IXV. CoRADS Development team (2016) describes the nominal heat flux of the IXV, where

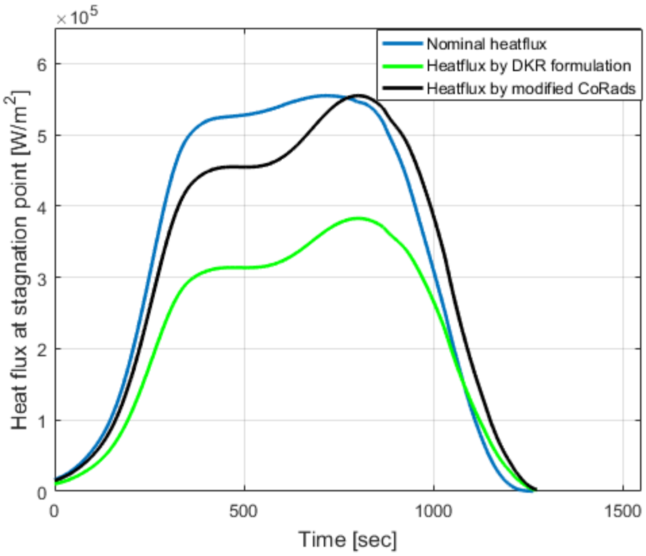



**Figure 3-23:** Heat flux profile at stagnation point

the results obtained from the modifed CoRADS are further scaled by 1.23 times to consider the uncertainties considered for the IXV test case.

Additionally, this tool computes the maximum heat flux value and not the profile with the time. Thus, to plot this profile the heat flux profile obtained by DKR method as shown in Figure 3-20a is scaled with the maximum heat flux obtained from the modified CoRADS. Thus the difference in the peaks of the heat flux also increases as compared to the heat flux profile shown in Figure 3-20a.

Chapter 4

# Verification of the Design Process for Optimization of a Non-Lifting Re-entry Vehicle

Legostaev and Minenko (1994) describe the Russian Raduga capsule as a non-lifting axis symmetric re-entry vehicle, which followed a ballistic trajectory to return the payload from the ISS to the Earth. Choosing a simple configuration for the initial verification of the design process allows one to verify the application of MDO techniques for the re-entry vehicle. It also allows one to plug and play with different tools and improve the necessary interface templates to create a generalized design process within the chosen design framework.

This chapter describes the workflow created for the feasibility study of the Raduga-like capsule in Section 4-1. Furthermore, this section describes the interfaces involved in this workflow and highlights the interface verification. Section 4-2 gives the system level verification of the design process. This verification includes the discussion of the results of the sensitivity analysis and the MDO of the Raduga-like configuration using the developed design process within the Dakota framework.

## 4-1 Workflow and Interface Verification

Chapter 2 described the design process developed within the Dakota framework. This section describes the implementation of the developed design process for the feasibility study of a re-entry vehicle, where the axis-symmetric capsule (Raduga) is considered for the verification. The basic workflow described in Figure 2-5 can be compared with the workflow created for the feasibility study of the Raduga-like re-entry vehicle. Figure 4-1 describes the workflow, where the design parameters flow from the Dakota optimizer to the tool-chain and values of the necessary cost functions computed by the tools within the tool-chain, are returned back to the Dakota optimizer. The optimizer analyzes these results and creates a new set of design parameters. The simulations continue until a converged solution is obtained through this closed loop.

A software interface is a bridge that allows two disciplines to share information with each other, even if they are developed using different programming languages. An interface will often use a standard file format such as XML or ASCII to move information from one system to another. For

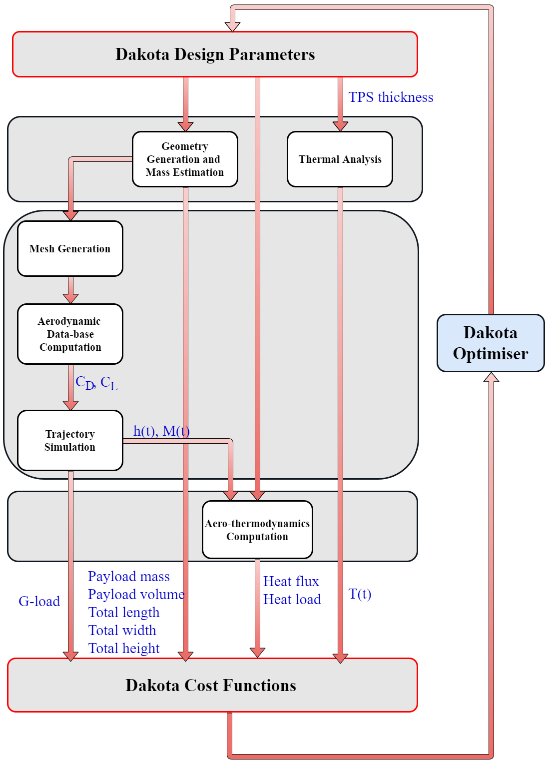



**Figure 4-1:** Workflow for feasibility study of Raduga like capsule

the Dakota framework, ASCII file format is selected to transfer the information as the Dakota optimizer accepts input file and generate an output file in the ASCII format. The tools can be developed in any other programming languages, but once they are compatible with the Dakota framework, the interfaces between them shall enable the user to transfer the data within the tool-chain and eventually automate the design process.

As described in Chapter 2, a interface template is created for each tool as shown in Figure 2-4. Each tool extract the necessary inputs using the dprepro function and fill-up their input interface template. Once the input template is filled with the necessary inputs, the tool executes the code and the results obtained are written in the output template of the tool using the same dprepro function. As described in Chapter 2, the output and the input templates are written in a specific format, such that the dprepro function can read the input as well as write the output in the same format.

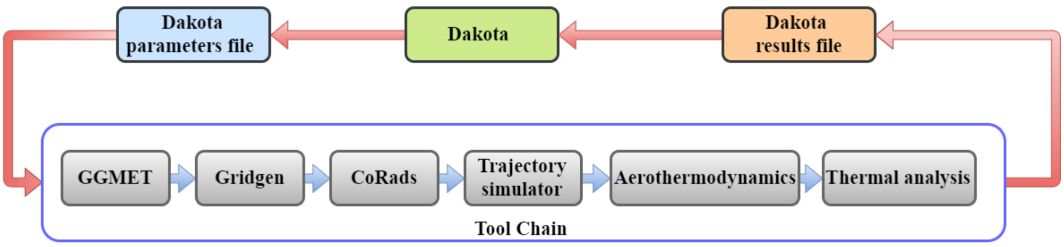



**Figure 4-2:** Toolchain for feasibility study of Raduga like capsule

The tool-chain shown in Figure 4-2 includes 6 software, which require 5 interfaces, additionally there are 2 interfaces between Dakota and tool-chain. GGMET is used for the geometry generation of the Raduga- like configuration and to compute the payload mass of the capsule. Gridgen is a commercial software considered for the mesh generation. The mesh is used to compute the aerodynamic database of the vehicle. To compute these aerodynamic database the CoRADS tool is considered. Furthermore, the trajectory simulator and the modified CoRADS is used for the trajectory simulation and the heat flux computation, respectively. Additionally, a tool for the thermal analysis is considered, which computes the temperature at the cold structure for a given TPS thickness. Chapter 3 described these tools and their unit-test verification. The following paragraphs describes the interface between all the tools and the Dakota framework and their verification.

The Dakota optimiser generates a design parameters file called 'params.in'. This is an ASCII file and is written in a format such that the considered dprepro function can be used to read and extract the necessary parameters from this file. Figure 4-3 shows the parameter file which indicates the values assigned for all the design parameters by the Dakota optimizer. As described in Chapter 2, the dprepro function is used to read the required inputs and fill-up the templates of each tool. To do so, all the templates of the tools are created in the same format as Dakota input and output file, to maintain the consistency in the workflow as well as to utilize a common dprepro function to read inputs and write the output file. Thus, the design process is generalized.

```
{ DAKOTA_VARS     =                      12 }
{ RBo             =  2.492000000000000e-01 }
{ RN              =  5.000000000000000e-01 }
{ L1              =  9.695000000000000e-01 }
{ L2              =  3.140000000000000e-01 }
{ L3              =  5.000000000000000e-02 }
{ theta1deg       =  3.000000000000000e+00 }
{ theta2deg       =  1.600000000000000e+01 }
{ theta3deg       =  0.000000000000000e+00 }
{ TPSThk          =  7.700000000000000e-02 }
{ TPSThkcone2     =  6.240000000000000e-02 }
{ TPSThkcone3     =  5.000000000000000e-02 }
{ TPSThkcone4     =  5.000000000000000e-02 }
```

**Figure 4-3:** Parameter file generated by the Dakota

The interface between the Dakota and the tool-chain makes sure that the design parameters flow correctly to the required tools as input. During the interface verification, it is made sure that all the required parameters are transferred to the input templates of each of the tool. The interface template was described in Chapter 2, Figure 2-4.

Initially, the design parameters required to generate the geometry is extracted by the GGMET from the params.in file. The input parameters for the GGMET includes the length, cone angle and TPS thickness of each section, front cone radius and nose radius. These design parameters flow from the Dakota parameter file (params.in) to input template of GGMET. The tool executes the code using this inputs and generate the necessary parameters like total length, width, height, payload mass and payload volume of the capsule along with the coordinates for mesh generation. These outputs are stored in the GGMET output file.

Table 4-1 gives the interface between Dakota design parameter file and GGMET input file. From the interface verification, it is checked that exactly same values from Dakota design parameter file are transferred to the input template of the GGMET.

Furthermore the GGMET generates the geometry and the output from this tool is given as input to the Gridgen, which is used for the mesh generation. The interface between GGMET should be able to transfer the coordinates of new configuration as well as generate the mesh automatically. A glyph script is used to generate the mesh automatically in a workflow. Thus, an interface using

**Table 4-1:** Interface between Dakota design parameter file and GGMET input file

| Parameter | GGMET input file | Unit |
|---|---|---|
| Front cone nose radius (RBo) | 0.25 | [m] |
| Nose radius (RN) | 0.50 | [m] |
| Length of first section (L1) | 0.97 | [m] |
| Length of second section (L2) | 0.31 | [m] |
| Length of third section (L3) | 0.05 | [m] |
| Cone angle of first section ($\theta1$) | 3.00 | [deg] |
| Cone angle of second section ($\theta2$) | 16.00 | [deg] |
| Cone angle of third section ($\theta3$) | 0.00 | [deg] |
| TPS thickness of nose section (TPSthk) | 0.07 | [m] |
| TPS thickness of first section (TPSthk1) | 0.06 | [m] |
| TPS thickness of second section (TPSthk2) | 0.05 | [m] |
| TPS thickness of third section (TPSthk3) | 0.05 | [m] |

this glyph script is developed between GGMET and the Gridgen, where the co-ordinates of new geometry are automatically extracted from the output of the GGMET to create a new mesh. As the glyph script is also a ASCII file, thus the same file is converted into a similar template, where dprepro function is used again to read the co-ordinates from the GGMET and fill-up the input template of the Gridgen (glyph script template). Table 4-2 gives the interface between GGMET output file and Gridgen input template, where it is observed from the interface verification that exactly same values flow from the output file of GGMET to the input template of Gridgen.

**Table 4-2:** Interface between GGMET output file and Gridgen input template

| Parameter | Gridgen input file | Unit |
|---|---|---|
| Ax | 0.000 | [m] |
| Ay | 0.000 | [m] |
| Bx | 0.066 | [m] |
| By | 0.249 | [m] |
| Cx | 1.036 | [m] |
| Cy | 0.300 | [m] |
| Dx | 1.350 | [m] |
| Dy | 0.390 | [m] |
| Ex | 1.400 | [m] |
| Ey | 0.390 | [m] |
| Ox | 1.400 | [m] |
| Oy | 0.000 | [m] |

Figure 3-2 described the design parameters and the co-ordinates of the configuration. Once the mesh is created for a given configuration, it is transferred to the aerodynamics computation tool, CoRADS. The mesh file is saved in the working directory of the simulation and is directly read by CoRADS to compute the aerodyanmic database. The output of the mesh generation is in the .stl format, whereas the aerodynamics computation tool requires input in the .mat format. Thus, a bridging function to convert this format from the .stl to .mat is used as an interface for the data transfer. This function is included in the CoRADS. The derived values of the aerodynamic coefficients are used to compute the ballistic or lifting parameter. The total mass of the vehicle is considered as constant and thus this value is included in the CoRADS as fixed input. The additional parameters such as the surface area and the reference length are also required to be transferred from the GGMET output file to the input template of CoRADS. For the given Mach number and altitude the aerodynamic coefficients are computed by CoRADS. Table 4-3 gives the interface between the GGMET and the CoRADS. From the interface verification, it is observed that these values flow from the output file of GGMET to the input file of CoRADS.

**Table 4-3:** Interface between GGMET output file and CoRADS input file

| Parameter | CoRADS input file | Unit |
|---|---|---|
| Surface area | 0.47 | [m$^2$] |
| Reference length | 1.40 | [m] |

Once the aerodynamic database is generated, it is further transferred to the trajectory simulator. The drag co-efficient and the corresponding ballistic coefficient is computed by the CoRADS. These data flow from the CoRADS output file to the input template of the trajectory simulator. Additionally, the trajectory simulator requires the radius of nose and the reference diameter computed from the GGMET tool. Here the reference diameter is twice the maximum base radius. Thus, this interface will extract all the required values from both coupled disciplines and transfer the necessary data to trajectory simulator. Table 4-4 and Table 4-5 gives the interface between the output file of GGMET and input template of FMST and interface between output file of CoRADS and the input template of the trajectory simulator respectively.

**Table 4-4:** Interfaces between output file of GGMET and FMST input file

| Parameter | FMST input file | Unit |
|---|---|---|
| Nose radius (RN) | 0.50 | [m] |
| Reference diameter | 0.78 | [m] |

**Table 4-5:** Interfaces between output file of CoRADS and FMST input file

| Parameter | FMST input file |
|---|---|
| Drag coefficient ($C_D$) | 0.70 |
| Ballistic coefficient | 1032 |

The trajectory data is required as an input for the heat-flux computation. The trajectory simulator computes the heat flux using the DKR formulation. The altitude and the Mach number at which maximum heat flux occurs by DKR formulation is considered for the heat flux computation by modified CoRADS. The trajectory data is saved in the working directory of the simulation and since trajectory simulator and the modifed CoRADS both are developed using Matlab, this trajectory data is directly read by the modified CoRADS to compute the atltitude and Mach number at which maximum heat flux occurs by DKR formulation. The locations at which the heat flux is computed in given as input to this tool. Thus, there is an interface between the GGMET and the modified CoRADS. Table 4-6 gives the interface between the GGMET and the modified CoRADS, where the design parameters flow from the output of the GGMET to the input template of the modified CoRADS.

Further, the TPS thermal analysis tool is a Fortran script. The TPS thickness and material properties are required as input for the TPS thermal analysis. The material properties are considered as constant for the MDO of Raduga configuration. There is an interface between the Dakota parameter file (params.in) and TPS thermal analysis input template. Table 4-7, gives the flow of the design parameters from the Dakota parameters file (params.in) to the input template of the thermal analysis tool. It computes the maximum temperature at the cold structure and TPS recession.

**Table 4-6:** Interface between the GGMET and the modified CoRADS

| Parameter | modified CoRADS input file | Unit |
|---|---|---|
| Bx | 0.0665 | [m] |
| Cx | 1.0360 | [m] |
| Dx | 1.3500 | [m] |
| Ex | 1.4000 | [m] |

**Table 4-7:** Interface between the Dakota parameter file and the thermal analysis tool

| TPS thickness | Thermal analysis tool input file | Unit |
|---|---|---|
| Nose section | 0.070 | [m] |
| First section | 0.062 | [m] |
| Second section | 0.050 | [m] |
| Third section | 0.050 | [m] |

Once the simulations are completed from all the tools, the values of cost functions are send to the Dakota results template. Thus, there is an interface between the results template and the output files of all the necessary tools.

**Table 4-8:** Interface between the output file of GGMET and Dakota results file

| Parameter | Dakota results file | Unit |
|---|---|---|
| Payload mass | 150 | [kg] |
| Payload volume | 0.12 | [$m^3$] |
| Total length of vehicle | 1.40 | [m] |
| Maximum base radius | 0.39 | [m] |

**Table 4-9:** Interface between the output file of FMST and Dakota results file

| Parameter | Dakota results file | Unit |
|---|---|---|
| G−load | 9.2 g | [$m/s^2$] |

Table 4-8 gives the flow of information from the GGMET output file to the Dakota results template. The payload mass and volume is considered further as objective of the optimization, whereas for the selection of the launcher the total length and the maximum base radius are considered as the constraints. The g-load is considered as constraint for the optimization. Table 4-9 gives the flow of this information from the trajectory simulator output file to the Dakota results file supplied to the Dakota optimizer.

Table 4-10 gives transfer of the value of the maximum heat flux at stagnation point computed by the modified CoRADS to the Dakota results template. Table 4-11 gives the flow of information from the thermal analysis tool output file to the Dakota results file. From the above results, it is checked that the information from one tool to other is transferred through the interfaces. The following section describes the system level verification of design process.

**Table 4-10:** Interface between the modified CoRADS and Dakota results file

| Parameter | Dakota results file | Unit |
|---|---|---|
| Maximum heat-flux at stagnation point | 3.75 | [MW/m$^2$] |

**Table 4-11:** Interface between the thermal analysis tool and Dakota results file

| Parameter | Dakota results file | Unit |
|---|---|---|
| TPS recession at nose section | 0.003 | [m] |
| Maximum temperature at nose section | 480 | [K] |
| Maximum temperature at first section | 480 | [K] |
| Maximum temperature at second section | 480 | [K] |
| Maximum temperature at third section | 480 | [K] |

# 4-2 System Verification

This section describes the system verification of the design process used for the feasibility study of the Raduga-like capsule. The feasibility study includes the MDO of the Raduga capsule. To perform the MDO of the re-entry vehicle, it is important to perform the sensitivity analysis to analyse the influence of the design parameters on the cost functions. Subsection 4-2-1 gives the sensitivity analysis of the Raduga configuration. Furthermore, using the sensitivity analysis Subsection 4-2-2 gives the results of the MDO.

## 4-2-1 Sensitivity Analysis

As described in Section 2-4, the design of experiments is performed to explore the input domain or the design space to gain the better understanding of the range in the output, for a particular input domain. From the design of experiments, one can also observe which inputs have the most influence on the output. Here, this is considered as the sensitivity analysis. Adams (2009) describes the LHS as one of the stochastic method used for Distributed Design and Analysis of Computer Experiments (DDACE). In addition to obtain the statistical summary, the LHS method within the Dakota framework computes the simple and partial raw correlations using Pearson's correlation coefficient; and simple and partial rank correlations using Spearman rank correlation coefficient as described in Section 2-4. The raw correlations refer to correlations performed on the actual input and output data. Rank correlations refer to correlations performed on the ranks of the data, which gives the correlation between two variables by controlling the influence of the third variable. This gives the perfect idea about the correlation between two variables involved in a multi-variable problem.

Table 4-12 gives the design parameters considered for LHS. 1000 samples are computed using the LHS. For these samples, the design parameters are varied to $\pm50$ % from the reference Raduga capsule parameters as described by Legostaev and Minenko (1994) to explore the design space. Figure 4-4 gives the design parameters mentioned in Table 4-12.

Using the samples generated by the LHS, the partial rank correlation factors are computed using the Spearman's correlation as given in Equation 2-6. The LHS method within the Dakota framework provide the facility to compute the partial rank correlation factors.

Figure 4-5 gives the partial rank correlation factors between the each design parameter and the objectives such as payload mass and payload volume. The value of the partial rank correlation

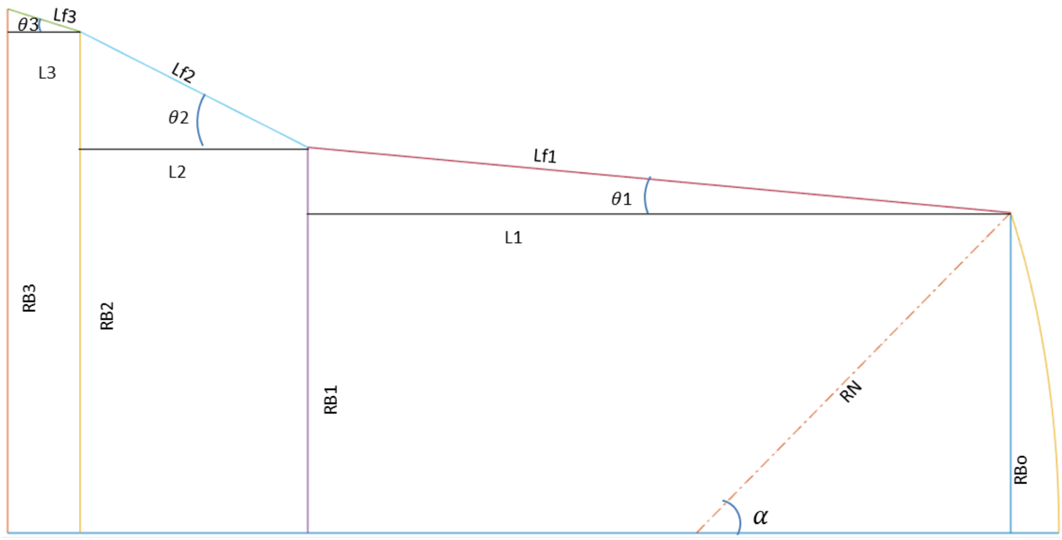



**Figure 4-4:** Parameterisation of the curve generating the solid of revolution

**Table 4-12:** Design parameters

| Parameter | Value | Upper bound | Lower bound | Unit |
|---|---|---|---|---|
| Front cone radius (RBo) | 0.24 | 0.37 | 0.12 | [m] |
| Nose radius (RN) | 0.50 | 0.75 | 0.25 | [m] |
| Length of first section (L1) | 0.96 | 0.75 | 0.48 | [m] |
| Length of second section (L2) | 0.31 | 0.47 | 0.16 | [m] |
| Length of third section (L3) | 0.05 | 0.07 | 0.02 | [m] |
| Cone angle of first section ($\theta 1$) | 3.00 | 4.5 | 1.5 | [deg] |
| Cone angle of second section ($\theta 2$) | 16.00 | 24 | 8 | [deg] |
| Cone angle of third section ($\theta 3$) | 0.00 | 1.0 | 0.0 | [deg] |
| TPS thickness of nose section (TPSthk) | 0.07 | 0.08 | 0.05 | [m] |
| TPS thickness of first section (TPSthk1) | 0.06 | 0.08 | 0.02 | [m] |
| TPS thickness of second section (TPSthk2) | 0.05 | 0.08 | 0.02 | [m] |
| TPS thickness of third section (TPSthk3) | 0.05 | 0.08 | 0.02 | [m] |

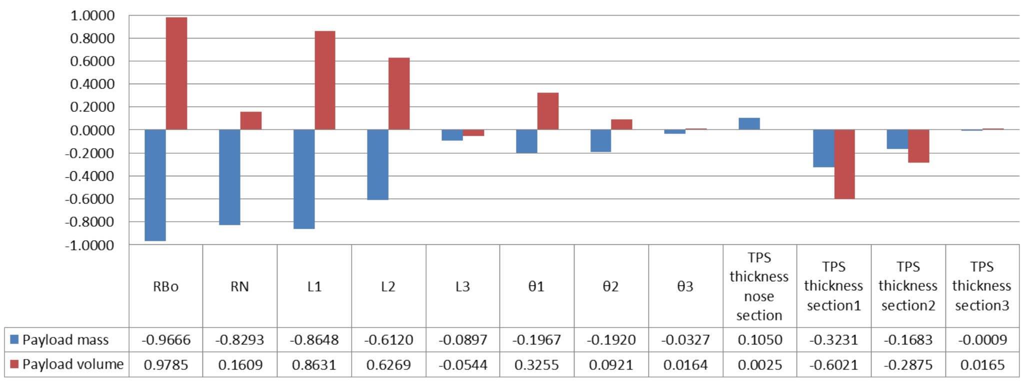


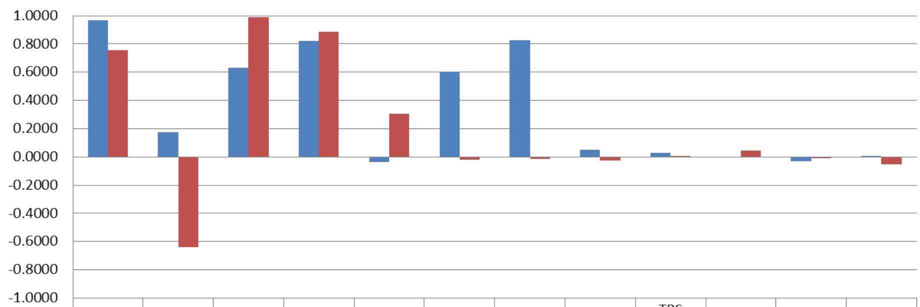

**Figure 4-5:** Partial rank correlations

varies from -1to 1, where -1 indicates inverse linear correlation and +1 indicates positive correlation. 0 indicates no correlation between the parameters. The higher the absolute value of this partial rank correlation factor, the stronger is the correlation. The absolute value of the partial rank correlation factors less than 0.2 is considered as a very small influence.

The GGMET is responsible to compute the payload mass and payload volume. The total mass of the vehicle is considered as constant, whereas the variation in the configuration influence the TPS and the cold structure mass. Scaling up of the vehicle eventually increase the lengths and base radius. This increases the TPS and the cold structure mass and thereby reduces the payload mass capacity of the vehicle. This results into inverse correlations between the design parameters and the payload mass. Whereas the increase in the lengths and the base radius of the vehicle increase the payload volume capacity. The payload section is described in the first and the second section. Thus, the $L1$ and $L2$ have strong correlation than the other design parameters. Additionally, the $RBo$ is the front cone radius and the increase in $RBo$ eventually increase the $RB1$ and $RB2$. Thus, $RBo$ gives a strong positive correlation between payload volume. The length of the nose section is given by the following relation:

$$Length\, of\, nose\, section = RN(1 - \cos\alpha) \tag{4-1}$$

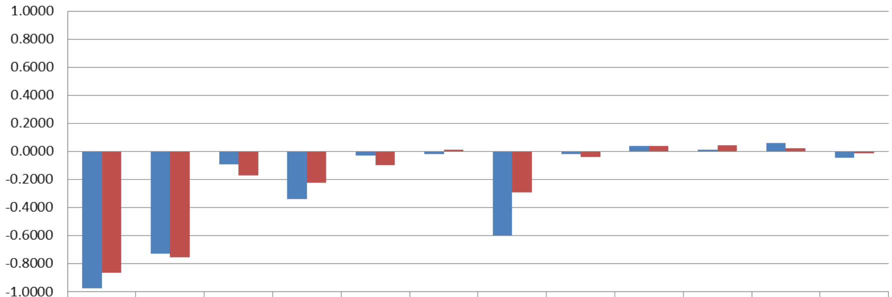
where, sin$\alpha = \frac{RN}{RBo}$. Thus, the length of the nose is influenced by the radius of nose and front cone radius. The increase in the length increase the cold structure mass as well as the TPS mass. This is indicated from the correlation between the RN and RBo with the payload mass respectively. Furthermore, $L1 \geq L2 \geq L3$ thus $L1$ strongly influence the payload mass than $L2$. The $L3$ is very small compared to the other lengths and thus its influence is very less. The cone angle of each section indirectly influence the length of the section and thus the influence of change in the cone angle is mitigated with the variation of the length of each sections. Even though the thickness of the TPS is having less influence on the payload mass, but the volume of the TPS is directly related to the length of the sections, which thereby increase the TPS mass. The higher the TPS thickness, higher is the mass of the TPS and lower is the payload mass capacity. Additionally, if the TPS thickness is higher, the payload volume is lower. Since section 1 is larger than section 2, the influence of the TPS thickness of section 1 is also higher than section 2.




| | RBo | RN | L1 | L2 | L3 | θ1 | θ2 | θ3 | TPS thickness nose section | TPS thickness section1 | TPS thickness section2 | TPS thickness section3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Maximum base radius | 0.9678 | 0.1751 | 0.6295 | 0.8219 | -0.0382 | 0.6031 | 0.8253 | 0.0521 | 0.0312 | -0.0015 | -0.0306 | 0.0021 |
| Maximum total length | 0.7543 | -0.6418 | 0.9883 | 0.8858 | 0.3059 | -0.0209 | -0.0173 | -0.0267 | 0.0075 | 0.0450 | -0.0089 | -0.0539 |

**Figure 4-6:** Partial rank correlations

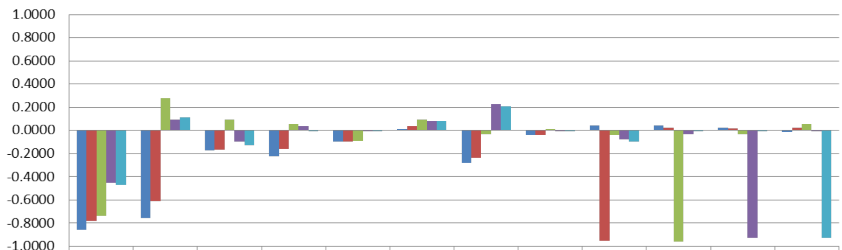
Figure 4-6 gives the correlations between the total base radius and total length of the vehicle. These results are computed from the GGMET. The higher the lengths of the section higher will be the total length of the vehicle and similarly, higher the base radius higher is the maximum base radius of the vehicle. The base radius is directly related to the nose base radius and the cone angle of each section. This is seen from the co-relations factors that $RBo$ and $RN$ influence the nose length, but the $RBo$ and $RN$ have inverse relation. If RN increases then the nose section length decreases and thus gives a inverse correlation with the total length of the vehicle, whereas if $RBo$ increases the nose section length increases and gives a positive correlation and can be also analyzed from Equation 4-1. Furthermore, the nose base radius and length of each section also influence base radius as given in Equation 3-3. Thus, higher the length and cone angle of each section, higher is the base radius.




| | RBo | RN | L1 | L2 | L3 | θ1 | θ2 | θ3 | TPS thickness nose section | TPS thickness section1 | TPS thickness section2 | TPS thickness section3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| G-load | -0.9736 | -0.7266 | -0.0929 | -0.3380 | -0.0304 | -0.0208 | -0.5982 | -0.0206 | 0.0390 | 0.0115 | 0.0626 | -0.0472 |
| Heatflux at stagnation point | -0.8658 | -0.7525 | -0.1706 | -0.2247 | -0.0985 | 0.0143 | -0.2910 | -0.0378 | 0.0370 | 0.0428 | 0.0248 | -0.0148 |

**Figure 4-7:** Partial rank correlations

Figure 4-7 gives the partial rank correlation factor of all design parameters with g-load and maximum heat flux at the stagnation point. As the g-load depends on the drag coefficient Concurrent Design (CD) and surface area. Thus the design parameters that influence CD and surface area are highly co-related with the g-load. The aerodynamic data base is computed using the Newtonian method, thus the highest cone angle strongly influence the CD. Additionally, the maximum base radius influence the surface area of the vehicle. The maximum base radius is influenced by the front cone radius and the nose radius. Thus, the maximum g-load is also influenced by the radius of the nose and the front nose radius. The heat-flux is having high co-relation between the radius of the nose and the front nose radius. The radius of nose and the front cone radius given by the relation: sin$\alpha$=(RBo/RN), where $\alpha$ is the nose cone angle. The front cone radius also influence the maximum-heat flux. The heat flux is strongly co-related with the $RBo$ and $RN$, where $RBo$ influence larger than $RN$ is due to this formulation within the GGMET.




| | RBo | RN | L1 | L2 | L3 | θ1 | θ2 | θ3 | TPS thickness nose section | TPS thickness section1 | TPS thickness section2 | TPS thickness section3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TPS recession in nose section | -0.8564 | -0.7534 | -0.1704 | -0.2221 | -0.0989 | 0.0138 | -0.2799 | -0.0367 | 0.0447 | 0.0418 | 0.0230 | -0.0146 |
| Max Temperature at nose section | -0.7839 | -0.6071 | -0.1643 | -0.1609 | -0.0979 | 0.0369 | -0.2344 | -0.0401 | -0.9516 | 0.0251 | 0.0183 | 0.0246 |
| Max Temperature at first section | -0.7374 | 0.2806 | 0.0927 | 0.0564 | -0.0887 | 0.0941 | -0.0314 | 0.0093 | -0.0385 | -0.9564 | -0.0312 | 0.0564 |
| Max Temperature at second section | -0.4542 | 0.0955 | -0.0951 | 0.0397 | -0.0057 | 0.0822 | 0.2257 | -0.0102 | -0.0760 | -0.0349 | -0.9254 | -0.0100 |
| Max Temperature at third section | -0.4678 | 0.1151 | -0.1296 | -0.0038 | -0.0007 | 0.0814 | 0.2053 | -0.0054 | -0.0936 | -0.0033 | -0.0039 | -0.9265 |

**Figure 4-8:** Partial rank correlations

Figure 4-8 gives the partial rank correlation factor between of all the design parameters and the maximum temperature at the cold structure of the respective section and the TPS recession at the nose section, respectively. The TPS thickness of the each section strongly influence the maximum temperature of the cold structure of the respective section. The heat flux influence at stagnation point influence the maximum temperature at cold structure. From Figure 4-7, there is a strong negative correlation between the $RN$, $RBo$ and heat flux at the stagnation point, thus there is also strong influence of the $RBo$ and $RN$ over the maximum temperature at the nose section. The TPS recession at the nose section is influenced by the $RN$ and $RBo$, as these parameters also influence the maximum temperature at the nose section. The higher the temperature at the cold structure higher is the recession.

**Table 4-13:** Best set of design parameters for initialization of MDO

| Parameter | Value | Unit |
|---|---|---|
| Front cone radius (RBo) | 0.29 | [m] |
| Nose radius (RN) | 0.48 | [m] |
| Length of first section (L1) | 0.61 | [m] |
| Length of second section (L2) | 0.27 | [m] |
| Length of third section (L3) | 0.58 | [m] |
| Cone angle of first section ($\theta 1$) | 2.49 | [deg] |
| Cone angle of second section ($\theta 2$) | 15.53 | [deg] |
| Cone angle of third section ($\theta 3$) | 0.11 | [deg] |
| TPS thickness of nose section (TPSthk) | 0.075 | [m] |
| TPS thickness of first section (TPSthk1) | 0.065 | [m] |
| TPS thickness of second section (TPSthk2) | 0.072 | [m] |
| TPS thickness of third section (TPSthk3) | 0.071 | [m] |

**Table 4-14:** Values of the cost function for the given best set of design parameters

| | | |
|---|---|---|
| Payload mass | 156 | [kg] |
| Payload volume | 0.109 | [$m^3$] |
| Maximum heat flux at stagnation point | 2.91 | [$MW/m^2$] |
| Maximum G-load | 92.1 g | [$m/s^2$] |
| Maximum base radius | 0.396 | [m] |
| Maximum total length | 1.038 | [m] |
| Maximum TPS recession at nose section | 0.001 | [m] |
| Maximum temperature at nose section | 474.46 | [K] |
| Maximum temperature at first section | 457.51 | [K] |
| Maximum temperature at second section | 388.10 | [K] |
| Maximum temperature at second section | 394.08 | [K] |

The LHS within the Dakota framework also gives an best of design parameters, which can be used for further feasibility study. In this case, this design parameters are selected to initiate the MDO of the Raduga capsule. Table 4-13 gives this set of design parameters and Table 4-14 gives the corresponding values of the cost functions.

## 4-2-2 Optimization Results

From the above sensitivity analysis, the complete design space is explored using the LHS. To verify the design process for the MDO Raduga capsule the optimization problem is setup. The LHS also provided the best set of design parameters and the corresponding values for the cost function. This solution also gives an estimate of the location of the optimized solution within the design space.

To perform the optimization, the MOGA should be tuned according to the requirements of the problem. Table 4-15 gives the considered tunings for the optimization. As described in Chapter 2, the output option controls the amount of information provided to the user, for this design process debug option is considered to derive the maximum information possible. The random seed control the mechanism for making a stochastic method repeatable, thus a default value of seeds are used. The maximum function evaluation option describes the stopping criteria of the simulations. If converge criteria is not mentioned, the simulations stops at maximum evaluations, but if it is mentioned it stops, when solutions converge. As this is user defined, thus maximum value is selected based on the initial trial simulations. The value is selected such that simulation stops only once there is convergence. In this case, it is considered as 5000, where the simulation stops as the solutions meet the convergence criteria. Metric tracker is used for the convergence, where the percentage change in the generation and number of generations considered for the convergence. The tracker observes the variations from generation to generation in the non-dominated frontier. As these changes fall below a user specified threshold, for a user specified number of generations, the algorithm stops and the solution is considered as converged solution. For this case, the percentage change is considered as 0.1 for 5 successive generations as a convergence criteria.

**Table 4-15:** Tuning of the MOGA

| Tuning parameters | Unit |
|---|---|
| Output | debug |
| Seed | 20000 |
| Final solutions | 30 |
| Max function evaluations | 5000 |
| Initialization type | unique random |
| Cross over type | shuffle random |
| Number of offspring | 2 |
| Number of parents | 2 |
| Crossover rate | 0.8 |
| Mutation type | offset uniform |
| Mutation scale | 0.15 |
| Mutation rate | 0.08 |
| Fitness type | domination count |
| Replacement type | elitist |
| Convergence type | metric tracker |
| Percent change | 0.1 |
| Number of generations | 5 |

The final solution option describes the number of final optimized solutions derived by Dakota. In this case, 30 optimized solutions are derived. The unique random initialization method is selected to avoid duplication of a solution. Shuffle random is considered as a crossover method, as choosing this the design variables are randomly shuffled from a specified number of parents enough times, such that the requested number of children are produced. Since Shuffle random is useful, when there are more design parameters and the user is unable to define which parameters are crossed this method is considered for the crossover. Furthermore, the default value of the crossover rate is maintained for the simulation with standard value of 2 parents and 2 offspring. Once the crossover is performed the mutation process is carried out. For mutation, the offset uniform method is considered as it introduces random variation by adding a uniform random amount to a variable value. The random amount depends on the mutation scale. The mutation rate controls the number of mutation performed. For this case, the default value of the mutation scale and the mutation rate is considered as 0.15 and 0.08, respectively. The new population is evaluated by the fitness assessor. In this case the domination count is considered as the fitness assessor as it works by ordering population members by the negative of the number of designs that dominate them. The higher the fitness number, the better is population. Furthermore, once, the fitness of the population is assessed, it is required to replace the population with members selected to continue in the next generation. To select the members, the Elitist selector is considered as it chooses the required number of designs taking the most fit to the next generation. Table 4-16 gives the design parameters considered for the MDO of Raduga like re-entry capsule.

**Table 4-16:** Design parameters

| Parameter | Initial Value | Lower bound | Upper bound | Unit |
|---|---|---|---|---|
| Front cone nose radius (RBo) | 0.25 | 0.12 | 0.37 | [m] |
| Nose radius (RN) | 0.50 | 0.25 | 0.75 | [m] |
| Length of first section (L1) | 0.97 | 0.48 | 1.45 | [m] |
| Length of second section (L2) | 0.31 | 0.15 | 0.47 | [m] |
| Length of third section (L3) | 0.05 | 0.02 | 0.07 | [m] |
| Cone angle of first section ($\theta1$) | 3.0 | 1.5 | 4.5 | [deg] |
| Cone angle of second section ($\theta2$) | 16 | 8 | 24 | [deg] |
| Cone angle of third section ($\theta3$) | 0.5 | 0.0 | 1.0 | [deg] |
| TPS thickness of nose section | 0.07 | 0.05 | 0.08 | [m] |
| TPS thickness of first section | 0.06 | 0.02 | 0.08 | [m] |
| TPS thickness of second section | 0.05 | 0.02 | 0.08 | [m] |
| TPS thickness of third section | 0.05 | 0.02 | 0.08 | [m] |

**Table 4-17:** Objectives for the optimization

| Objectives | |
|---|---|
| Payload mass | Maximize |
| Payload volume | Maximize |

**Table 4-18:** Constraints for the optimization

| Constraints | |
|---|---|
| Maximum heat flux at stagnation point | $\leq$ 3.6 [MW/m$^2$] |
| Maximum G-load | $\leq$ 92 g [m/s$^2$] |
| Maximum base radius | $\leq$ 0.39 [m] |
| Maximum total length | $\leq$ 1.4 [m] |
| Maximum TPS recession at nose section | $\leq$ 0.003 [m] |
| Maximum temperature at nose section | $\leq$ 480 [K] |
| Maximum temperature at first section | $\leq$ 480 [K] |
| Maximum temperature at second section | $\leq$ 480 [K] |
| Maximum temperature at second section | $\leq$ 480 [K] |

Table 4-17 and Table 4-19 gives the objectives and constraints considered for the MDO of Raduga like capsule. The values for the constraints are derived from the reference Raduga configuration as described by Legostaev and Minenko (1994). Furthermore, the maximum temperature at the all the section is defined by the material properties of cold structure. It is also verified from the LHS that there are solutions that can satisfy the given constraints, to ensure that one can derive a converged solution. MOGA is considered as discussed in Chapter 2 as an optimization technique. From the above inputs, the following results are derived from the design process.

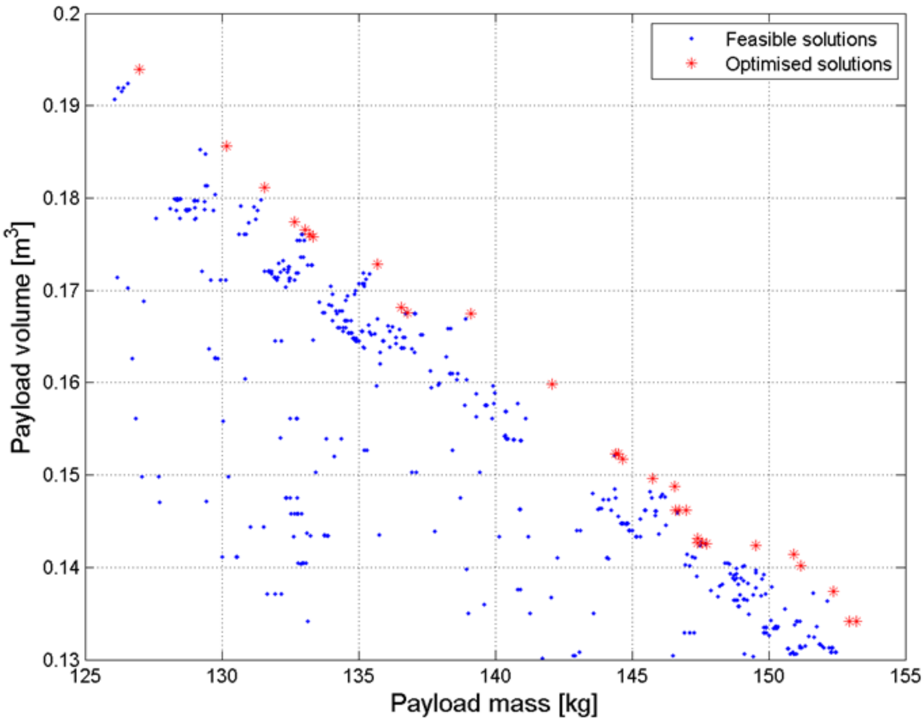



**Figure 4-9:** Pareto front

Figure 4-9 gives the optimization Pareto front between the objectives; payload mass and payload mass. The payload mass and payload volume gives a inverse correlation. As the payload mass capacity increases the payload volume decreases. This can be also explained from the sensitivity analysis results, which indicates that increase in the dimensions increases the payload volume capacity but add on to the TPS and cold structure mass that reduces the payload mass capacity. In this figure, the solutions that satisfy all the constraints (blue) are compared with the optimized solutions (red). From the figure, it is observed that all the optimized solutions forms a Pareto front, where the for a given payload volume one can chose the optimized solution with maximum payload max. 30 optimized solutions are derived from the optimization process. Depending on the requirements of the mission and choice of the user, the best configuration can be chosen from this set of solutions. The optimized configurations are compared with the Raduga configuration. The Raduga configuration are results are derived from the open loop as described in Table 4-8, Table 4-9, Table 4-10 and Table 4-11.

Figure 4-10 gives the comparison of the values of the constraints considered for the optimization with respect to the reference values of Raduga configuration as mentioned in Table 4-9, Table 4-10 and Table 4-11 respectively. In this figure, the first solution is the Raduga configuration and other 30 solutions performance is described with respect to this reference configuration.

Figure 4-10 describes that all the optimized solutions not only satisfy the constraints but also shows better performance with respect to the Raduga reference configuration. The constraint over g-load is satisfied as well as the value is approximately equal to the reference configuration of the Raduga configuration. Furthermore, the optimized configuration satisfy the maximum heat flux constraint as well as the maximum value of the heat flux of the optimized configuration 10%

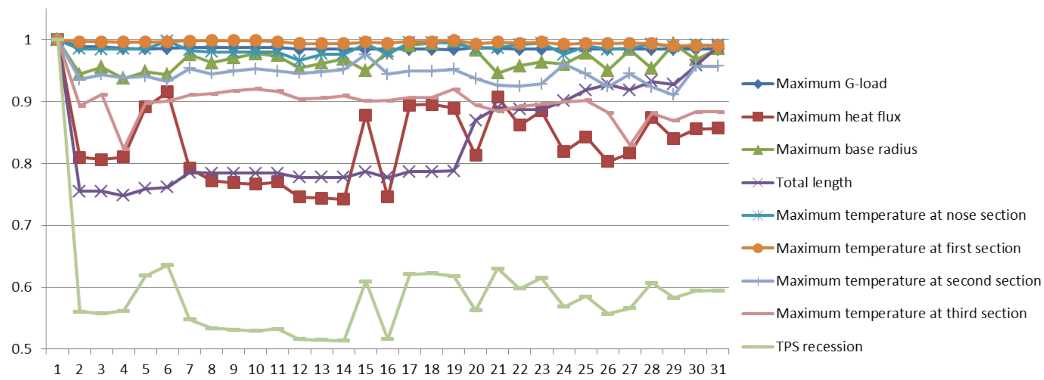



**Figure 4-10:** Constraints values comparison with reference configuration of Raduga

less than the reference configuration. As there is constraint on the maximum temperature at all the sections, to satisfy this constraint the maximum heat flux at the stagnation point is also reduced. From the sensitivity analysis, it is observed that there is a correlation between the heat flux and the maximum temperature at the cold structure. The maximum temperature at the cold structure for all sections is below 480 K as is maintained within the constraint for all the optimized configurations. The TPS recession at the nose section is well below the constraint, thus indicates that the available TPS thickness considering the recession still protects the cold structure of the re-entry vehicle. The total length and the maximum base radius also satisfy the constraints that are defined by the requirements of the launcher. Thus, all the optimized configurations are satisfying all the constraints and shows better performance.

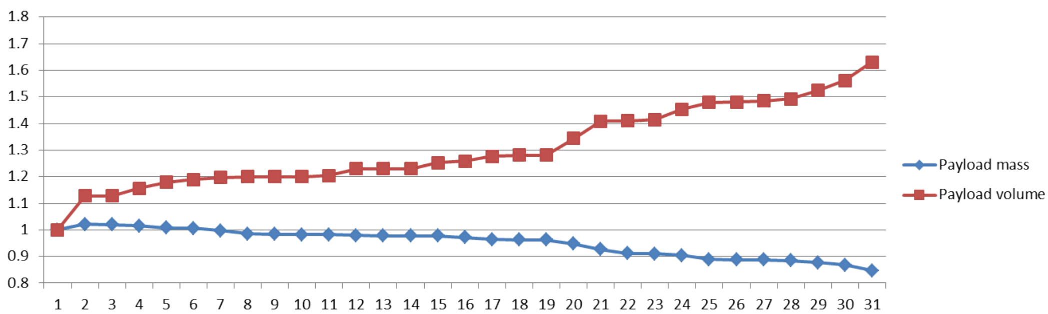


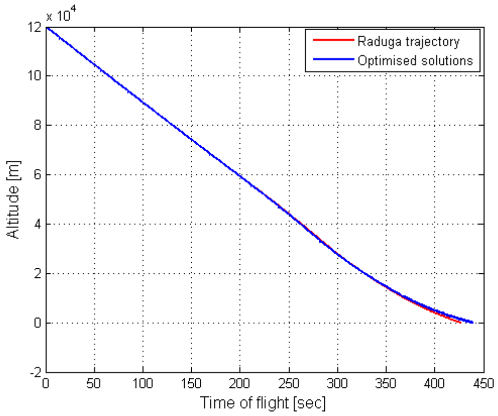

**Figure 4-11:** Objectives comparison with reference configuration of Raduga

Figure 4-11 gives the comparison of the payload mass and payload volume of optimized configurations with respect to the reference Raduga configuration, where the first configuration is the Raduga configuration and other 30 solutions are compared with this reference configuration. The reference values of the Raduga configuration are given in Table 4-8. The configuration with maximum payload mass can carry minimum payload volume due to the inverse co-relation between them as observed in the sensitivity analysis. The first 6 optimized configurations indicates higher payload mass and payload volume than the reference configuration. Thus, the results indicates that the MOGA optimization technique within the design process successfully derive a configuration with better performance than the reference configuration.

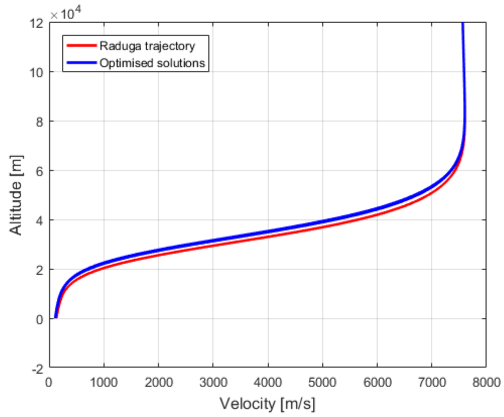
As all the configurations are axis-symmetric vehicles re-entering with zero angle of attack, they follow a ballistic trajectory. During the ballistic trajectory, there is no lift acting on the vehicle, so for an atmospheric flight, the trajectory is governed by drag and gravity only. The trajectory depends on the initial entry conditions and the ballistic parameter. The ballistic parameter is defined as $K=\frac{W}{C_D S}$, where $W$ is weight of the vehicle, $C_D$ is the drag co-efficient and $S$ is the surface area. Table 4-19 gives the entry conditions considered for the trajectory simulation. For the optimization the entry conditions and the total mass is considered as constant and thus, the trajectory only depends on the $C_D$ and the surface area variation.

**Table 4-19:** Entry conditions for the trajectory simulator

| | | |
|---|---|---|
| Altitude ($h_E$) | 120000 | [m] |
| Velocity ($V_E$) | 7563.71 | [m/s] |
| Flight-path angle($\gamma_E$) | -2.35 | [deg] |
| Heading angle ($\chi_E$) | 96.68 | [deg] |
| Latitude ($\delta_E$) | 51.45 | [deg] |
| Longitude ($\tau_E$) | 46.82 | [deg] |

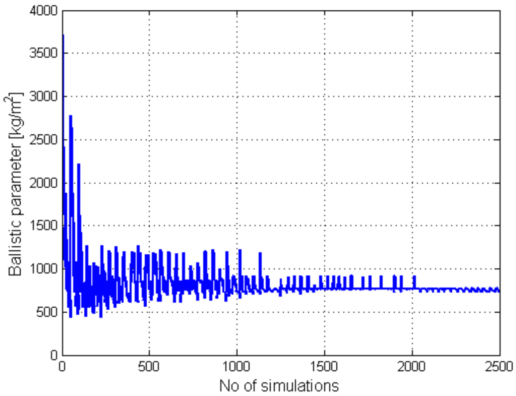



**(a)** Altitude profile

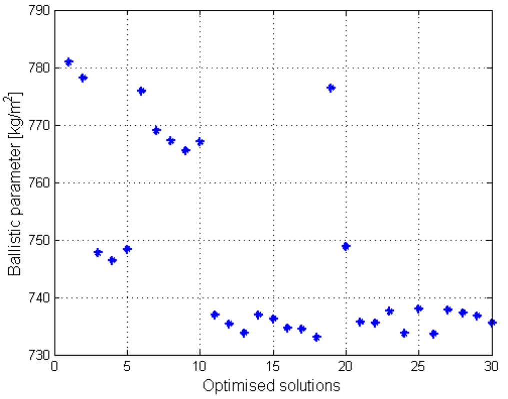



**(b)** Altitude-velocity profile

**Figure 4-12:** Comparison of the altitude and velocity profile with the Raduga configuration trajectory




**(a)** Ballistic parameter

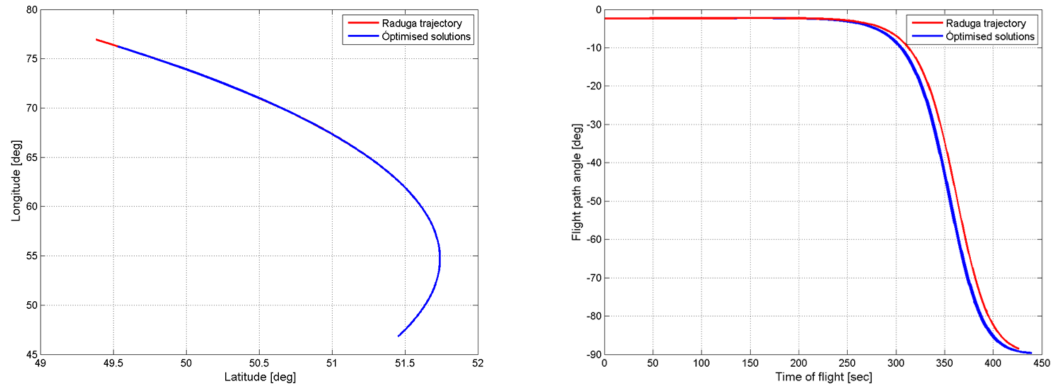



**(b)** Ballistic parameter of optimised solutions

**Figure 4-13:** Ballistic parameter variation

Figure 4-12a and Figure 4-12b gives the altitude and altitude-velocity profile of the optimized solutions and their comparison with the Raduga configuration. For the initial phase, where the largest mechanical and thermal loads occur and velocity is large, the flight-path angle can be assumed as constant thus Figure 4-12a gives a rectilinear path during the initial segment of the descent.

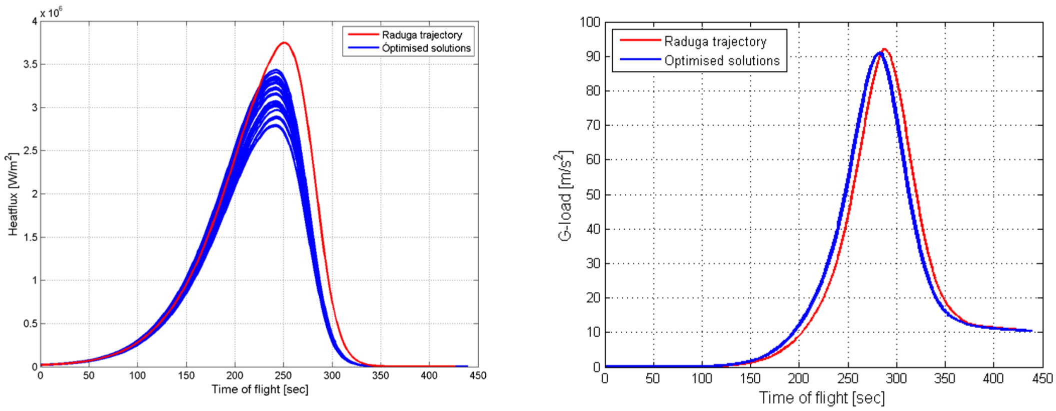
Figure 4-13 gives the ballistic parameter variation, where it can be observed that $K$ for the optimized configuration is between 733 to 781 kg/m$^2$. Figure 4-14b gives the flight-path angle profile. It is observed that the flight path angle is constant for first 300 seconds. For this duration a rectilinear profile is followed by the capsule. Furthermore, the flight-path increases rapidly as the capsule approaches the surface. The change of $\frac{V}{V_E}$ with altitude is dependent on the ballistic parameter $K$ and the entry flight-path angle $\gamma_E$. As the value of $K$ and $\gamma_E$ gets larger the vehicle will penetrate deeper into the atmosphere before the vehicles velocity is reduced significantly. Here the entry velocity is considered as fixed, thus the variation in vehicle's velocity is observed with the variation in the $K$.




**(a)** Latitude longitude profile

**(b)** Flight-path angle profile

**Figure 4-14:** Comparison of the velocity and flight-path angle profile with the Raduga configuration trajectory



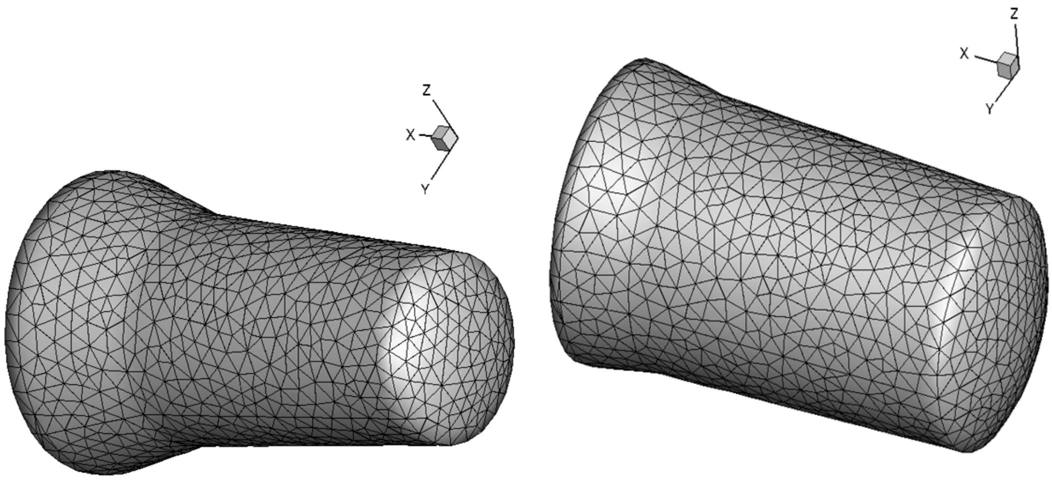

**(a)** Heat flux profile

**(b)** G-load profile

**Figure 4-15:** Comparison of the heat flux and G-load profile with the Raduga configuration trajectory

Figure 4-14a gives the latitude and longitude profile comparison of all the optimized configurations with the Raduga configuration. Figure 4-15a and Figure 4-15b gives the heat flux and the g-load comparison of the optimized solutions with the Raduga configuration. It is observed that the peak of the heat flux occurs before the peak of the g-load, thus the altitude at which maximum heat flux occurs at the stagnation point is larger than the altitude for max deceleration. The maximum deceleration changes with the initial conditions of entry velocity ($V_E$) and entry flight-path angle ($\gamma_E$) but is independent of the ballistic parameter (K). For all the configurations the entry conditions are same, thus maximum deceleration is same for all the configuration. The altitude at which maximum deceleration occurs is dependent on the ballistic parameter. Thus the altitude at which the maximum deceleration changes with the difference in the ballistic parameter.

From the trajectory results, it is observed that all the configurations gives better performance with respect to the reference re-entry vehicle. From the above 30 optimized configurations, based on the payload mass and the payload volume, the best configuration are selected. If the payload mass is maximum the payload volume capacity is minimum. Thus, 3 solutions are selected from the 30 optimized solution namely; configuration with maximum payload mass, configuration with maximum payload volume and one intermediate configuration.




**(a)** Original Raduga

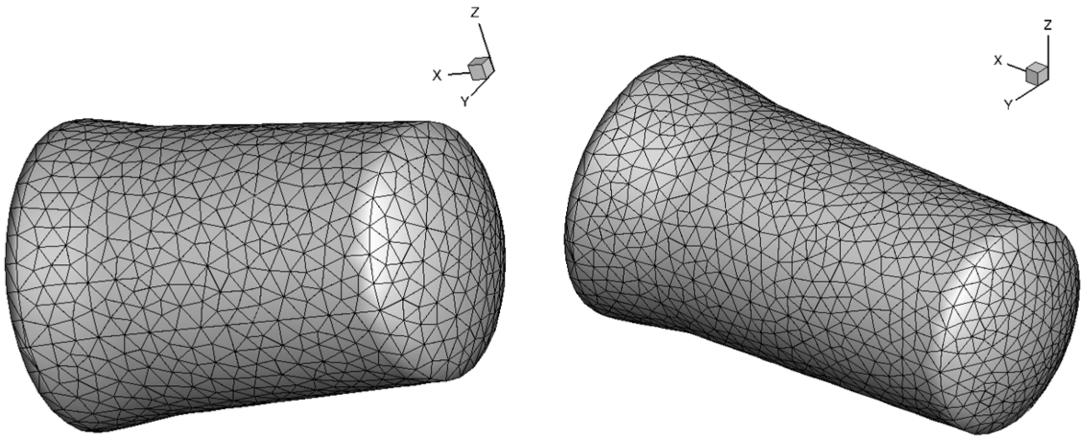
**(b)** Intermediate configuration

**Figure 4-16:** Comparison of the original Raduga configuration with the intermediate optimized configuration

Figure 4-16a and Figure 4-16b comparison of the original Raduga configuration with the intermediate optimized configuration. The intermediate configuration can carry 135 kg payload mass with 0.17 $m^3$ payload volume capacity, whereas the original configuration can carry 150 kg of payload mass with 0.11 $m^3$ payload volume capacity.

Figure 4-17a and Figure 4-17b gives the comparison of the configuration with maximum payload mass and configuration with maximum payload volume respectively. The maximum payload mass configuration can carry 153.2 kg payload mass with 0.14 $m^3$ payload volume capacity, whereas the configuration with maximum payload volume can carry 127 kg of payload mass with 0.20 $m^3$ payload volume capacity.

Based on the choice of the user, the best configuration can be selected out of these three. Considering the payload mass as first choice to select the configuration considering a payload volume at least greater than the reference configuration, the configuration with maximum payload mass is considered as the best choice from the optimized solution. Table 4-20 gives the comparison of all




**(a)** Maximum payload mass **(b)** Maximum payload volume

**Figure 4-17:** Comparison of the heat flux and G-load profile with the Raduga configuration trajectory

the values of the objectives and the constraints of the selected optimized configuration with the Raduga configuration.

**Table 4-20:** Comparision of the selected optimized solution with the Raduga Configuration

| Configuration | Original Raduga | Maximum payload mass | Unit |
|---|---|---|---|
| Payload mass | 150 | 154 | [kg] |
| Payload volume | 0.11 | 0.14 | [$m^3$] |
| Maximum base radius | 0.39 | 0.36 | [m] |
| Total length | 1.40 | 1.05 | [m] |
| Maximum heat flux at stagnation point | 3.74 | 3.03 | [$MW/m^2$] |
| Maximum g-load | 9.2 g | 9.1 g | [$m/s^2$] |
| Maximum temperature at cold structure: | | | |
| Nose section | 479 | 472 | [K] |
| First section | 478 | 478 | [K] |
| Second section | 478 | 449 | [K] |
| Third section | 478 | 428 | [K] |

From Table 4-20, gives the comparison of the selected optimized configuration with the reference Raduga configuration. It can be seen that, the derived solution through the design process gives the better solution than the reference Raduga configuration.

Dakota framework is an open source design environment and was not used yet for the re-entry vehicle application. Re-entry vehicle design is a complex system and to perform the MDO of the re-entry vehicle, it is required to integrate all the design disciplines on a single platform. The design process developed within the Dakota framework as described in Chapter 2 allowed to integrated all the disciplines of re-entry vehicle on a single platform successfully. The generalized templates and a common function to read and write the data to automate the design process, allowed to integrate all the tools irrespective of the programming languages used to develop them. This workflow included tools that are developed using Matlab, Simulink and Fortran. Additionally, the workflow also included a commercial software called Gridgen for mesh generation. The complete design process is automated as well as this design process allowed the developer to keep their software as a complete black-box. Furthermore, the utilization of the available MDO techniques within Dakota framework, such as the MOGA and LHS assisted the user to derive the optimized configuration of Raduga like capsule, using a low cost design process.

Chapter 5

# Verification of the Design Process for the Trajectory Optimization of a Lifting Body

This chapter describes the verification of the design process for the trajectory optimization of a lifting body re-entry vehicle. As seen previously, the design process developed within the Dakota framework successfully derived an optimized configuration of an axis-symmetric Raduga-like capsule. The axis-symmetric capsule followed a ballistic trajectory and thus, the trajectory of the vehicle is coupled with the geometric parameters of the vehicle, whereas for the lifting body, the steering of the vehicle plays an important role to define the trajectory. Thus, for the trajectory optimization, the bank angle and the corresponding time are also considered as the design parameters, whereas the angle-of-attack is considered as constant.

To answer the main research question, it is important to verify the design process for the trajectory optimization of a lifting body, where the bank angle maneuvering can be performed to guide the vehicle during the re-entry phase. Since all the necessary data of the IXV configuration is available at TASI, it is considered as a reference re-entry vehicle. The nominal trajectory of the IXV configuration will be compared with the optimized trajectories of IXV derived by using the design process. Section 5-1 gives the workflow for the trajectory optimization and the open loop simulation for the interface verification. Furthermore, Section 5-2 describes the trajectory optimization of the IXV configuration using the developed design process and the comparison of the optimized trajectories with the nominal trajectory of the reference vehicle, the IXV.

## 5-1 Workflow and Interface Verfication

As described in the Chapter 2, the developed design process within the Dakota framework can easily plug and play different tools and software in the tool-chain. Thus, the workflow created for the non-lifting body re-entry vehicle (Raduga) optimization can be easily modified by removing/replacing tools and interfaces that are necessary for the trajectory optimization of a lifting body re-entry vehicle. This section describes the workflow created for the trajectory optimization and verification of the interfaces between the tools and between the Dakota optimizer and the toolchain of the workflow.

The verification of the design process for the trajectory optimization of a lifting body is performed for the given geometry and the aerodynamic database of reference vehicle, the IXV as described by Santilli and Sudars (2014). Since the geometry and aerodynamic database is fixed, the geometry generation and the aerodynamic database computation tools as described in Figure 4-1 are not required for the verification of the design process for the trajectory optimization. Thus, for this workflow, where the verification of the design process is performed for a given geometry and the aerodynamic database, only two tools namely: trajectory simulator and the modified CoRADS are used. Figure 5-1 gives the workflow for the trajectory optimization of the IXV configuration. The description and the unit test verification of these tools were discussed in Chapter 3. Compared to the workflow of the non-lifting body optimization, here the interface verification is performed again as the parameters flowing from the interfaces are different.

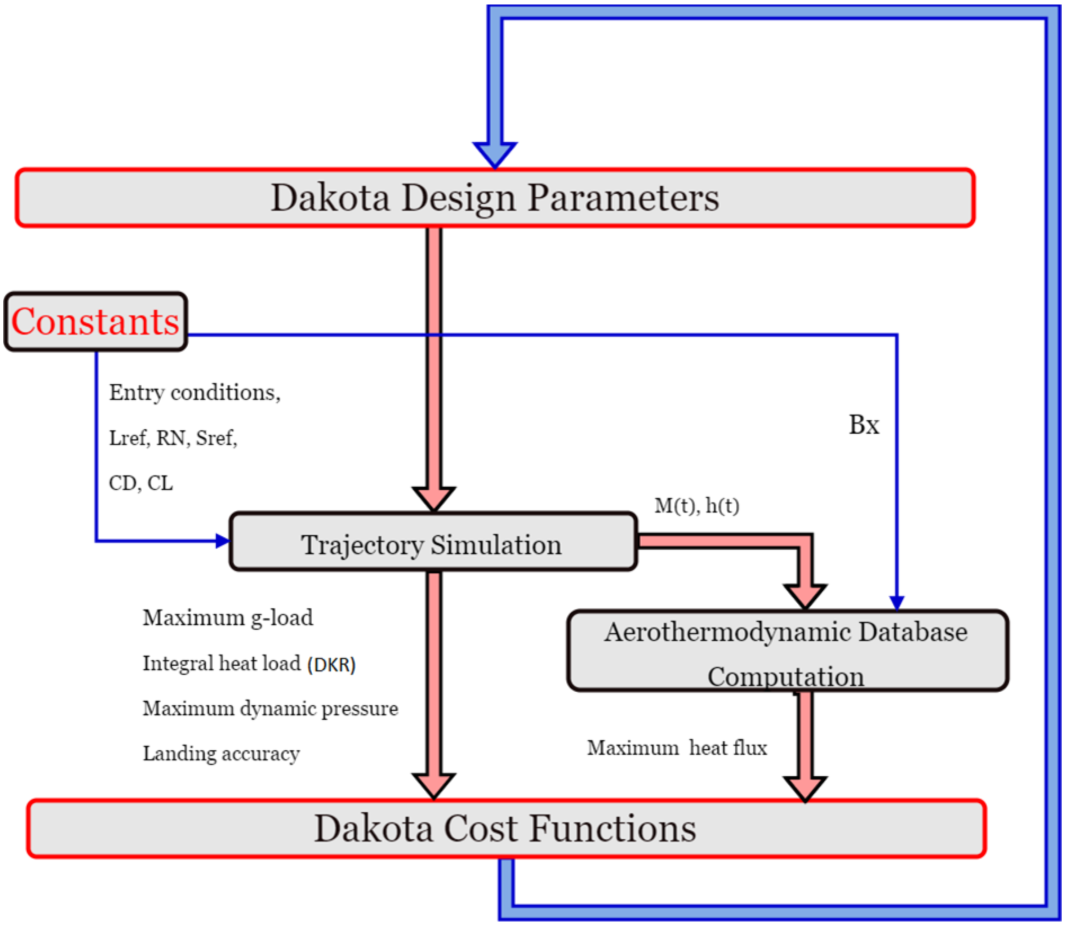



**Figure 5-1:** Workflow for trajectory optimization of IXV

As shown in Figure 5-1, the design parameters flow from the Dakota design parameters file to the input template of the trajectory simulator. The design parameters are the absolute value of the bank angle and the corresponding time intervals. As discussed in Section 3-1-2, Figure 3-15 describes these design parameters, that are considered as inputs for the trajectory simulator. For the unit test trajectory simulator verification as discussed in Section 3-1-2, 20 absolute values of bank angle and time intervals were given by the user and not by the Dakota optimizer, whereas for performing the trajectory optimization the values of design parameters are selected by the Dakota optimizer. It was observed that to compare the derived optimized trajectory with the reference trajectory, Dakota should select the design parameters (absolute value of bank angle and time interval) which give a similar bank angle profile as that of reference IXV bank angle profile. Thus, to compare the results with the nominal trajectory, 20 absolute values of bank angles and the corresponding time intervals are maintained for the trajectory optimization. The initial absolute bank angle and the corresponding time interval is considered to be 0 deg and 200 sec respectively. Table 3-25 gives the other 19 bank angles and time intervals, that flows from the Dakota parameter file to the trajectory simulator input template. As the interface verification, it is observed that exactly same values are flowing from the Dakota parameter file to the trajectory simulator input template as shown in Figure 2-4.

Additionally, the objective of this verification is to implement the same workflow to answer the main research question. The main research question deals with the shape optimization as well as with the trajectory optimization of the IXV like configuration. Thus, to perform the shape and the trajectory optimization, the workflow for geometry optimization and the trajectory optimization will be clubbed together. To keep the flexibility to utilize the same workflow of the trajectory optimization for the design case, an additional constants block is introduced between the geometry and the trajectory workflow. This gives the flexibility to perform only trajectory optimization of a fixed configuration, while keeping the geometry and its aerodynamic data set as constant during the optimization. The constants such as the entry conditions, target landing location, the radius of a nose, surface area, total length and the total mass considered for the trajectory simulation flow from the constants block to the input template of the trajectory simulator. Table 3-24 gives these constants that are stored in the constant block as shown in Figure 5-1. As the interface verification, it is observed that exactly same values flow from the constant block to the trajectory simulator input template.

The trajectory simulations are performed for the given aerodynamic database and the geometry parameters of IXV. Santilli and Sudars (2014) describe the IXV geometry and the corresponding aerodynamic database. This available aerodynamic database is computed using the Computational Fluid Dynamics (CFD) computations, thus the simulations for the trajectory optimization of IXV is stopped at Mach=1.5. As described in Chapter 3, the trajectory simulator is responsible for computing the trajectory of the re-entry vehicle considering the lateral guidance logic, that considers the bank angle maneuvering to reach the desired location. Trajectory simulator also computes the g-load, dynamic pressure, landing accuracy, heat-flux and the corresponding integral of the heat load. In the trajectory simulator, the heat flux is computed by using the DKR method. The necessary outputs are extracted from the output file of the trajectory simulator to fill-up the template of the Dakota results file.

Once, the trajectory simulations are performed, the trajectory data is saved in the working directory, which is used by modified CoRADS as input. Aerothermodynamics database is computed using the modified CoRADS. This tool computes the heat flux at different locations by using the Fay-Riddle method, which is based on the boundary layer equations and the similarities transformations as described in Chapter 3.

From the trajectory results, the altitude and Mach number at which maximum heat flux occurs by DKR method are considered as input by the modified CoRADS tool. It is also considered that the trajectory simulator results of the respective computation are saved in the current working directory. Since the modified CoRADS compute the maximum heat flux value at the stagnation point without the variation of the heat flux with time. The heat flux profile with respect to time computed by the DKR method from the trajectory simulator is considered and is scaled accordingly to obtain the results from the modified CoRADS. Additionally, the modified CoRADS also requires the location at which the heat flux is computed and the reference length of the configuration. These values are stored in the constants block as given in Table 5-1. As an interface, these values flow from the constant block to the input template of the modified CoRADS.

This maximum heat flux is considered for the optimization, thus is collected in the Dakota result file as shown in Figure 5-1. The results from trajectory simulator and the modified CoRADS as the cost functions are then given back to the Dakota optimizer, which generates the new design parameters for the further simulation. Thus, there is an interface between the tools and the Dakota results file as described in the Figure 5-1. The Dakota results template includes the major cost

**Table 5-1:** Inputs for modified CoRADS

| | Constants block | Units |
|---|---|---|
| Location of stagnation point (Bx) | 4.51 | [m] |
| Reference length (Lref) | 5.05 | [m] |

functions which are considered for the trajectory optimization of the vehicle. These values are read by Dakota to generate new design parameters, thus it is also important to verify during the interface verification that values of the objective function derived from the trajectory simulator are transferred to the input template of the Dakota results template. Table 5-2 gives values of the objective function derived from trajectory simulator. Similarly, Table 5-3 gives the results from the modified CoRADS.

**Table 5-2:** Parameters in output file of trajectory simulator

| Parameters | Trajectory simulator output | Units |
|---|---|---|
| Maximum heat load | 282.88 | [MJ/m$^2$] |
| Landing accuracy | 173.17 | [km] |
| Maximum g-load | 2.01 g | [m/s$^2$] |
| Dynamic pressure | 5608.92 | [Pa] |
| Maximum longitude | 235.36 | [deg] |
| Change in flight-path angle | 0.037 | [deg] |

**Table 5-3:** Parameters in output file of modified CoRADS

| Parameters | Modified CoRADS | Units |
|---|---|---|
| Maximum heat flux | 0.46 | [MW/m$^2$] |

During the interface verification, it is seen that exactly same values are transferred from the trajectory and the modified CoRADS output file to the Dakota results template using the dprepro function. During the optimization, based on these results, new design parameters are generated by Dakota and the continues the cycle till a converged solution is obtained. This interface verification qualifies the workflow for the system level verification, where the workflow will be used for the trajectory optimization of the IXV configuration.

## 5-2 Optimisation Results

As described in Chapter 4, Table 4-15 gives the tunings for the MOGA technique considered for the Raduga configuration. Prior to the optimization of the Raduga configuration, several tests were performed to chose the settings for the MOGA. From the results of the Raduga configuration optimization, it is observed that the chosen tunings give a converged solution, thus along with the optimization technique these settings are also fixed and considered for the trajectory optimization.

This section describes the trajectory optimization results of the IXV configuration. Furthermore, the optimized trajectories derived from the design process are compared with the nominal trajectories of the reference lifting vehicle, IXV as described by Rodrigo et al. (2016). The initial bank angle is considered to be 0 deg for first 200 seconds. Table 3-24 gives the entry conditions for the trajectory optimization. As mentioned previously, to perform the trajectory optimization of the IXV configuration, the geometry and the aerodynamic database is considered as constant as mentioned by Santilli and Sudars (2014). The $S_{ref}$= 7.25 m$^2$, $RN$ = 1.05 m and the total mass = 1957 kg are the geometry parameters considered are trajectory optimization.

**Table 5-4:** Design parameters for the trajectory optimization

| Design parameters | Lower bound | Upper bound | Unit |
|---|---|---|---|
| Absolute values of bank angle | 0 | 90 | [deg] |
| Time interval | 5 | 300 | [sec] |

Table 5-4 gives the design parameters for the trajectory optimization. The absolute value of the bank angle varies between 0 to 90 deg and the time interval varies from 5 sec to 300 sec. To avoid the steep bank angle variation, the minimum time interval is considered as 5 seconds. During the optimization, it is also required to define the initial values of the design parameter to start to optimization. Thus, the bank angle and time interval of the nominal trajectory of IXV, are considered as initial values for the design parameters as described in Table 3-25.

Table 5-5 and Table 5-6 gives the design objectives and constraints considered for the trajectory optimization.

**Table 5-5:** Objectives for the optimization

| |
|---|
| Minimize integral heat load |
| Minimize distance from landing to target |

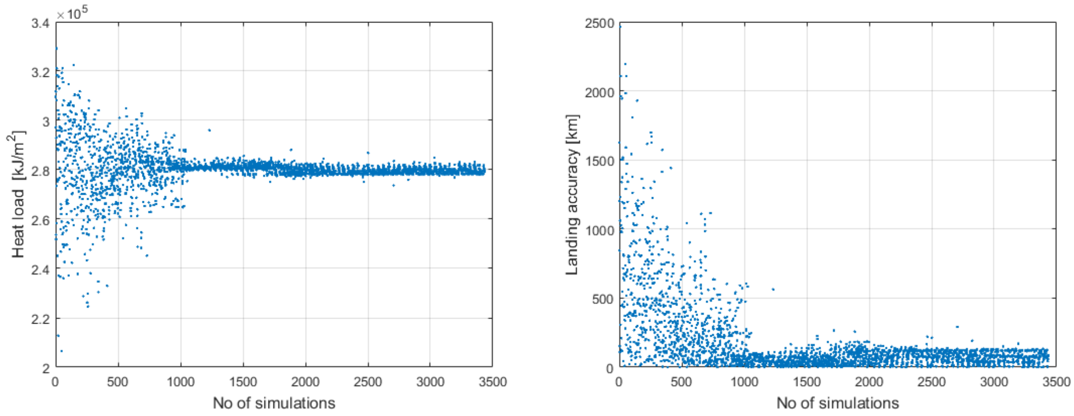
**Table 5-6:** Constraints for the optimization

| Constraints | Values | Unit |
|---|---|---|
| Maximum heat flux | $\leq$ 550 | [KW/m$^2$] |
| G-load | $\leq$ 3 | [g] |
| Dynamic pressure | $\leq$ 5000 | [Pa] |
| Maximum longitude | $\leq$ 237 | [deg] |
| Change in flight path angle | $\leq$ 0.15 | [deg] |

Considering the these objectives and constraints described in Table 5-5 and Table 5-6 a converged solution for the trajectory optimization is derived using the MOGA. Figures 5-2a and 5-2b gives the performance of the optimizer, which indicates the convergence of the objective functions: heat-load and the landing accuracy. As this is a multi-objective problem, the solution gives a band of converged solutions, the algorithm stops as soon as the changes occurring over a user specified number of generations fall below a user specified threshold. Here, the percentage change is 0.1 and 5 generations are considered as the convergence criteria, as described in Table 4-15.

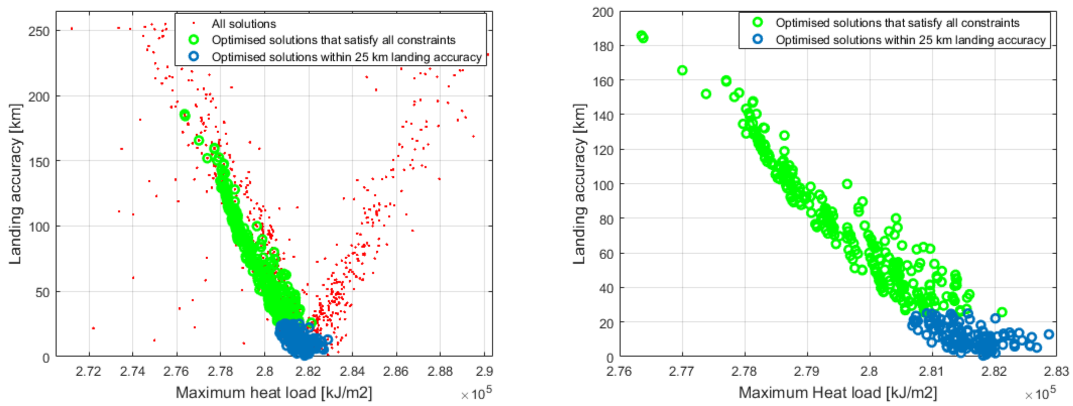
Once the optimization results are derived, the solutions with landing accuracy below 25 km are selected to compare the results with the nominal trajectory of IXV. Figure 6-21 gives the Pareto front between the landing accuracy and the maximum heat load.

Figure 5-3a gives the scatter plots, which describes the all the simulated solutions (red), solutions that satisfy all the constraints (green), and furthermore solutions that satisfy all constraints as well as within 25 km landing accuracy (blue). There are 258 solutions that are optimized trajectories within 25 km. The discussion of the results is narrowed down to these optimized trajectories within 25 km. Table 5-8 describes the range of the values of the constraints and objectives derived




**(a)** Convergence of Heat-load **(b)** Convergence of Landing accuracy

**Figure 5-2:** Optimiser Performance




**(a)** Scatter plot **(b)** Scatter plot

**Figure 5-3:** Scatter plot between landing accuracy and the maximum heat load

from the optimization and their comparison with the values of the nominal trajectory of the IXV configuration as described by Rodrigo et al. (2016).

**Table 5-7:** Output of optimized trajectories

| Objective and constraints | Minimum value | Maximum value | Nominal value | Unit |
|---|---|---|---|---|
| Integral heat load | 280 | 282 | 287 | [MJ/m$^2$] |
| Maximum heat flux | 541 | 549 | 555 | [KW/m$^2$] |
| G-load | 1.48 | 1.84 | 1.96 | [g] |
| Dynamic pressure | 3900 | 4800 | 5059 | [Pa] |
| Landing accuracy | 0.5 | 24.5 | 4.9 | [km] |

From Table 5-7 it is observed that the all the solutions not only satisfied all given constraints but also showed a better performance than the nominal trajectory of IXV for the considered systems

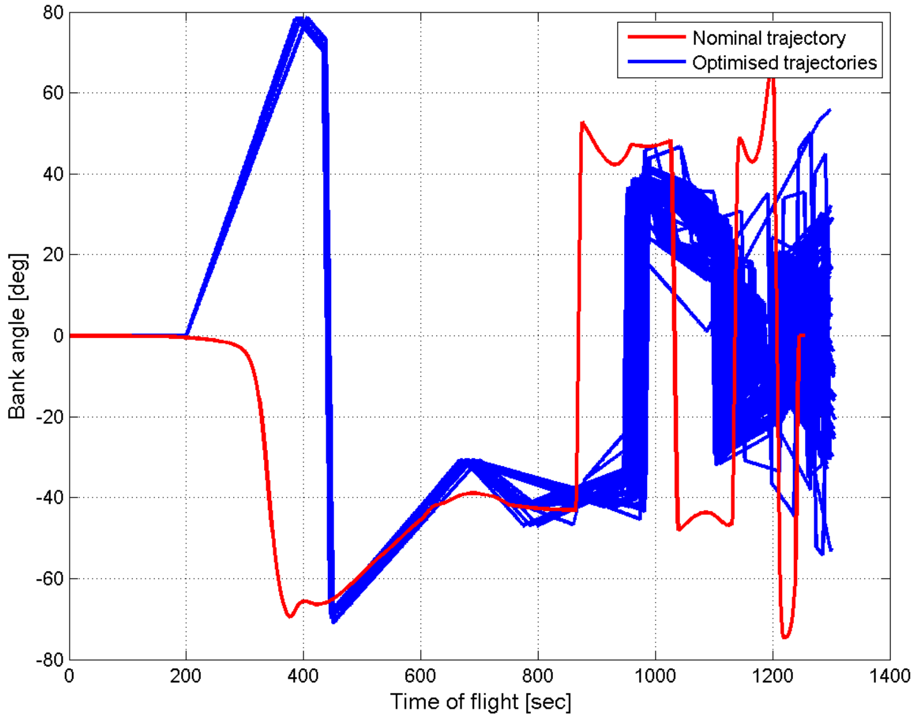
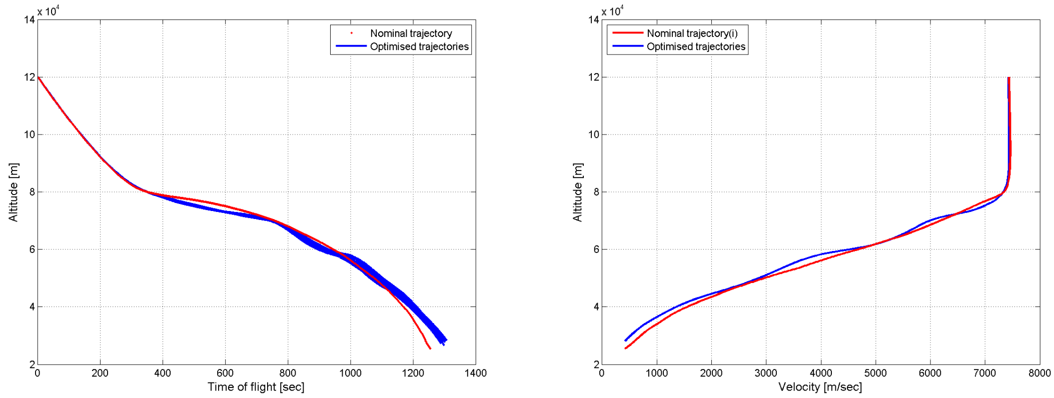



**Figure 5-4:** Comparison of the bank angle profile of the optimized trajectories with the nominal trajectory of IXV

requirements. Figure 5-4 gives the comparison of the bank-angle profile of the optimized trajectory with the nominal trajectory of IXV. During the entry flight, the angle-of-attack is assumed to be at 45 deg in hypersonic phase to ensure good aero-thermodynamics performance. The Dakota optimizer selects the absolute value of bank-angle and corresponding time interval, thus the value of second bank angle is opposite in signs. The absolute value of the bank-angle and the time interval is defined as shown in Figure 3-15.

The trajectory simulations are performed till 1.5 Mach as the reference aerodynamic database computed by Navier stokes equations is available for the verification of the design process for the trajectory optimization of IXV. The optimized trajectory bank angle profile shows a steeper maneuver at the beginning of the entry and performs its first bank reversal within first 500 seconds of the re-entry, whereas the nominal trajectory performs the first bank reversal after 850 seconds.

This early bank reversal results in a deviation between the altitude profile as shown in Figure 5-5a. The absolute value of the bank angle is considered as input for the trajectory simulator and based on the heading angle error the lateral guidance logic performs the bank maneuvering. The lateral guidance is implemented as discussed previously in Section 3-1-2. Thus, if the absolute value of the bank angle is compared at 400 seconds, the optimized trajectories indicate higher bank angle as compared to the absolute value of bank angle of nominal trajectory. Therefore, the magnitude of the vertical lift vector for the optimized trajectories is lower than the nominal trajectory. This leads to a difference in the altitude profile, where one can easily see the difference between the altitudes of the optimized trajectory and the nominal one. The higher the magnitude of vertical lift higher is the altitude and thus nominal trajectory shows higher altitude than the optimized trajectories from 400 seconds to 800 seconds approximately. Furthermore, for the similar reasons the magnitude of the vertical lift for the nominal trajectory is lower than the optimized trajectory during the end phase thus, optimized trajectories show a slightly higher altitude than the nominal

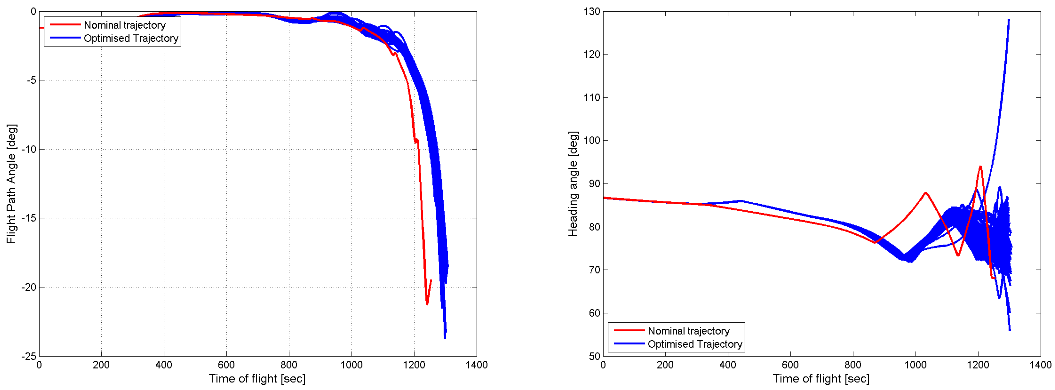



**(a)** Altitude profile **(b)** Altitude-velocity profile

**Figure 5-5:** Comparison of the altitude and velocity profile of the optimised trajectories with the nominal trajectory of IXV

one.

From Figures 5-5a and 5-5b, it is seen that all the altitude and altitude-velocity profiles of the optimized solutions match with the nominal trajectory of IXV. The difference in the absolute value of the bank angle is also depicted through Figure 5-5b, which gives the comparison of the altitude and velocity profile of the IXV trajectory.

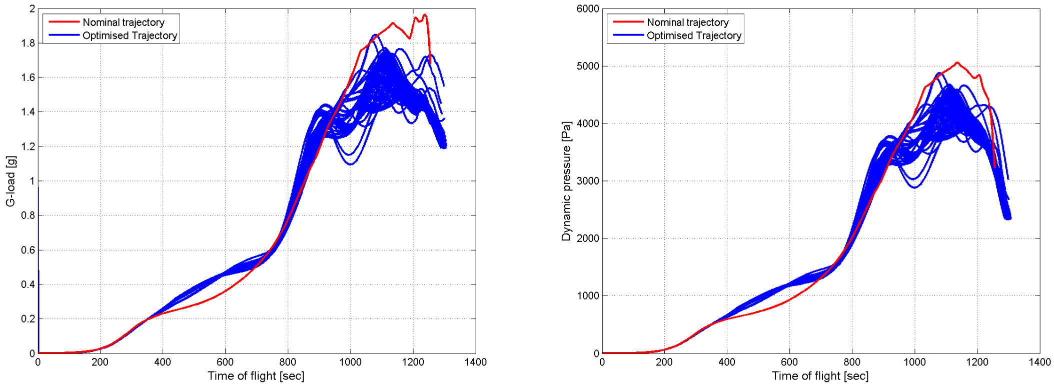



**(a)** Flight-path angle profile **(b)** Heading angle profile

**Figure 5-6:** Comparison of the flight-path angle and the heading angle profile of the optimized trajectories with the nominal trajectory of IXV

Figure 5-6a and Figure 5-6b gives the comparison of flight-path angle and the heading angle profile of optimized trajectories and nominal trajectory of IXV. The flight-path angle profile match with the nominal trajectory and the slight deviation are explained by the bank angle maneuverings. Similarly, the bank angle maneuverings is performed by the error in the heading angle. Thus, the difference in the bank angle profile of the optimized trajectories and the nominal trajectory of the IXV is also observed in the heading angle profile.

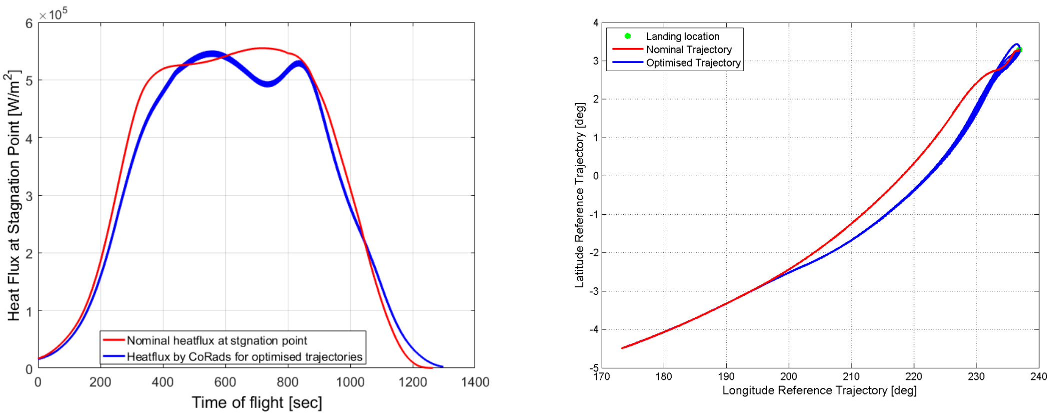
Figure 5-7a describes the comparison of the g-load profile of the optimized trajectories with the




**(a)** G-load profile **(b)** Dynamic pressure profile

**Figure 5-7:** Comparison of the G-load and the dynamic pressure optimized trajectories with the nominal trajectory of IXV

nominal trajectory of the IXV configuration. By comparing the peak of the g-load, it is observed that all the optimized trajectories give lower peak of g-load, as compared to the nominal trajectory of the IXV. Furthermore, Figure 5-7b describes the comparison of the dynamic pressure profile of the optimized trajectory, where it is observed that the peak of the dynamic pressure is lower than the peak of the nominal trajectory. Thus, as given in Table 5-7, in-terms of g-load and maximum dynamic pressures optimized trajectories gives better performance than the nominal trajectory.




**(a)** Heat flux at stagnation point **(b)** Latitude-longitude profile

**Figure 5-8:** Comparison of the maximum heat flux and the latitude and longitude of the optimized trajectories with the nominal trajectory of IXV

Figure 5-8a gives the maximum heat flux at the stagnation point. It is seen from the figure that the peak of the heat flux is also lower than the nominal trajectory. Thus, as given in Table 5-7 the maximum heat flux the optimized trajectories gives better performance than the nominal trajectory. From Figure 5-8a and Figure 5-7a, it is observed that the peak of the heat flux occurs the peak of g-load thus, all the optimized trajectories are a valid solution. Figure 5-8b gives the comparison of the latitude-longitude profile of the optimized trajectories with the nominal trajectory of IXV. As given in Table 5-7, the landing accuracy for all the optimized trajectory vary between 0.5 km to 24.5 km thus, reach the target landing location. It is also observed that there are trajectories that give a landing accuracy less than 4 km, and thus a better landing accuracy than the nominal trajectory.

From the results of the trajectory optimization, it is observed that for all the given objectives and constraints, the optimized trajectories gives better performance than the IXV configuration. Thus, the design process is verified for the trajectory optimization of the lifting body re-entry vehicle. Furthermore, Section 5-3 describes the selected best-optimized trajectory.

## 5-3 Selected Best-optimized Trajectory

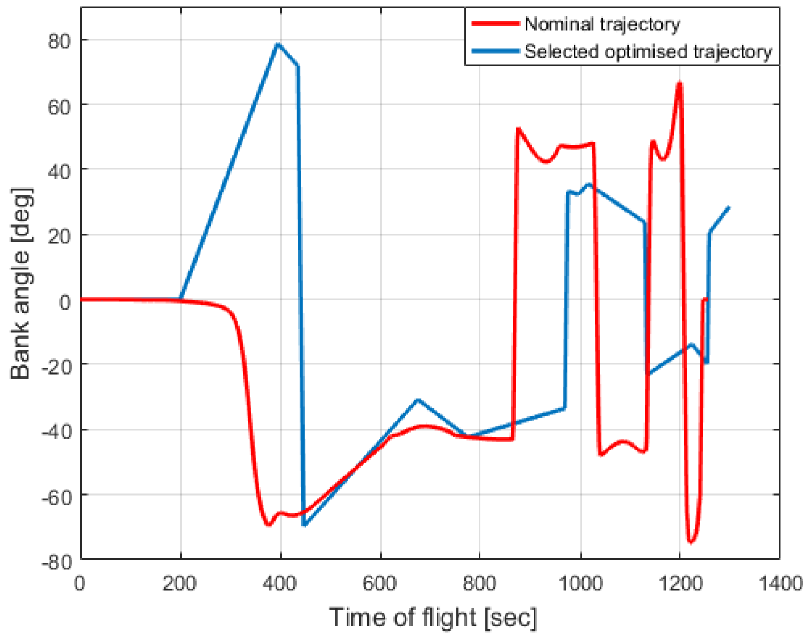


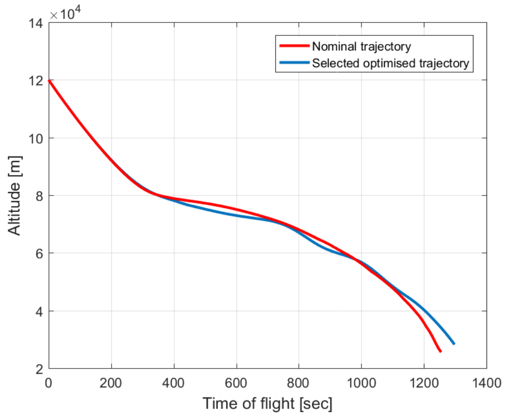

**Figure 5-9:** Comparison of the bank angle profile of the selected trajectory with the nominal trajectory of IXV

As all the optimized trajectories give better performance than the nominal trajectory, it is also narrowed down to select the best trajectory, that reaches the target with maximum landing accuracy. Table 5-8 compares the results of the optimized trajectory with landing accuracy of 0.5 km with the nominal trajectory.

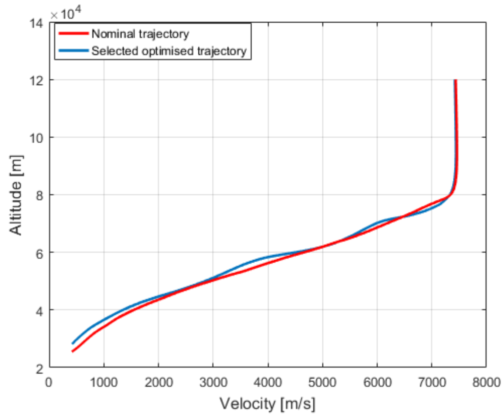
**Table 5-8:** Comparison of the selected trajectory with nominal trajectory of IXV

| | Selected trajectory | Nominal trajectory | Unit |
|---|---|---|---|
| Integral heat load | 281.81 | 287.16 | [MJ/m$^2$] |
| Landing accuracy | 0.5 | 4.9 | [km] |
| Maximum heat flux | 545.69 | 554.90 | [KW/m$^2$] |
| G-load | 1.67 | 1.96 | [g] |
| Dynamic pressure | 4414 | 5059 | [Pa] |

Figure 5-9 gives the bank angle profile of the selected trajectory and its comparison with the nominal trajectory of IXV. As discussed in the previous section, the bank angle profile depends on the lateral guidance within the trajectory simulator which computes the heading error and derives the required bank maneuver to satisfy all the constraints. From the figure, it is also seen that the bank angle at end of the trajectory reaches to 25 deg, whereas the nominal trajectory ends with zero bank angle. As no constraint is applied for the last bank angle during the optimization the Dakota optimizer choose the best possible design parameters such that it meets the constraints.

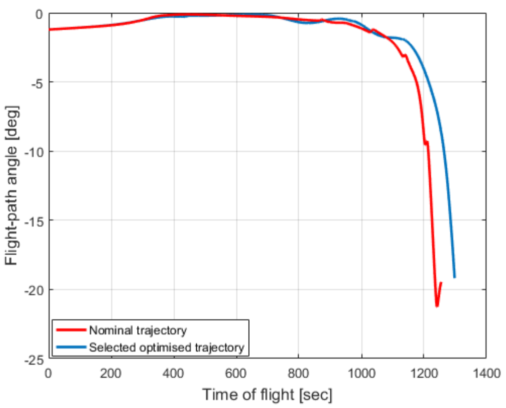



**(a)** Altitude profile

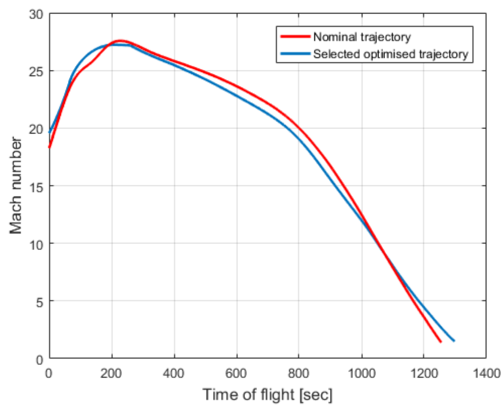



**(b)** Altitude-velocity profile

**Figure 5-10:** Comparison of the altitude and velocity profile of the optimised trajectories with the nominal trajectory of IXV



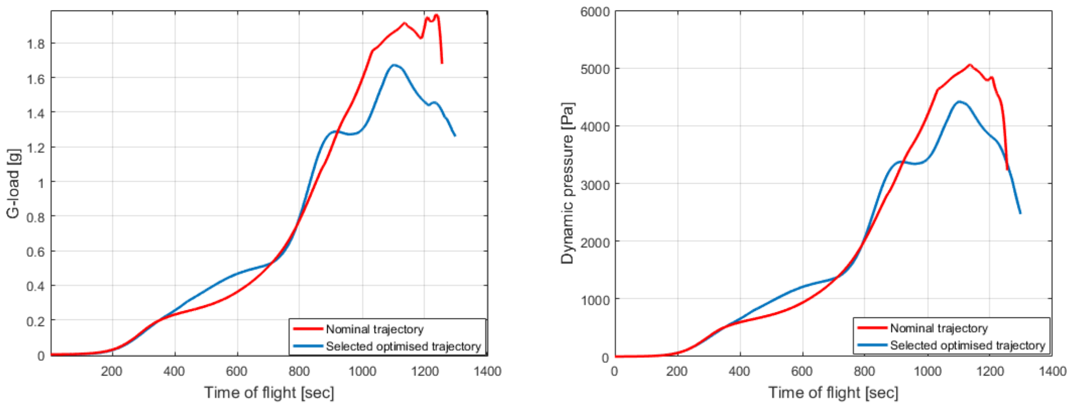

**(a)** Flight-path angle profile




**(b)** Mach number profile

**Figure 5-11:** Comparison of the flight-path angle and the Mach number profile of the optimized trajectories with the nominal trajectory of IXV

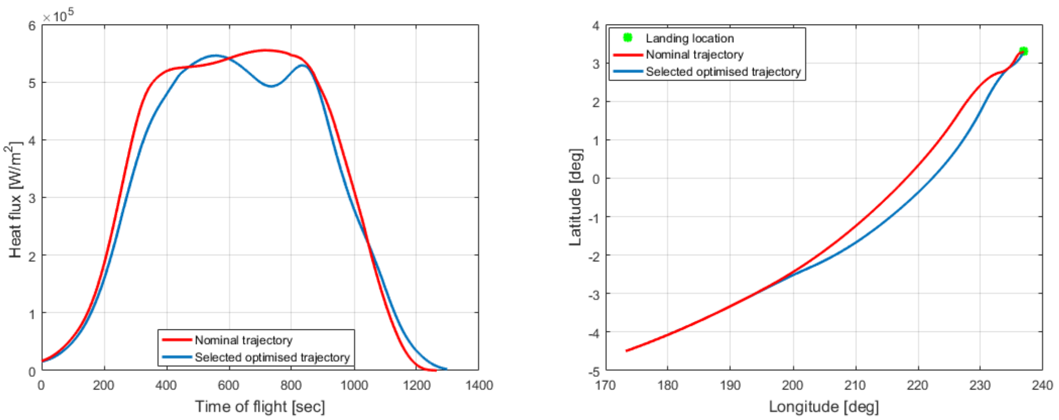
Figure 5-10a and Figure 5-10b gives the altitude and altitude-velocity profile of the selected optimized trajectory and its comparison with the nominal trajectory of IXV. Figure 5-11a and Figure 5-11b gives the flight-path angle profile and Mach number profile of the selected optimized trajectory and its comparison with the nominal trajectory of IXV. As previously described, the difference in the bank angle profile leads to a difference in the vertical lift vector and thereby the altitude and the corresponding velocity. From this comparison, it can be seen that, apart from the bank angle profile difference, the selected optimized trajectory perfectly match with the nominal trajectory of IXV configuration.




**(a)** G-load profile

**(b)** Dynamic pressure profile

**Figure 5-12:** Comparison of the G-load and the dynamic pressure optimized trajectories with the nominal trajectory of IXV




**(a)** Heat flux profile

**(b)** Longitude-latitude profile

**Figure 5-13:** Comparison of the flight-path angle and the heading angle profile of the optimized trajectories with the nominal trajectory of IXV

Figure 5-12a and Figure 5-12b gives the g-load profile and dynamic pressure profile of the selected optimized trajectory and its comparison with the nominal trajectory of IXV. From this comparison, it is observed that the maximum g-load of the selected optimize trajectory is 1.67g $\mathrm{m/s^2}$ whereas, the g-load of the nominal trajectory is 1.96g $\mathrm{m/s^2}$. Furthermore, for the selected optimized trajectory, the peak of the dynamic pressure is 4414 Pa whereas, for the nominal trajectory the peak is 5059 Pa. Thus, there is noticeable reduction in the peak g-load and the dynamic pressure by using this design process for the trajectory optimization. Figure 5-13a and Figure 5-13b gives the heat flux profile and longitude-latitude profile of the selected optimized trajectory and its comparison with the nominal trajectory of IXV.

From the comparison, it is observed that the maximum heat flux for the selected trajectory is 545.69 $\mathrm{KW/m^2}$ whereas, for the nominal trajectory it is 554.90 $\mathrm{KW/m^2}$. Thus, the maximum heat flux is also reduced significantly through optimization using the developed design process. From the landing accuracy, it is observed that the selected trajectory gives maximum landing accuracy of

0.5 km whereas the nominal trajectory reached the target location with 4.9 km landing accuracy. From the selected optimized trajectory, it is verified that for the right choices of the objectives and constraints, one can derive the optimized trajectory of a complex re-entry vehicle using the developed design process within the Dakota framework. Thus, the workflow can be considered further to answer the main research question, where the shape optimization is performed along with the trajectory optimization.

# Chapter 6

# Results: Design Case

The main research question deals with deriving an optimized re-entry vehicle similar to the IXV configuration with payload carrying capacity using the developed design process within the Dakota framework. This design process is verified individually for the optimization of the axis-symmetric Raduga capsule as described in Chapter 4 and for the trajectory optimization of the IXV configuration as described in Chapter 5. After this verification, it is made sure that the design process developed within the Dakota framework is capable of deriving an optimized configuration of Raduga as well as the trajectory of a given complex configuration like IXV.

The design process is developed in such a way that a new tool can be easily plugged and played into the toolchain using the generalized interface templates as described in Chapter 2. To answer the main research question, it required to generate a complex geometry re-entry vehicle similar to the IXV and also compute the payload mass capacity of the vehicle. As described in Chapter 3, the SMMET is developed to derive the design case through the shape morphing of the original mesh of the IXV as well as to compute the payload carrying capacity of the derived configuration. Using the same integration methodology developed with the Dakota framework, a separate workflow is created for the design case, where SMMET, CoRADS, trajectory simulator, modified CoRADS are integrated together. This workflow is created from the previous workflows, that were used for the optimization of the Raduga-like capsule and for the trajectory optimization of the IXV.

The unit test of all the tools involved in the design process is described in Chapter 3. This section describes the workflow and the verification of the interfaces between all the tools and between the Dakota optimizer and the toolchain of the workflow described in Figure 6-1. Section 6-1 gives the requirement of a new workflow for the design case and highlight the differences between the workflow for design case and previously used workflows. Section 6-2 describes the workflow created for the design case feasibility study, which includes the geometry as well as the trajectory optimization. Furthermore, it gives the interface verification of the workflow. The interface verification describes the flow of the information within the tool-chain and between the Dakota optimizer and the tool-chain. After these verifications starting from unit test of the tools to the verification of the design process, qualifies the design methodology to answer the main research question. Section 6-3 describes the derived results, where the design process developed within the Dakota framework is used to derive an optimized configuration similar to the IXV that is capable of carrying a payload from the ISS to the Earth.

## 6-1 Comparison of Workflows

This section highlights the differences between the workflows created for the non-lifting body configuration and for the IXV-like. Although separate workflows are created for the lifting body and the non-lifting body feasibility study, they are created using the same integration methodology within the Dakota framework. As the tool created for the non-lifting body geometry generation (GGMET) cannot be used to create a complex lifting body vehicle similar to IXV, there is a requirement to replace this tool. Thus, SMMET as the new tool is created to morph the baseline configuration of IXV and derive new configurations. Furthermore, this tool can compute the payload capacity of the vehicle, and thus is introduced in the workflow for the design case.

Additionally, SMMET use the mesh of the baseline configuration of IXV as input, thus the commercial mesh generation tool (Gridgen) which was used previously for non-lifting body workflow is not required for this workflow. Use of Gridgen in the non-lifting body workflow demonstrated that the developed design process can be easily extended to the commercial tools.

The TPS mass of the design case is computed by the SMMET using the baseline IXV configuration and TPS sizing is not included in the optimization process, thus the thermal analysis tool used for the non-lifting body workflow is also not required for the design case workflow. A constants block is added into the workflow of the design case to give the user a flexibility to perform either trajectory optimization of a given configuration or to perform the trajectory as well the geometry optimization of a lifting body.

Furthermore, as the design parameters are flowing from the Dakota parameter file to the toolchain in the non-lifting body optimization workflow are different than that of the lifting-body, the parameters flowing through the interfaces between these tools are also different.

## 6-2 Design Case Workflow

Figure 6-1 gives the workflow for the design case, where all the necessary tools are integrated on a single platform. This workflow can perform the shape morphing of the IXV baseline configuration and then estimate the payload mass, compute the aerodynamic database as well as perform the trajectory simulations and compute the corresponding aero-thermodynamics database.

As described in Chapter 2, a similar interface template is created for SMMET. The workflow shows that the design parameters, such as the scaling factors to morph the shape of the vehicle initially flows to the SMMET input template. These design parameters required for the SMMET flow from the design parameter (params.in) file to the input template of the SMMET. To verify the flow of parameters through this interface all the scaling factors are considered as 1, thus the SMMET gives the baseline configuration as shown in Figure 3-8. As described in previous chapters, an interface verification is performed, where it is observed that the design parameters from the Dakota parameter file are transferred to the input template of the tool. Here also, the scaling factors = 1 are exactly transferred from the Dakota parameter file to the input template of SMMET.

The scaling factors of the vehicle are described in Figure 3-6, the morphing of the configuration is performed with respect to the original IXV configuration. $X_{nose}$,$X_{body}$ and $X_{flap}$ are the scaling factors for the nose section, body section, and the flap section respectively. Furthermore, $Y_{start}$ is the scaling factor along width at the intersection of main-body and nose section and $Y_{end}$ is the scaling factor at the intersection of the main-body with flap section. Similarly to the scaling along the width,$Z_{start}$ and $Z_{end}$ are the scaling factors along the height of the vehicle.

The configuration is similar to the IXV in terms of the shape, whereas for the design case, the re-entry vehicle is considered with the payload capacity of the Space Rider as described in Rufolo (2016). The total mass of the vehicle is considered to be 2150 kg. From the interface between

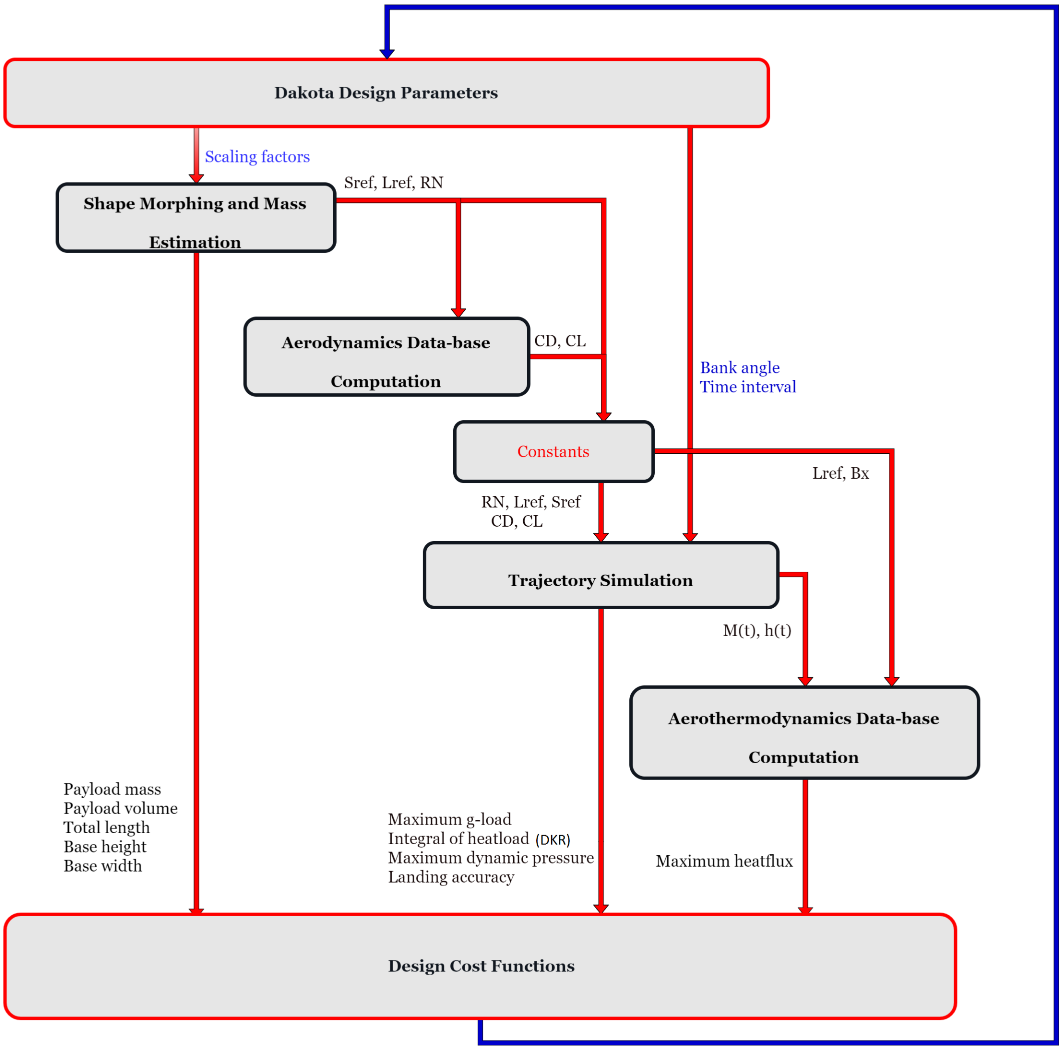



**Figure 6-1:** Workflow of design case

the Dakota parameter file and input template of the SMMET, the scaling factors are transferred correctly. Using this input the geometry simulations are carried out, where the dimensions, payload mass and volume of the re-entry vehicle are computed and then the output template of the SMMET is filled with the necessary parameter. Furthermore, the output from the tool is written in the same format as described in the Chapter 2 to utilize a common function to read and write the data in the interface templates. Table 6-1 gives the output of the SMMET for the given inputs.

Table 6-1 indicates that the design parameters flow correctly from the Dakota parameters file to the input template of SMMET and furthermore, the tool computes the desired output. The necessary results from the output template are then extracted to fill the Dakota results template. Here the values of payload mass, payload volume, total length, maximum base height and maximum base width are the major output from SMMET that flows from the output file of SMMET to the Dakota results template. The Dakota results template is created to collect all the values of cost functions involved in the design process. Additionally, a new geometry mesh is created by the SMMET for the given inputs and is saved in the working directory. As the interface verification between the output of SMMET and the input template of the Dakota results file, it is verified that exactly same values are transferred to the Dakota results file.

This geometry mesh is considered as input for the CoRADS tool, which is responsible for computing

**Table 6-1:** Output from SMMET

| Parameters | Value | Unit |
|---|---|---|
| Payload mass | 450 | [kg] |
| Payload volume | 0.71 | [m$^3$] |
| Total length | 5.05 | [m] |
| Maximum base height | 1.54 | [m] |
| Maximum base width | 2.23 | [m] |
| Radius of nose (RN) | 1.05 | [m] |
| Surface area (Sref) | 7.25 | [m$^2$] |
| Payload-bay height | 0.92 | [m] |
| Payload-bay width | 1.10 | [m] |
| Location of stagnation point (Bx) | 4.51 | [m] |

the aerodynamic database. In this case morphed design with the scaling factor of 1 is created and saved in the .mat format in the working directory. This mesh file is considered as the input for the aerodynamic database computation by CoRADS. Furthermore, the tool also requires additional parameters like surface area, reference length, and radius of the nose. Here, the total length of the vehicle is considered as the reference length for the aerodynamics database computation. Thus, there is an interface between the SMMET output file and the input template of the CoRADS. During the interface verification, it is verified that these values as mentioned in Table 6-1 flows successfully from the output file of SMMET to input template of the CoRADS. Using these inputs, the CoRADS tool compute the aerodynamic database such as the drag coefficient and the lift coefficient (CD and $C_L$) for different Mach number and angle-of-attack. This aerodynamic dataset is saved in the working directory and then considered for the trajectory simulator as input. Since for the interface verification, the scaling factors are considered as 1, the lift and the drag coefficient of the configuration computed by the CoRADS tool are same as described in Table 3-23.

Additionally, to keep the flexibility to perform only trajectory optimization of a fixed configuration, the user need to keep the geometry and its aerodynamic dataset as constant during the optimization. To maintain the simplicity in the workflow, a constants block is introduced into the workflow, which stores the variables from the SMMET and CoRADS, as well as the other constants such as the entry conditions, the total mass of the vehicle and target landing location coordinates. In case, the user needs to run only trajectory optimization, one can just copy the mesh of the configuration, its aerodynamic dataset and constants block in the working directory and then run the simulations. The constant block is written in the format of the interface template as discussed in Chapter 2. Thus, makes it user-friendly to store the constants or allow the user to flow the information from the geometry output file to the trajectory simulator input template.

The design parameters considered for the trajectory simulator flows from the Dakota parameter file to the input template of the trajectory simulator. Figure 3-15 describes the absolute bank angle and corresponding time interval considered as inputs for the trajectory simulator. To maintain the accuracy of the results, 20 absolute values of bank angles and the corresponding time intervals are considered for the trajectory simulator. The initial bank angle and the corresponding time interval is considered to be 0 deg and 200 sec respectively. Whereas, Table 3-25 gives the other 19 absolute bank angle and time intervals, which are derived from the nominal trajectory of the IXV as discussed in the trajectory optimization of the IXV in Chapter 5. The interface is same as the one used for the trajectory optimization of the given configuration as discussed in Section 5-1. Furthermore, the aerodynamics database and the geometric parameters such as the surface area, radius of the nose and the reference length computed from the SMMET is used as input by the trajectory simulator. Table 6-1 gives the values of the surface area, the radius of nose and reference length, which flows through the constants block for each simulation and is transferred to the input template of the trajectory simulator. The constants such as the total mass, entry conditions and target location coordinate flow to the simulator. The trajectory simulator is responsible for computing the trajectory of the re-entry vehicle considering the lateral guidance logic, that considers the bank angle maneuvering to reach the desired location. Table 3-24 gives the entry conditions, the target landing location co-ordinate, which are considered as constants for the trajectory simulator. For the design case, the total mass is considered as 2150 kg.

Using these inputs the trajectory simulator computes the g-load, dynamic pressure, landing accuracy, heat-flux and the corresponding integral of heat load. In the trajectory simulator, the heat flux is computed by using the DKR method. The necessary outputs are extracted from the output of the trajectory simulator and saved copied to the Dakota results template. Table 6-2 gives the output from trajectory simulator. These results are transferred to the Dakota results file through the interface between them. As an interface verification between the trajectory simulator results and the Dakota results file template, it is verified that the output described in Table 6-2 exactly transfers to the Dakota output file.

**Table 6-2:** Output from trajectory simulator

| Trajectory simulator output | | Units |
|---|---|---|
| Maximum heat load | 282.88 | [MJ/m$^2$] |
| Landing accuracy | 173.17 | [km] |
| Maximum g-load | 2.01 g | [m/s$^2$] |
| Dynamic pressure | 5608.92 | [Pa] |
| Maximum longitude | 235.36 | [deg] |
| Change in flight-path angle | 0.037 | [deg] |

The trajectory data is saved in the working directory, which is used by modified CoRADS as input. Additionally, as described in the workflow the trajectory results, the altitude and Mach number at which maximum heat flux occurs by DKR method are considered as input by the modified CoRADS tool. Furthermore, to derive the heat flux and time profile, the profile obtained by the DKR method are scaled by the considering the maximum heat flux value computed by the modified CoRADS. The modified CoRADS also requires the location at which the heat flux is computed and the reference length of the configuration. This is computed by the SMMET, which is stored in the constants block. Thus, there is an interface between this constant block and the modified CoRADS as described in Table 6-3.

**Table 6-3:** modified CoRADS

| Modified CoRADS inputs | | Units |
|---|---|---|
| Location of stagnation point (Bx) | 4.51 | [m] |
| Reference length (Lref) | 5.05 | [m] |

Here, the location of the stagnation point (Bx) is required input for the modified CoRADS to compute the maximum heat flux at the stagnation point. The location is measured from the intersection point of the main body and the flap. Table 6-4 gives output from modified CoRADS. These results flow from the interface between the output of Modified CoRADS and the input template of the Dakota results file. From the interface verification, it is seen that the exactly same values as given in Table 6-4 are transferred to the input template of the Dakota results file.

The results from all the tools within the toolchain are collected in the Dakota results file. The Dakota results template includes the major cost functions which are considered for the feasibility

**Table 6-4:** Interface between the output of Modified CoRADS and the input template of the Dakota results file

| Parameters | Modified CoRADS | Units |
|---|---|---|
| Maximum heat flux | 0.46 | [MW/m$^2$] |

study of the vehicle. These values of the cost functions are then given back to the Dakota optimizer, which generates the new design parameters for the further simulation.

The above verification of the interfaces is performed for a configuration with scaling factor =1, i.e IXV with payload carrying capacity. Thus the results of this open loop verification are considered in Section 6-3 to compare the results of optimized configurations of the design case. The comparison of these results with the optimized configuration will allow one to see the improvement in the performance of the re-entry vehicle with respect to the initial configuration, which will answer the main research question.

# 6-3 Results

This section discusses the results that answer the main research question. Subsection 6-3-1 describes the sensitivity analysis results, where the influence of the design parameters of the cost function is analyzed. Once, the sensitivity analysis is performed the objective and constraints for the optimization problem are defined and their values are set to derive the optimized configuration. The results of the optimization of the design case are discussed in Subsection 6-3-2. Furthermore, this section discusses the best-optimized configuration and the trajectory in details.

For the sensitivity analysis and for the MDO of the re-entry vehicle, the verified workflow as shown in Figure 6-1 is used. Table 6-5 gives the entry conditions and the target landing location considered for the sensitivity analysis and for the MDO of the design case.

**Table 6-5:** Initial conditions for the feasibility study of the design case

| | | |
|---|---|---|
| Entry altitude | 120 | [km] |
| Entry latitude | -4.47 | [deg] |
| Entry longitude | 173.48 | [deg] |
| Entry velocity | 7434.85 | [m/s] |
| Entry heading angle | 86.68 | [deg] |
| Entry flight-path-angle | -1.21 | [deg] |
| Entry angle-of-attack | 45 | [deg] |
| Target landing latitude | 3.27 | [deg] |
| Target landing longitude | 236.87 | [deg] |
| Initial bank angle | 0 | [deg] |
| Initial time interval | 300 | [sec] |

## 6-3-1 Sensitivity Analysis

As described in Section 2-4, the LHS explores the complete design space. Additionally, the LHS method available technique within the Dakota framework also provides partial rank correlation factors as an output, which use the Spearman relationship to compute the correlations between two variables by controlling the influence of the third variable, as given in Equation 2-6. This gives the perfect idea about the correlation between two variables involved in a multi-variable problem. Thus, this thesis considers the sensitivity analysis using the LHS method and deriving the partial rank correlations factors using the Spearman's formulation. These results will answer the following questions:

- Which design parameters contribute to the results and how much?
- What should be the constraints values for the optimization?

These results allow one to re-define the optimization problem by selecting the appropriate range of the design parameters and values for the constraints, considered during the optimization. Furthermore, the analysis of the samples gives the user an idea of the location of the optimized solution within the design space.

### Design Parameters

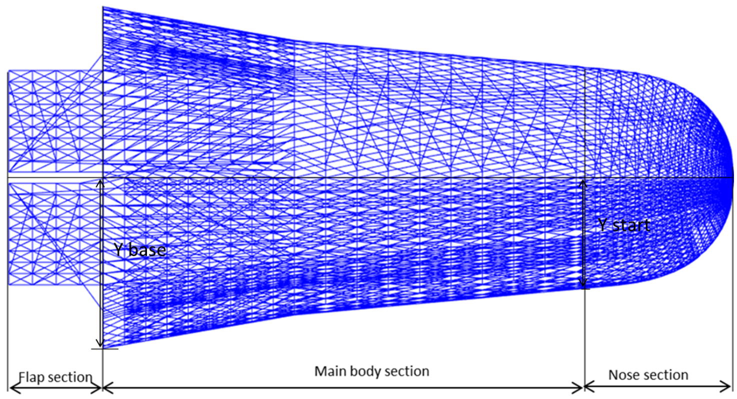



**Figure 6-2:** Sections along the length of configuration

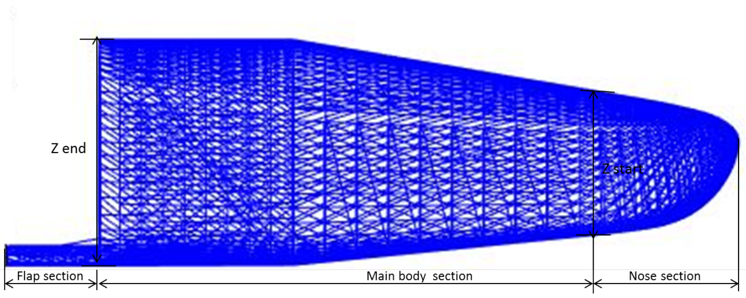
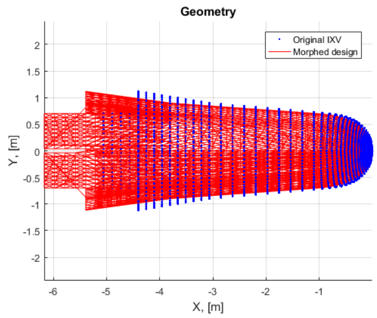



**Figure 6-3:** Sections along the length of configuration

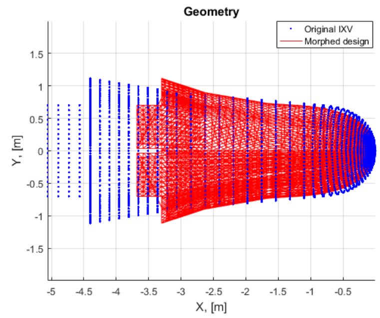
Figure 6-2 and Figure 6-3 gives the scaling factors considered for the shape morphing of the baseline configuration. The scaling factors $X_{nose}$, $X_{body}$ and $X_{flap}$ corresponds to the scaling of the nose, main body and the flap sections along the length respectively. Furthermore, to morph the design along the width, $Y_{start}$ and $Y_{end}$ are the scaling factors introduced as shown in Figure 6-2. Similarly, to morph the design along the height, $Z_{start}$ and $Z_{end}$ are the scaling factors introduced as shown in Figure 6-3. Using this scaling factors, the initial mesh of IXV is morphed along the length, width, and height. The scaling along the width, and the height are performed by linear interpolation between the nose section to the flap section interface with the main body section.

As a function of these scaling factors, the baseline mesh of IXV can be morphed. The following figures give the new configurations and their comparison with the baseline IXV. Figure 6-4a and

Figure 6-4b gives the variation of the configuration due to scaling along the length while keeping the width and height same as baseline IXV.

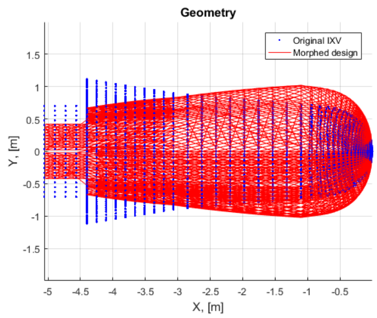



**(a)** $X_{nose}$=1.3, $X_{body}$=1.2, $X_{flap}$ =1

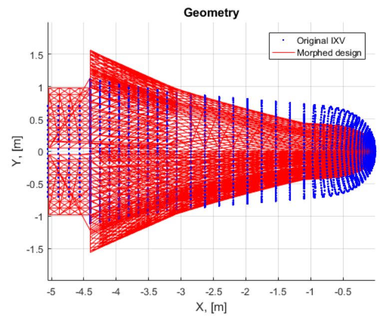



**(b)** $X_{nose}$=1.5, $X_{body}$=0.5, $X_{flap}$ =0.8

**Figure 6-4:** Illustration of configurations with scaling along length

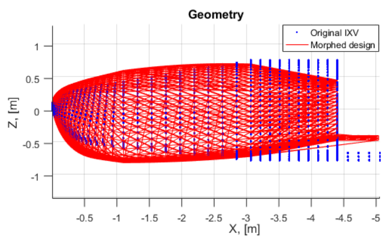



**(a)** $Y_{start}$ =1.4, $Y_{end}$= 0.6

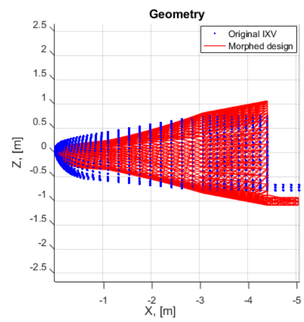



**(b)** $Y_{start}$ =0.6, $Y_{end}$= 1.4

**Figure 6-5:** Illustration of configurations with scaling along width




**(a)** $Z_{start}$ =1.4, $Z_{end}$= 0.6




**(b)** $Z_{start}$ =0.6, $Z_{end}$= 1.4

**Figure 6-6:** Illustration of configurations with scaling along height

Figure 6-5a and Figure 6-5b gives the configurations with scaling along the width while keeping the length and height same as baseline IXV. Furthermore, Figure 6-6a and Figure 6-6b gives the configurations with scaling along the height while keeping the length and width same as baseline IXV.

From these figures, it can be seen that there is a large variation in the configuration with the variation in the scaling factors. All these 7 scaling factors can be simultaneously varied to morph the baseline IXV mesh. As described in Chapter 3, the SMMET involves the linear interpolation for the body section. Similarly, more than 3 sections can be introduced to add more design parameters and increase the complexity of the problem.

As described in Chapter 3, Figure 3-15 describes the trajectory related design parameters such as bank angle and time interval. The initial bank angle and time interval is 0 deg and 200 seconds, respectively. While the other absolute value of bank angle varies between 0 to 90 deg and corresponding time interval varies from 5 to 300 sec. In the design case, CoRADS is considered for the aerodynamic dataset computation. As described in Chapter 3, CoRADS use modified Newtonian method to compute the aerodynamic dataset, thus the trajectory simulations are stopped once the Mach number is less than 3.0 Mach.

Table 6-6 gives the range of the design parameters used considered to generate the samples using the LHS method. The design parameters are varied such that complete design space can be explored by using the LHS.

**Table 6-6:** Design Parameters

| Design parameters | Range | Units |
|---|---|---|
| Scaling factor $X_{nose}$ | $\pm$ 40 % | [ - ] |
| Scaling factor $X_{body}$ | $\pm$ 40 % | [ - ] |
| Scaling factor $X_{flap}$ | - 40 % | [ - ] |
| Scaling factor $Y_{start}$ | $\pm$ 40 % | [ - ] |
| Scaling factor $Y_{end}$ | $\pm$ 40 % | [ - ] |
| Scaling factor $Z_{start}$ | $\pm$ 40 % | [ - ] |
| Scaling factor $Z_{end}$ | $\pm$ 40 % | [ - ] |
| Bank angle | 0 to 90 | [deg] |
| Time interval | 5 to 300 | [sec] |

As described in Chapter 2, the LHS method within the Dakota framework is used to explore the design space and furthermore compute the partial rank correlation factors using the Spearman relationship as given in Equation 2-6. Thus using this relationship, the partial rank correlation factors are computed between the design parameters and the cost functions. Figure 6-7 gives the partial rank correlation factors between the geometry related design parameters and cost functions.

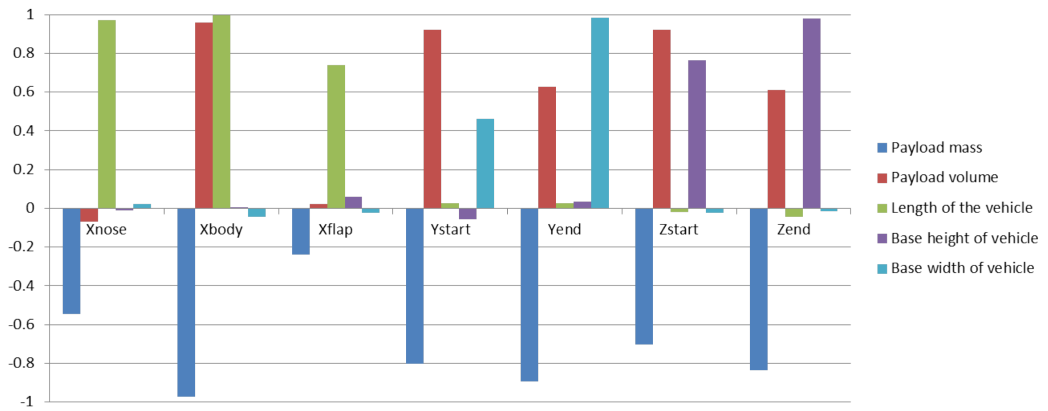



**Figure 6-7:** Scaling factors influence on the objectives

From 6-7, the payload mass has an inverse co-relation with all the scaling factors. If the vehicle is scaled up, the mass of the TPS and the cold structure increases. Since the total mass of the vehicle is assumed to be constant, the capacity of payload mass decreases. The TPS sizing is assumed to be uniform in the respective sections and the mass of the TPS is computed from the baseline configuration as explained in Section 3-1-1, while considering the total mass of the vehicle as constant. The body section is the largest section of the vehicle and thus $X_{body}$ largely influence the payload mass of the vehicle. The geometry generation involves a linear interpolation along the width and the height of the vehicle, thus $Y_{start}$ and $Z_{start}$, $Y_{end}$ and $Z_{end}$ indicates approximately a similar correlation factor values. The volume of the payload section increases with the increase in the scaling factors, whereas the payload section volume depends majorly on the scaling factor of the body section. This can be seen from the higher correlation between the $X_{body}$ and the payload volume.

The total length of the vehicle depends on the scaling factors along the length thus $X_{nose}$, $X_{body}$ and $X_{flap}$ influence the total length. The length of the flap is defined as the function of $X_{nose}$ and $X_{body}$, thus the effect of the scaling the flap on the total length is mitigated to the scaling factors of $X_{nose}$ and $X_{body}$. Similarly, the width and the height of the vehicle, is influenced by scaling factor along the width and height of the vehicle. Thus, the scaling factors $Y_{start}$, $Y_{end}$ and $Z_{start}$,$Z_{end}$ largely contribute to the width and the height of the vehicle respectively.

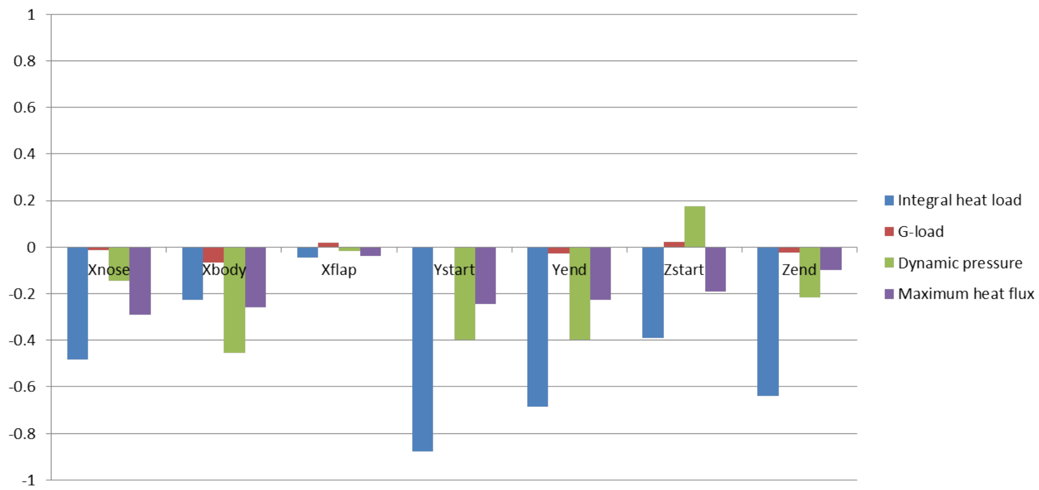



**Figure 6-8:** Scaling factors influence on the objectives

Figure 6-8, shows the influence of the scaling factors on the integral heat load, g-load, heat flux and dynamic pressure. The heat load is the integral of the heat flux which is directly depended on the scaling factors. The radius of the nose is defined by the geometry generation as a function of the scaling factors of the nose section. As discussed in Chapter 5, the integral of heat load is considered as an objective for the trajectory optimization, to improve the heat flux profile. The integral heat load considered here is the integral of the heat flux computed by the DKR method through trajectory simulator. The heat flux considered in the optimization process is computed through the modified CoRADS, which considers the uncertainties. Using the maximum heat flux value computed from the modified CoRADS, the heat flux time profile derived from the trajectory simulator is scaled to derive the heat flux time profile. Thus, the heat flux and heat load are not having the same correlation with the scaling factors as expected. Furthermore, the radius of the nose is defined as the function of the scaling factors and the correlation of the radius of the nose with the heat-flux gives a high correlation as shown in Figure 6-9.

The dynamic pressure depends on the surface area, whereas the scaling factor influences the surface area of each section. To compute the surface area, the geometry tool considers scaling of the surface area based on the ratio of the total surface area of the morphed design to the total surface area of the unscaled geometry. Thus, the correlations show the influence of the each section surface area

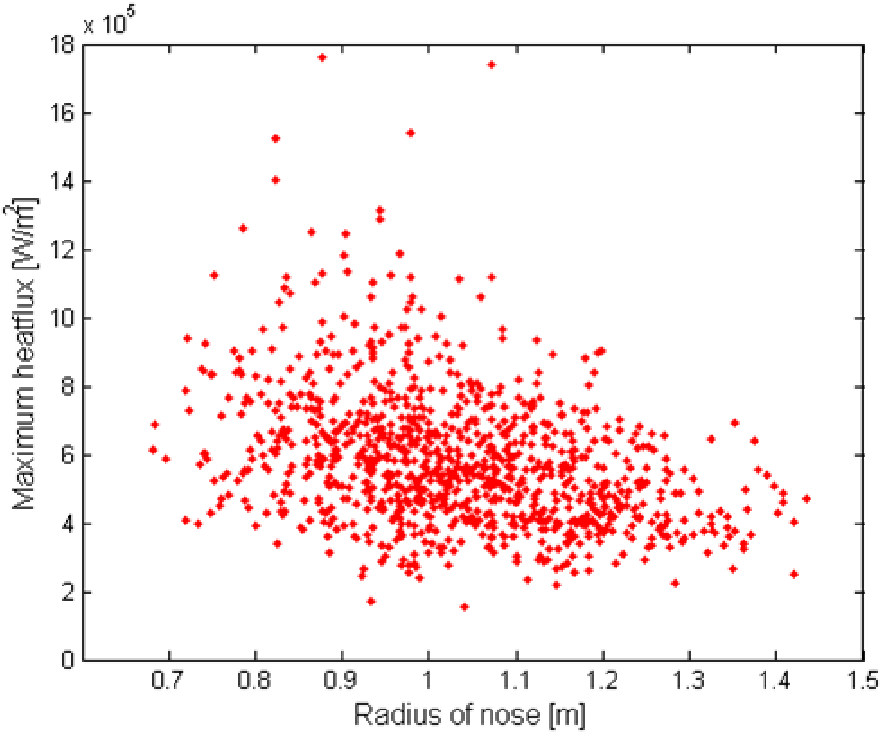



**Figure 6-9:** Correlation of the heat flux with the radius of the nose

as the total surface area is a summation of the surface area of each section. Depending on the number of the elements in each section, the contribution to the total surface area. Furthermore, the surface area of the triangular elements is considered thus, the scaling along the height of the vehicle is not contributing much to the total surface area of the vehicle.

Similar to the trajectory optimization of IXV configuration as discussed in Chapter 5, The design parameters related to the trajectory optimization include 19 bank angle and 19 corresponding time intervals. From the results, it is observed that only initial 10 bank angle and their corresponding time intervals largely influence the cost functions. Figure 6-10, gives the partial rank correlation factors between these design parameters and the cost functions.

From the figure, it is seen that the peak of the dynamic pressure and the g-load occurs in the later phase of the trajectory and thus the initial bank angle does not influence peak of the g-load and dynamic pressure. One can observe the higher correlation for the later phase bank angle such as bank angle 4 to bank angle 8, where the peak of g-load and dynamic pressure occurs. Similarly, the peak of heat flux occurs before the peak of g-load and thus bank angle 4 largely influence the peak of heat flux. The heat load is an integral of the heat flux and thus the initial bank angle also highly influence the integral heat load. Furthermore, the lateral guidance computes the required bank angle maneuvering based on the heading angle error to reach the target landing location. Thus, even though initial bank angle shows a correlation above 0.2, there is no direct strong correlation between the landing accuracy and the bank angles. Additionally, the partial correlations factors between the time interval and the cost functions are between 0 to 0.2, which is a very small correlation, and thus does not majorly influence the cost functions values. As described in Chapter 5, the design parameters are the time intervals and not the absolute time at which the respective bank angle occurs. Thus, there is no major influence of these time intervals on the cost function. Considering the sensitivity analysis results, the multi-disciplinary optimization of the design case is performed to answer the main research question in the following section.

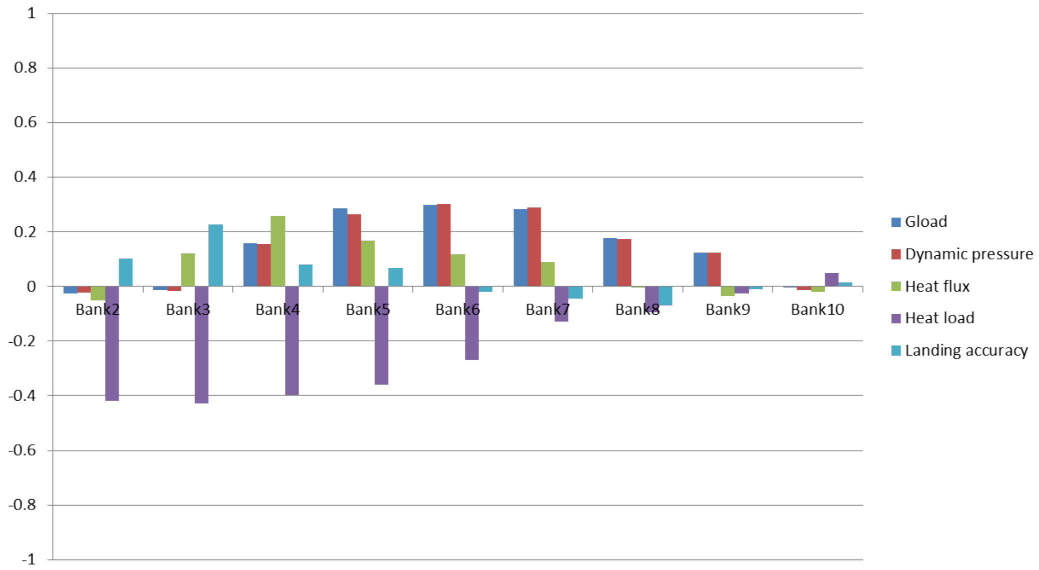



**Figure 6-10:** Effect of the variation of the bank angle on the objectives

## 6-3-2 Optimization Results

Once the sensitivity analysis is performed, the results are used to setup the problem for the optimization. The design parameters and entry conditions are mentioned in Table 6-6 and Table 6-5 respectively. It is observed that all the geometry related design parameters considered during the sensitivity analysis influence the cost functions. Furthermore, from the sensitivity analysis, it is observed that initial 10absolute values of bank angle only influence the results. Additionally, the time intervals do not have any major effect on the results. Thus, the problem can be setup with less number of bank angle and time intervals, but as observed during the verification of the trajectory optimization, to match the bank angle profile with the nominal bank angle profile of the IXV large number of bank angle and time intervals are considered. Similarly, for the open loop verification of the design case workflow, the bank angle and corresponding time intervals are same as IXV nominal trajectory. The optimized results will be further compared with the open loop trajectory results to see the improvement in the performance of the design case. Thus, all 19 bank angle and time intervals are considered for the optimization. The initial bank angle and the time interval to be 0 deg and 200 seconds respectively. MOGA is considered as a MDO technique for optimization of the re-entry vehicle.

As discussed previously, for the design case, the geometry and the trajectory optimization workflows are clubbed together, the major objectives and the constraints are also derived from the previous workflows. Table 6-7 and Table 6-8 gives the objectives and constraints used for the optimization.

**Table 6-7:** Objectives

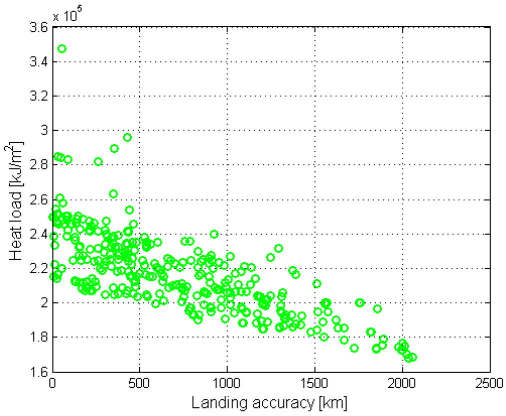
| Design objectives |
|---|
| Maximize the payload mass |
| Maximize the payload volume |
| Minimize the integral heat load |
| Minimize the distance from landing to target |

From Table 6-7, it is observed that the payload mass and the payload volume is considered to be maximized. The objectives are derived from the mission requirement to carry maximum cargo from the ISS to the Earth. Furthermore, the integral heat load is minimized. This heat load is the integral of the heat flux computed by the trajectory simulator using the DKR method. As described previously, the heat flux from the modified CoRADS is used to compute the value of the maximum heat flux which is then used to scale the heat flux time profile derived from the DKR method. Thus, minimizing the integral heat load improves the heat flux profile of the vehicle, where the re-entry vehicle follows a trajectory with the maximum heat flux for a longer duration. Additionally, to reach the target landing location, the landing accuracy is maximized. The landing accuracy is maximum if the distance between the target location and the actual landing location is minimum.

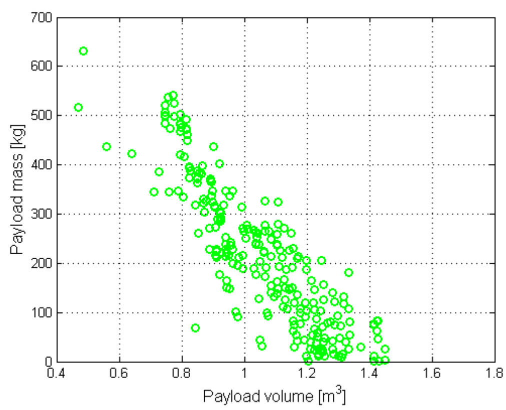
**Table 6-8:** Constraints

| Constraints | | |
|---|---|---|
| Maximum heat flux | $\leq 0.6$ | [MW/m$^2$] |
| Maximum g-load | $\leq 3$ g | [m/s$^2$] |
| Maximum Dynamic pressure | $\leq 5000$ g | [Pa] |
| Maximum total length | $\leq 7.2$ | [m] |
| Maximum base height | $\leq 2.9$ | [m] |
| Maximum base width | $\leq 2.9$ | [m] |
| Change in flight-path angle | $\leq 0.2$ | [deg] |
| Maximum longitude | $\leq 237$ | [deg] |

From Table 6-8, the constraints include the limit over the maximum heat flux, g-load, and dynamic pressure. These constraints are defined for the trajectory optimization, whereas the bank angle maneuvering is performed such that the vehicle meets the necessary limits on the cost functions. Additionally, to fit the vehicle inside the launcher, there are constraints over the configuration envelope volume, which is defined by the constraints over the total length, width, and height. As described in Chapter 5, the trajectory overshoot the target landing location as well as there is unrealistic behavior in the flight-path angle. To avoid the overshoot of the trajectory, a constraint is implemented on the maximum longitude. Additionally, a constraint over the change in flight path angle is introduced such that there is no steep change in flight-path angle. The angle-of-attack is considered to be 45 degrees during the hypersonic phase for the trajectory simulations. Considering the objectives and the constraints, MDO of the re-entry vehicle is performed.




**(a)** Heatload v/s landing accuracy




**(b)** Payload mass v/s payload volume

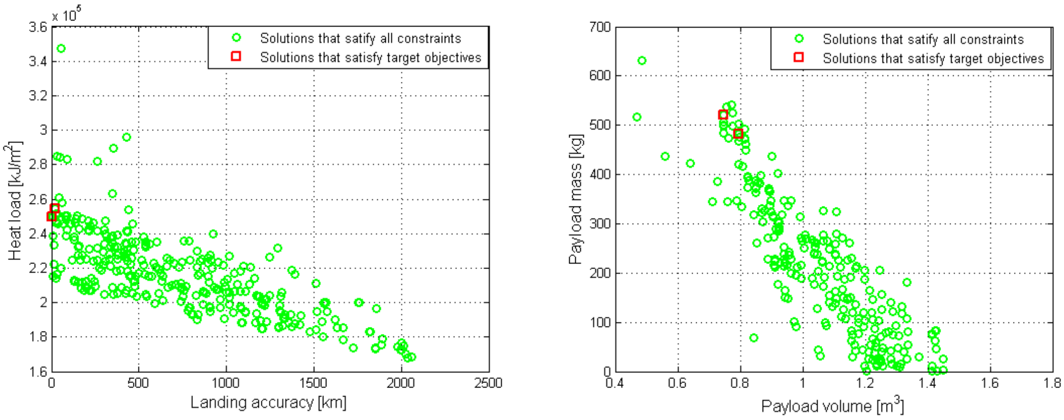
**Figure 6-11:** Pareto front

Figure 6-11a and Figure 6-11b gives the Pareto front between the objectives considered for the optimization. Figure 6-11a gives the Pareto front between the integral heat load and the landing accuracy, whereas the Figure 6-11b, gives the Pareto front between the payload mass and the payload volume. These solutions satisfy all the constraints described in Table 6-8.

Various challenges were faced during the optimization. The major challenges involved in performing the trajectory and shape optimization simultaneously. Thus, to simplify the problem, the optimization is performed with major objective functions and constraints to obtained a converged solution, and then the solutions that meet the target values of the objective functions are filtered from the complete set of the optimized solutions. To select these desired solutions from the pool of the obtained optimized solutions, target values for the objectives are set from the requirements of design case, as described in Table 6-9.

**Table 6-9:** Designcase configuration characteristics

| Target of objectives | | |
|---|---|---|
| Integral heat load | $\leq 300$ | [MJ/m$^2$] |
| Payload mass | $\geq 450$ | [kg] |
| Payload volume | $\geq 0.7$ | [m$^3$] |
| Landing accuracy | $\leq 25$ | [km] |




**(a)** Heatload v/s landing accuracy

**(b)** Payload mass v/s payload volume

**Figure 6-12:** Solutions that meet the target of objective functions

Figure 6-12a and Figure 6-12b gives the solutions that satisfy all the target values of the objectives. From the figures, it is observed that there are 2 optimized solutions that meet the target values of all the objectives of the re-entry vehicle. The further discussion is narrowed down to these solutions and their comparison with the open loop results to observe the improvement in the performance of the vehicle by the utilization of the design process developed with the Dakota framework. The results of the open loop simulations were mentioned in section 6-2. The configuration considered here is derived by considering the geometry design parameters same as IXV, (scaling factors =1). The trajectory design parameters of the open loop simulation as same as IXV nominal trajectory. Figure 6-13 gives the baseline configuration, where scaling factors are equal to 1.

Design case 1 can carry 520.33 kg payload mass with 0.74 m$^3$ payload volume capacity, whereas design case 2 can carry 482.26 kg payload mass with 0.79 m$^3$ payload volume capacity. If this is compared with the baseline payload mass of 450 kg and 0.71 m$^3$ payload volume capacity, both the design cases selected gives better performance. If the geometry is compared, based on the mission requirement if the payload mass capacity is the first preference then design case 1 is

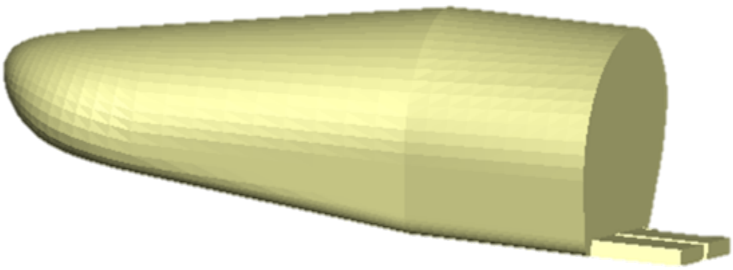


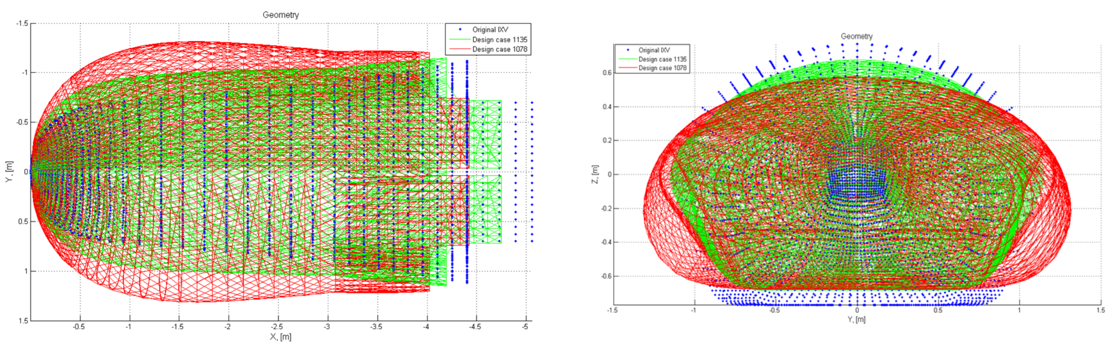
**Figure 6-13:** Initial configuration

better solution else design case 2 for the payload volume capacity. Table 6-10 gives the comparison of the scaling factors of the derived optimized configurations with the baseline configuration. The baseline geometry is a configuration with the scaling factor =1.

**Table 6-10:** Design parameters for shape morphing

| Parameters | Baseline geometry | Design case 1 (ID:1078) | Design case 2 (ID:1135) |
|---|---|---|---|
| Scaling factor $X_{nose}$ | 1 | 1.286 | 1.286 |
| Scaling factor $X_{body}$ | 1 | 0.843 | 0.843 |
| Scaling factor $X_{flap}$ | 1 | 0.6943 | 0.9125 |
| Scaling factor $Y_{start}$ | 1 | 1.360 | 1..360 |
| Scaling factor $Y_{end}$ | 1 | 1.029 | 1.029 |
| Scaling factor $Z_{start}$ | 1 | 1.06 | 1.146 |
| Scaling factor $Z_{end}$ | 1 | 0.7004 | 0.7004 |

Figure 6-14a and Figure 6-14b gives the comparison of the top view and the front view of the configurations. Furthermore, Figure 6-15 gives the side view of 2 optimized solution's configurations respectively.




**(a)** Top view **(b)** Front view

**Figure 6-14:** Comparison of the configuration

From these solutions of the design cases, it can be seen that the scaling factors are almost similar

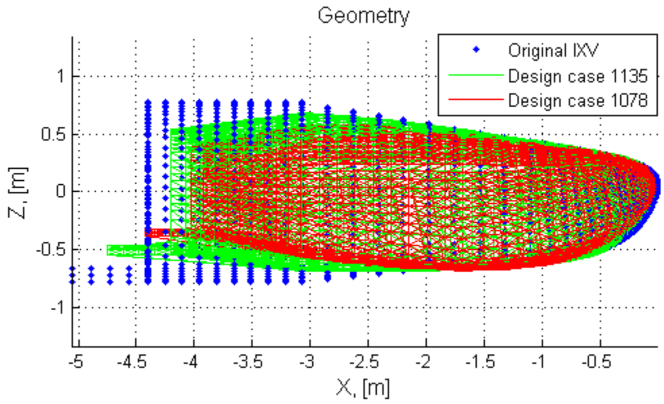



**Figure 6-15:** Side view

to each other. From Table 6-10, it is observed that the $X_{nose}$, $X_{body}$, $Y_{start}$, $Y_{end}$ and $Z_{end}$ are exactly same for both the configurations. The difference is observed in the $X_{flap}$, furthermore, as the total mass of the vehicle is kept constant only payload mass objective indirectly drives the size of the flap and this effect is reflected in the values of payload mass of these configurations. Furthermore, the $Z_{start}$ of design case 2 is slightly larger than the design case 1 and the effect can be explained the sensitivity analysis results as shown in Figure 6-7, where it can be observed that the payload volume is positively correlated with the $Z_{start}$.

Furthermore, the trajectory plots of the design case 1 (ID:1078) and design case 2 (ID:1135), are compared with the initial trajectory of the baseline configuration (scaled factor =1). Figure 6-16 gives the bank angle profile of the design cases and their comparison with the initial trajectory. It is observed that the initial conditions are same for the design cases and the initial trajectory, but the number of bank angle reversal reduces for the design cases. As there is no strict requirement on the number of bank-angle reversals, the trajectories are only derived to meet the objectives and the satisfy the constraints. The design case 1 required a single bank reversal to reach the landing location, whereas the design case 2 reach the landing location with 2 bank reversals.

Figure 6-17a and Figure 6-17b gives the altitude-time profile and the altitude-velocity profile of the design case 1 (ID:1078) and design case 2 (ID:1135) respectively. These results are compared with the initial trajectory of the baseline configuration. As described in Chapter 5, the difference between the bank angle profile leads to the variation in altitude-time profile and altitude-velocity profile. The trajectory of the design case 2 (ID:1135), gives a smooth profile as compared to the trajectory of the design case 1 (ID:1078). The design case 1 (ID:1078) follows a trajectory with deviations, which are caused due to the bank angle maneuvering to reach the target landing location, the magnitude of the vertical lift varies during the bank angle maneuvering that results into the deviations during the trajectory.

Figure 6-18a gives the comparison of the flight-path angle profiles of the design cases with the initial trajectory. The effect of the difference in the bank angle profile also depicted in the flight-path angle profile, where the slight deviation in the design case 2 is explained by the bank angle maneuverings. Figure 6-18b gives the heading angle profile of the design cases and it's comparison with the initial trajectory. The difference can be explained by the number of bank reversals. As observed in the bank angle profile, the design case 1 performs single bank reversal and thus heading angle do not change the signs of its slope, whereas for the design case 2, since there are 2 bank

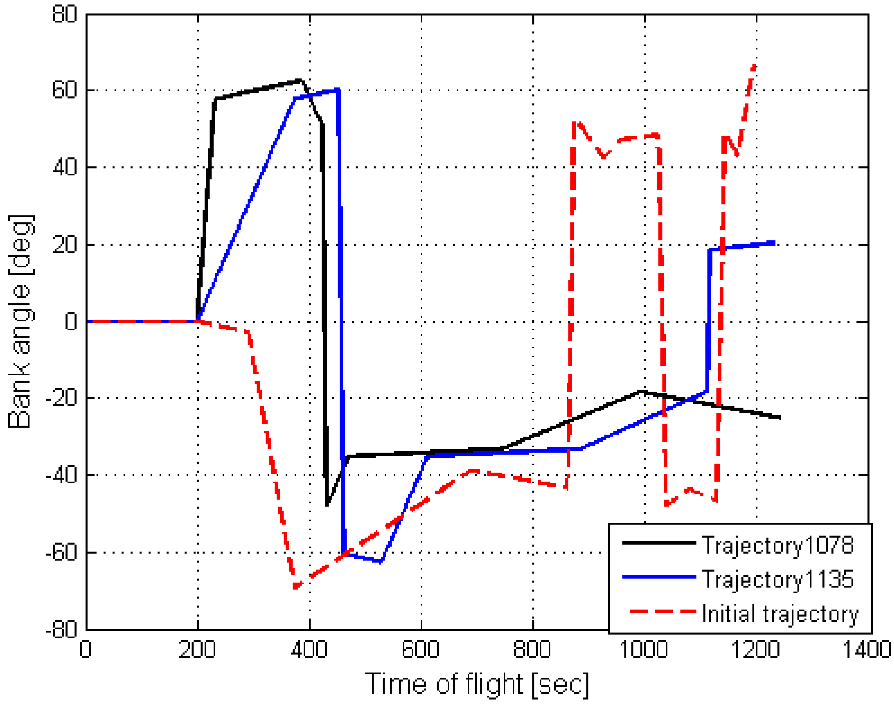
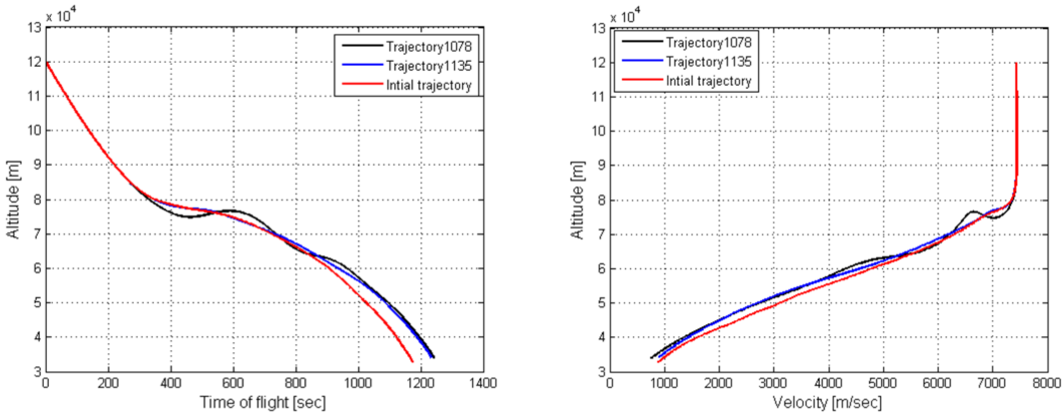



**Figure 6-16:** Comparison of the bank angle profile of design cases with the initial trajectory

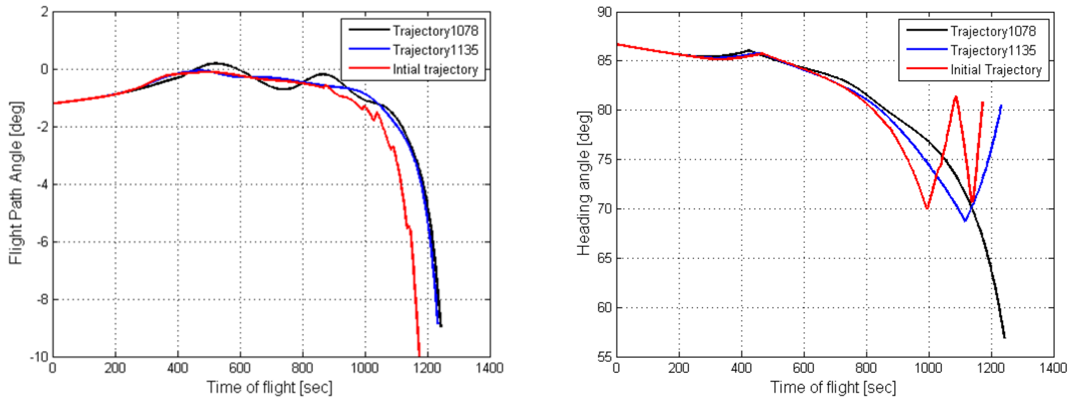



**(a)** Altitude profile **(b)** Altitude-velocity profile

**Figure 6-17:** Comparison of the altitude and velocity profile of the optimized trajectories with the initial trajectory

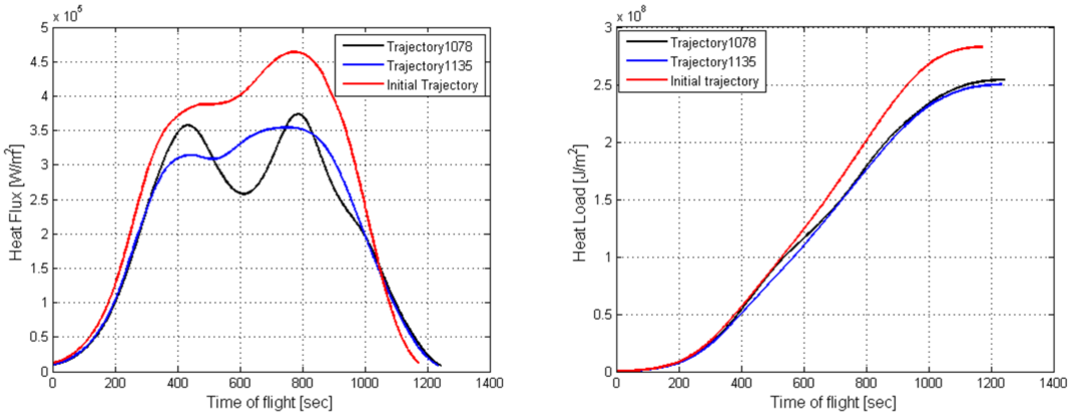
reversals there is a change in the signs of its slope.

This is also explained further by the heat flux profile of these design cases, where the design case 1 (ID:1078) fly with more variation in the flight-path angle resulting in the large difference between the peak values of the heat flux profile, whereas the design case 2 (ID:1135) continues to fly at the maximum heat flux constraint for longer period of time resulting in the lower heat load as well as the less difference between the peak of the heat flux as shown in Figure 6-19a and Figure 6-19b.




**(a)** Flight-path angle profile **(b)** Heading angle profile

**Figure 6-18:** Comparison of the flight-path angle and the heading angle profile of the optimized trajectories with the initial trajectory

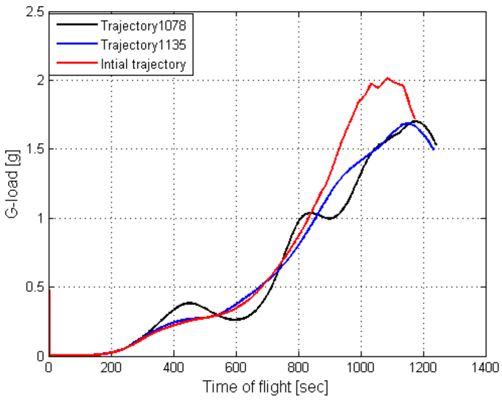



**(a)** Heatflux profile **(b)** Heatload profile

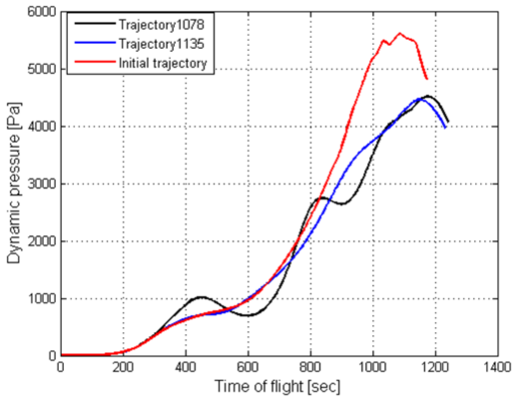
**Figure 6-19:** Comparison of the heatflux and heatload profile of the optimized trajectories with the initial trajectory

By comparing the objectives the maximum heat flux and the heat load values are lower for the design case 1 and design 2 with respect to the initial trajectory, thus these solutions are better than the initial trajectory. Furthermore, in terms of the heat flux profile, for a realistic flight, the vehicle should fly along the maximum heat flux to reduce the integral of the heat load, thus design case 2 profile can be considered as better than the design case 1 as well as the initial trajectory.

From the g-load profile and dynamic pressure profile of the design cases as shown in Figure 6-20a and Figure 6-20b it is observed that both the design cases shows better trajectory in terms of the maximum g-load and maximum dynamic pressure. The variation in the flight-path angle of the design case 1 also influences the dynamic pressure and g-load profile. As the velocity and the acceleration are the functions of the flight path angle, the g-load and the dynamic pressure are also influenced by its variations. Even though both the design cases are better solutions than the initial trajectory, the design case 2 is better trajectory as the maximum g-load and the dynamic pressure is lower for the design case 2 than the design case 1 trajectory. Thus, in terms of the g-load and



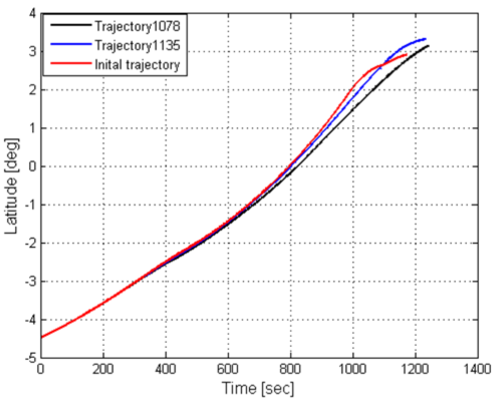

**(a)** G-load profile



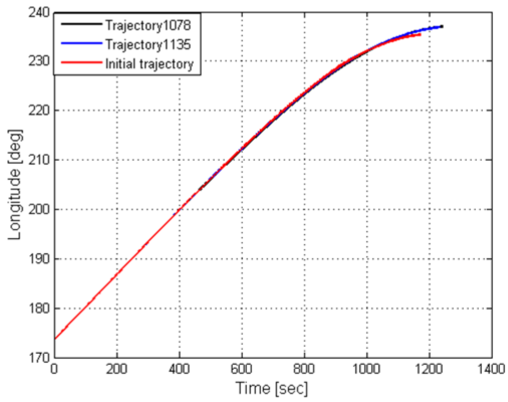

**(b)** Dynamic pressure profile

**Figure 6-20:** Comparison of the G-load and the dynamic pressure optimized trajectories with the initial trajectory

the dynamic pressure as well the design case 2 is considered as better optimized trajectory than the design case 1.




**(a)** Latitude profile




**(b)** Longitude profile

**Figure 6-21:** Comparison of the latitude and the longitude profile of the optimized trajectories with initial trajectory

Figure 6-21a and Figure 6-21b gives the latitude and the longitude profile of the design cases and their comparison with the initial trajectory. Additionally, Figure 6-22 gives the latitude and the longitude profile of the design case, where it can be observed that the design case 2 trajectories give the maximum landing accuracy of 0.4 km, whereas the design case 1 gives the landing accuracy of 18.87 km. Both the design case gives better landing accuracy than the initial trajectory, which gives a landing accuracy of 173 km.

For the comparison of the trajectory generally, the major objectives are considered to compare the solutions and the trajectories within the desired range of the landing accuracy are considered as the best solution. Here, it is observed that the design case 2 is not only giving a better landing accuracy but also better performance in terms of the maximum heat flux, g-load, heat-load and maximum dynamic pressure.

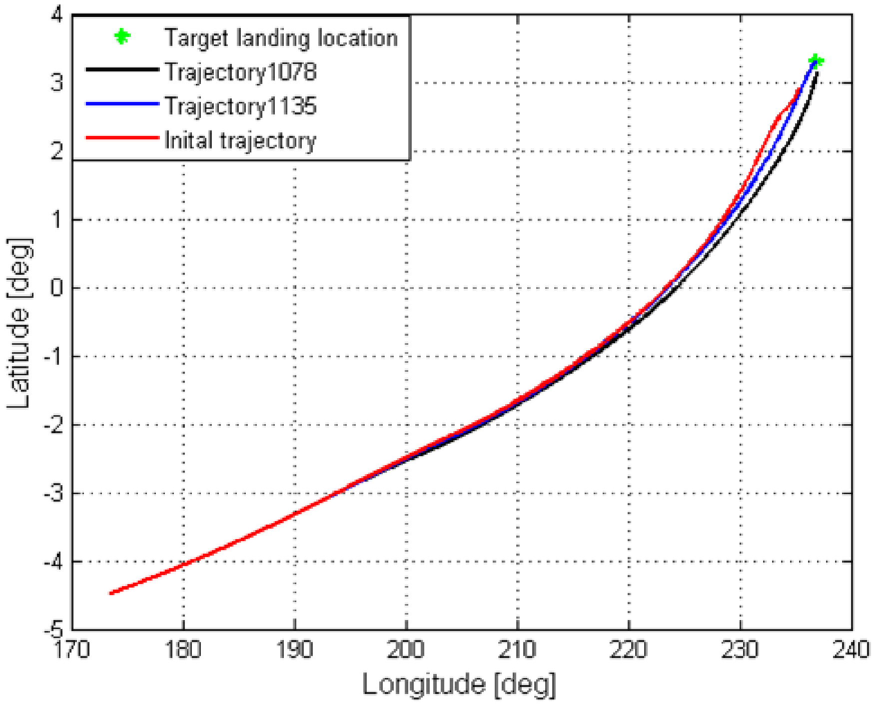



**Figure 6-22:** Comparison of the latitude v/s of longitude of design cases with the initial trajectory

**Table 6-11:** Comparison factors for the design case

| Results | Design case 1 ID:1078 | Design case 2 ID:1135 | Initial design design | Units |
|---|---|---|---|---|
| Payload mass | 520 | 482 | 450 | [kg] |
| Payload volume | 0.74 | 0.79 | 0.71 | [$m^3$] |
| Integral of heat load | 254.28 | 250.04 | 283.00 | [$MJ/m^2$] |
| Landing accuracy | 18.87 | 0.40 | 173 | [km] |
| Maximum heat flux | 0.37 | 0.35 | 0.46 | [$MW/m^2$] |
| Maximum g-load | 1.70 g | 1.68 | 2.01 | [$m/s^2$] |
| Maximum dynamic pressure | 4.51 | 4.45 | 5.61 | [KPa] |
| Maximum total length | 4.64 | 4.77 | 5.06 | [m] |
| Maximum base height | 1.29 | 1.35 | 2.24 | [m] |
| Maximum base width | 2.31 | 2.31 | 1.54 | [m] |

Table 6-11 gives the comparison of the design cases with the initial configuration. Both these design cases give better performance than the initial configuration in terms of the payload mass capacity of design case 1 (ID:1078) is higher than the design case 2 (ID:1135), whereas in terms trajectory performance, design case 2 is better than design case 1. Figure 6-23 gives the graphical comparison between the objectives of the design cases and the initial configuration.

Design case 2 is better than design case 1 in terms of the payload volume, integral of heat load and landing accuracy. Additionally, it is also considered that the trajectory profiled obtained for the design case 2 are better than the design case 1, thus design case 2 is chosen as the best-optimized

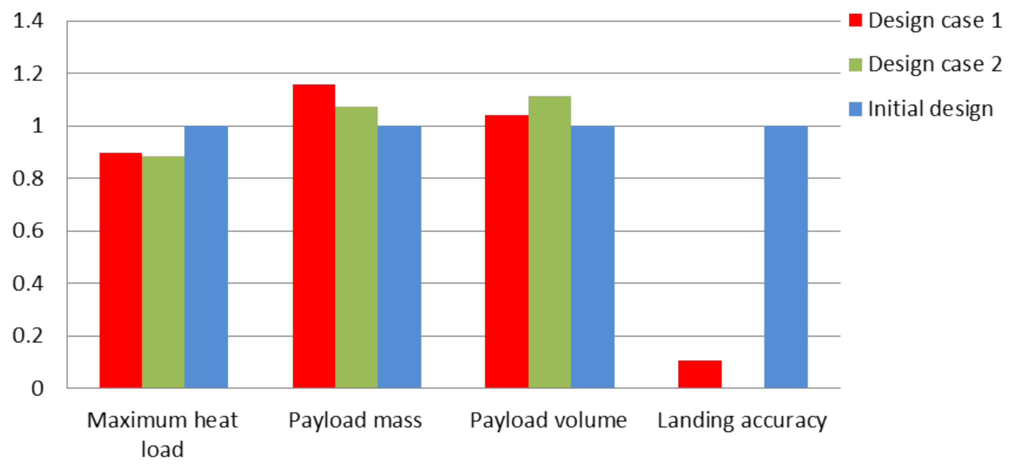



**Figure 6-23:** Comparison between all the objectives of all 3 design configurations

configuration. Thus, it can be stated that an optimized configuration is successfully derived from the design process developed within the Dakota framework. This also answers the main research question, where an optimized configuration similar to IXV is derived, for the given mission scenario and systems requirements.

Chapter 7

# Conclusions and Recommendations

The ambition and curiosity of humans drive advancements in space exploration technologies and to accomplish this dream, we need safe space transportation. It is important to have an efficient cargo retrieval system, that can enable frequent space transportation from the ISS to the Earth. Despite having efficient models for each discipline, during the preliminary design phase, we lack the specific and standardized design analysis techniques to be used by engineers and designer. To perform the feasibility study of a complex engineering system like re-entry vehicle, using a traditional approach is inefficient and consumes time and resources. CE approach has been increasingly used and implemented by enterprise environment in the space industry. This approach ensures that the customer needs are satisfied with the required quality, promoting a reduction of costs and development time. This thesis described a design process developed within an open source Dakota framework with an objective to derive an optimized re-entry vehicle configuration designed for a specific mission scenario and system requirements during the preliminary design phase. This chapter highlights the conclusions drawn from the results of this thesis in Section 7-1. Furthermore, Section 7-2 describes the recommendation; these are intended to assist in future work related to this thesis.

## 7-1 Conclusions

The initial design process of the IXV configuration followed a traditional approach, where engineers worked sequentially passing the design from engineer to engineer and then followed by iterations. This process eventually resulted in being an expensive methodology in-terms of cost, time and resources. Currently, there is a need to provide structured and common design methods involving all the design disciplines at the same time along with engineering data-exchange between the experts. This new design process can be useful for preliminary design analysis of the on-going Spacerider re-entry vehicle, which is derived from the IXV configuration with a payload carrying capacity. Thus, the main objective of the thesis was to derive an optimized re-entry vehicle similar to the IXV configuration during the preliminary design phase, where the vehicle is designed for a specific mission scenario and system requirements, using a design process developed within an open source Dakota framework. To solve this industrial problem within the work of thesis the main research question was framed as follows:

*"To what extent can an optimal re-entry vehicle similar to the IXV configuration be developed in the preliminary design phase using an integrated design process, where the vehicle is designed for a specific mission scenario and system requirements?"*

To answer the main research question, an integrated design process to perform the MDO of a re-entry vehicle is required. Thus, the two other sub-goals are formulated as follows:

*"To perform the feasibility study of the re-entry vehicles using the MDO techniques, to what extent is it possible to integrate the design disciplines on a single platform and automate the design process within the Dakota framework?"*

*"How and to what extent can the design techniques available within the framework, assist the engineering team during the conceptual design of complex systems to obtain better, faster and eventually cheaper design process?"*

To achieve the sub-goals, an integrated design methodology within an open source Dakota framework is developed. Initially, a simple axis-symmetric Raduga capsule is considered to verify the design process, for the re-entry vehicle application and then extended to the lifting body vehicle similar to IXV. This approach allowed the user to reduce the complexities from the vehicle point of view and focus on improving the design process for the feasibility study of the re-entry vehicle application using the MDO techniques. After verifying the design process for optimization of a simple axis-symmetric capsule and for trajectory optimization of the IXV, it is then used to answer the main research question. The conclusions drawn from this study are highlighted as follows.

### 7-1-1 Integration Methodology

Chapter 2 described the system requirements related to the integration methodology developed within the framework. The design process for the feasibility study of a system requires various numerical models to be integrated into a workflow. The results from one numerical model influence the results of the sequential tool's inputs and their response function. Several steps were taken to create a basic workflow, from which a generalized interface template is developed for the integration methodology within the Dakota framework. The design process is able to manage these inputs and the response function through a generalized interface template. Dakota provides a function called dprepro, which is used by the Dakota optimizer to write its design parameter file and read the Dakota results file. To utilize this function for the complete workflow, all the I/O file of each tool are written in the same format, such that dprepro can read/write the interface templates. In this way, the design process developed within the Dakota framework does not need any knowledge regarding the software inside the simulator.

The Dakota shell script is used to define the task performed within the simulator. This file indicates the steps taken to copy the necessary files in the working directories, creating an input file for each tool in the workflow, executing the tools and writing their output files. It also indicates the task to write the overall simulation output file. The shell script only requires the executable file that can run in the loop for the given design parameters. Thus, developers can provide only their executable file without any internal information of the tool, which makes this coupling as a black-box. This allowed the user to accept a black-box tool from a developer as well as allow the developer to upgrade or add any new features to the tool. The user can access the interface during the process to monitor or intervene the simulations. Additionally, the Dakota framework provides visualization infrastructure for monitoring the simulations and manage the data in a tabular format that can be easily exported to the Matlab for post-processing of results. The input file for the Dakota optimizer, allows the user to describe the feasibility study method, using the keywords defined by Dakota, thus gives the user flexibility to choose the necessary methods.

The design process gives freedom to utilize specific software for each discipline and flexibility to choose the level of analysis depending on the accuracy and the cost. The architecture is based on object-oriented approach, where plug and play of all the involved discipline tools are possible. The generalized interface templates allow the efficient communication among the disciplines as well as easy access to input and output files through these templates. This access also provides the fast and efficient debugging capabilities. There is automatic data transfer through the interfaces developed with the possibility of human intervention during in the process. Thus, from Chapter 2, it is concluded that design process satisfies the requirement mentioned in Section 2-2.

### 7-1-2 Design Process for the Re-entry Vehicle Application

This design process was then extended to the re-entry vehicle application using the axis-symmetric Raduga configuration, which is a non-lifting body. As discussed in Chapter 4, various tools developed in Matlab, Simulink, FORTRAN as well as commercial software like Gridgen was successfully integrated into the design process, using the generalized interface template for the optimization of the Raduga-like capsule. The tools can be considered as black-box if required. For Gridgen, a similar interface template is created using the Glyph script, where the dprepro function is used to read/write the co-ordinates of the geometry to automatically generate the mesh. From this verification, it is clear that the design process allowed the user to integrate the design disciplines on the common platform and furthermore automate the design process for better, faster and eventually low cost design process, than a traditional design approach. The generalized interface templates make this approach beneficial in collaborative projects, where different industries are working together, but cannot share their tools or software with each other. In such cases, the developer can only provide the inputs and output parameters with the tool as a black-box. The user can use the generalized interface template for any tool in a similar way, which provides the flexibility to choose the level of analysis depending on the accuracy and the cost and one can utilize a specific software for each discipline. The use of a generalized interface for the design process, also allowed the user to plug and play with all the involved design discipline tools very easily. Additionally, as the tools can be a black-box as well as any up-gradation to the tool is possible.

Along with the verification of the integration methodology, the results also gave an insight of application of MDO techniques for re-entry application. By using the LHS and Spearman correlation formulation the sensitivity analysis is performed as described in Chapter 2-4. From the samples, the partial rank correlation factors between the each design parameter and the cost functions were computed using the Spearman formulation. The correlation between the design parameters and objective function such as payload mass and payload volume gave the insight of which parameters contribute to the results and how much. Furthermore, it gave the optimum condition and the range of the design parameters, where the user can expect the result at the optimum condition. This sensitivity analysis gave the insight of influence of design parameters on the results and is considered as starting point for the optimization using the MOGA. Using this methodology, 30 optimized configurations are derived that satisfy the constraints and meets the objectives to maximize payload carrying capabilities. Based on the user's choice, the best-optimized configuration is selected, which gave an improvement of 4 kg payload mass capacity and 0.03 $m^3$ payload volume capacity, compared to the reference Raduga configuration.

From this chapter, it can be concluded that the MOGA as MDO technique can be used for the re-entry vehicle application. The design process is able to manage a large number of design parameters, equality and inequality constraints considered for the optimization of re-entry vehicle configuration. The framework produces an output file in a tabular format such that visualization of the results is possible using any appropriate graphical tool such as Matlab. It is possible to monitor the MDO process at any phase of the execution. These results answer the second subgoal, where the user can utilize the available design techniques within the Dakota framework to derive optimized solutions. This will assist the engineering team, during the preliminary design of complex systems. Since Dakota framework is an open source framework, the developed design process within the framework is of low cost, more efficient and more robust than the traditional approach for the feasibility study of re-entry vehicles at the preliminary design phase.

To answer the main research question, it was important to perform the geometry as well as the trajectory optimization of a lifting body similar to the IXV configuration. Chapter 5 gave the results of the trajectory optimization of a lifting body re-entry vehicle. Here, the IXV configuration is considered as a reference vehicle. For the given geometry and the aerodynamic database, the trajectory optimization is performed to demonstrate the capability of the design process for the feasibility study of a complex re-entry vehicle such as IXV. MOGA is used as optimization technique. This chapter highlighted the optimized trajectories derived from the design process and its comparison with the nominal trajectory of IXV. Although the nominal trajectories are derived by using sophisticated tools considering the uncertainties and requirements from other design disciplines, which are not considered in this problem formulation, the derived optimized trajectories from the design process are comparable to the nominal ones. These results indicated that the derived optimized trajectory gives improved performance than the initial (open-loop simulation) as well as the nominal reference trajectory of IXV, with respect to the values of the cost functions considered for the trajectory simulation. The best-optimized trajectory gave a landing accuracy of 0.5 km, while satisfying all the constraints.

Once the design process is verified for the optimization of Raduga configuration and the trajectory optimization of the IXV configuration. From these verifications, it is concluded that the design process is flexible to plug and play tools into the toolchain using the generalized interface template. This verification answers the sub-goals of this thesis. Furthermore, this step-by-step verification approach allowed to develop a relatively more generalized design process, that enables one to manage the inputs and the corresponding response function through a generalized interface template.

### 7-1-3 Conclusions from the Design Case Results

The main research question is answered by the results of the design case, where the geometry and the trajectory optimization is performed simultaneously for a lifting body re-entry vehicle, described in Chapter 6. The design case configuration is a payload carrying re-entry vehicle, which is derived from the shape morphing of the baseline IXV configuration. As the design process allows easy plug and play, SMMET is introduced into the workflow along with the tools dedicated for the aerodynamic database computation, trajectory simulation, and aerothermodynamic database computation. Similar to the non-lifting body optimization (Raduga), design samples are generated using the LHS method for the design case problem. Using these samples, the correlations between the input and output and set of the design factors are derived by the Spearman's correlation formulation. This sensitivity analysis provided an idea of the influencing design parameters and the range of the design parameters at which the optimized solution can be obtained. Considering this as a starting point, the optimization is performed using the MOGA, with the major objectives of minimizing the integral of the heat-load, maximizing the payload carrying capacity and the landing accuracy of the re-entry vehicle.

From the optimized solutions, two design cases are selected, which gives payload mass and payload volume higher than 450 kg and 0.71 $m^3$ respectively. Additionally, these design cases give an integral of the heat load, which is less than 300 $MJ/m^2$ and the landing accuracy within 25 km. Two optimized design cases are selected, which gives the payload mass and payload volume capacity of 482 kg, 0.79 $m^3$ and 520 kg, 0.74 $m^3$, respectively. Furthermore, these design cases indicated an improvement in landing accuracy, where the vehicle with a payload capacity of 482 kg reached target landing location with the landing the accuracy of 0.4 km, whereas the design case with payload mass capacity of 520 kg reached within 18 km. The results showed improved compared to the open-loop simulation, which indicates that using this developed design process a better solution can be derived.

This answers the main research question, where an optimized configuration similar to IXV is derived for the given mission scenario and systems requirements. As it is a multi-objective problem more than one solutions are derived and the user can select the configuration based on the requirements. The design case with payload mass capacity of 482 kg with landing accuracy of 0.4 km is considered as a final solution. This design case also gives better trajectory profiles than the other selected design case. The differences in the trajectory solution of the selected design cases also indicate a necessity to refine the results by separately performing the trajectory optimization of fixed configuration and vice-versa.

## 7-2 Recommendations

This section describes the recommendation of all the possible aspects to be further investigated. These suggestion are intended to assist in future work related to this thesis.

- The design process developed within the Dakota framework performed the feasibility study of the Raduga-like ballistic vehicle and the the IXV-like lifting body. Considering the limited time and the resources available, fast, limited-fidelity discipline codes to reduce the complexity of the system and the CPU time necessary for each simulation are used. In this case, simplified models, low-fidelity codes and low-level hypotheses have been applied. Thus more complex project should be investigated, that requires high fidelity tools and more sophisticated numerical models.
- The hypothesis such as angle of attack is considered as constant, only hypersonic flight conditions (Mach $\geq 3$) is considered for aero-thermodynamics analysis. Although the FMST is of sufficient level of fidelity. Thus, more complicated flight conditions should be analyzed as well as the whole trajectory should be examined from the aero-thermal point of view, which may require a more detailed aerodynamic database.
- Furthermore, the total mass and fixed mass of the configuration is considered as constant with variation in the configuration. The process of payload mass and volume computation involves assumptions and use of conventional thumb rules, thus the configuration model can be more complex to compute accurate masses and volume.
- Low-fidelity codes are used for the aero-thermodynamics discipline, thus full-Navier Stokes codes can be introduced in the design process. This will also require improving the necessary interfaces. Furthermore, this also adds the challenge to investigate and introduce the approximation techniques to reduce the computational time of a CFD computation. Subsonic flight conditions can be also analyzed using this high fidelity tools.
- For the structural analysis, Nastran and Hypermesh can be introduced in the design process. Furthermore, reinforcements in structure can be analyzed, using more complex mathematical models such as Abaqus and Ansys.
- During this thesis, the flap sizing and the stability of the re-entry vehicle are not considered, thus using the derived configuration can be considered as a starting point and the stability analysis can be performed using the design process.
- Furthermore, there are chances to improve the design process. As the Dakota framework is an open source framework and new MDO techniques can be introduced in the framework. For this thesis, LHS and MOGA have been tested, where as more sampling methods and optimization methods can be explored by either importing them to the Dakota framework or implementing the developed methodology in the commercial framework such as ISight as it provides wide range of other optimization techniques. Furthermore, commercial or open source framework can be tested.
- This thesis is focused on the MDO of re-entry vehicle, but once the process is verified and tested for the design case such as IXV, it can be used to perform the other activities such as:
  - Robustness assessment of the design.

  - Uncertainty analysis.
  - Detailed sensitivity analysis.
  - Worst case identification.
  - Design reliability assessment.

- In this thesis AAO problem formulation is used, where only Dakota optimizer is responsible for the optimization. The design process can be improved to introduce MDF problem formulation, where one can introduce disciplinary analyzer for each design discipline and system coordinator to handle the interdisciplinary relations to derive a consistent solution.
- In this thesis, although the optimized configuration is successfully derived using the developed design process. These results can be considered as a starting point for detailed analysis. From the results of the design case, it is observed that there are challenges while performing trajectory and the geometry optimization simultaneously and trajectory results of design case 1 requires more refinement. The results of the trajectory and geometry can be further improved by performing only trajectory for the selected optimized configuration or by geometry optimization for a selected trajectory. Additionally, the local optimization by using gradient base methods can be performed to improve the results.

# Bibliography


Adami, A., Mortazavi, M., Nosratollahi, M., and Hosseini, M., "Multidisciplinary Design Optimization of a Manned Re-entry Mission Considering Trajectory and Aerodynamic Configuration," *Recent Advances in Space Technologies (RAST), 2011 5th International Conference*, IEEE, 2011, pp. 598–603.

Adami, A., Mortazavi, M., and Nosratollahi, M., "Multidisciplinary Design Optimization of a Deorbit Maneuver Considering Propulsion, TPS, and trajectory," *International Journal of Computer Applications*, Vol. 116, No. 23, 2015.

Adams, B. M., *Dakota, a Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 5.0 users manual*, 2009.

Aguilar, J. A., Dawdy, A. B., and Law, G. W., "The Aerospace Corporations concept design center," *8th Annual International Symposium of the International Council on Systems Engineering*, Vol. 2, 1998.

Anderson, J. D., *Hypersonic and High Temperature Gas Dynamics*, AIAA, 2000.

Antoine, N. E. and Kroo, I. M., "Framework for Aircraft Conceptual Design and Environmental Performance Studies," *AIAA journal*, Vol. 43, No. 10, 2005, pp. 2100–2109.

Bianco, D., Ambrosio, D. D., and Mareschi, V., "A fully implicit material response code with ablation and pyrolysis for simulation of thermal protection systems," *European Space Agency*, 2015, pp. 1–10.

Chiandussi, G., Codegone, M., Ferrero, S., and Varesio, F. E., "Comparison of Multi-Objective Optimization Methodologies for Engineering Applications," *Computers & Mathematics with Applications*, Vol. 63, No. 5, 2012, pp. 912–942.

Chiarelli, C., "IXV Mass Budget, PHASE C2/D/E1, Extract INFO, DRL/DRD: SYS051," Thales Alenia Space, Technical internal document, 2014.

CoRADS Development team, *Modified Code for Rapid Aerodynamic Database Synthesis, Graphical User Interface utilization GUIDE*, Thales Alenia Space, Italy, 2016.

Dufour, R., de Muelenaere, J., and Elham, A., "Trajectory Driven Multidisciplinary Design Optimization of a Sub-Orbital Spaceplane using Non-Stationary Gaussian Process," *Structural and Multidisciplinary Optimization*, Vol. 52, No. 4, 2015, pp. 755–771.

Gang, C., Min, X., Zi-ming, W., and Si-Lu, C., "Multidisciplinary Design Optimization of RLV Re-entry Trajectory," *13th AIAA/CIRA International Space Planes and Hypersonic Systems and Technologies. Capua, Italy*, 2005.

Grossman, B., Gurdal, Z., Haftka, R., Strauch, G., and Eppard, W., "Integrated Aerodynamic/Structural Design of a Sailplane Wing," *Journal of Aircraft*, Vol. 25, No. 9, 1988, pp. 855–860.

Haftka, R. T., Starnes Jr, J. H., Barton, F. W., and Dixon, S. C., "Comparison of Two Types of Structural Optimization Procedures For Flutter Requirements," *AIAA Journal*, Vol. 13, No. 10, 1975, pp. 1333–1339.

Hammond, W. E., *Design Methodologies For Space Transportation Systems*, AIAA, 2001.

Hassan, R., Cohanim, B., De Weck, O., and Venter, G., "A comparison of particle swarm optimization and the genetic algorithm," *Proceedings of the 1st AIAA multidisciplinary design optimization specialist conference*, 2005, pp. 18–21.

Henderson, R. P., Martins, J., and Perez, R. E., "Aircraft Conceptual Design for Optimal Environmental Performance," *Aeronautical Journal*, Vol. 116, No. 1175, 2012, pp. 1–22.

Krevor, Z., Howard, R., Mosher, T., and Scott, K., "Dream chaser commercial crewed spacecraft overview," *17th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, AIAA, 2011, pp. 22–45.

Larson, V., *Human Spacefligt: Mission Analysis and Design*, McGraw-Hill, 1999.

Larson, W. J. and Wertz, J. R., "Space mission analysis and design," Microcosm, Inc., Torrance, CA (US), Tech. rep., 1992.

Lavagna, M., Parigini, C., and Armellin, R., "PSO algorithm for planetary atmosphere entry vehicles multidisciplinary guidance design," *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2012, pp. 27–60.

Legostaev, V. and Minenko, N., *The Returned Ballistic Capsule (Rainbow)*, Scientific Production Association, 1994.

Liang, Z., Ren, Z., and Bai, C., "Lateral Reentry Guidance for Maneuver Glide Vehicles with Geographic Constraints," *Control Conference (CCC), 2013 32nd Chinese*, IEEE, 2013, pp. 5187–5192.

Mareschi, V., "CAST-MDO Project, Feasibility Study on Multi Disciplinary Optimization for Aerospace Application, TASI/CAST/DT-62," Thales Alenia Space, Technical internal document, 2014.

Marler, A., "Survey of Multi-objective optimisation methods for engineering," *Struct Multidisc Optim, Springer-Verlag*, Vol. 26, 2004.

Nosratollahi, M., Mortazavi, M., Adami, A., and Hosseini, M., "Multidisciplinary Design Optimization of a Re-entry Vehicle using Genetic Algorithm," *Aircraft Engineering and Aerospace Technology*, Vol. 82, No. 3, 2010, pp. 194–203.

Priyadarshi, P. and Mittal, S., "Multi-objective Multi-disciplinary Design Optimization of a Semi-Ballistic Reentry Module," *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, AIAA 2010-9127*, 2010, pp. 1–10.

Rasmussen, C. E., "Gaussian Processes for Machine Learning," 2006.

Riccardi, A., *Multidisciplinary Design Optimization for Space Applications*, Ph.D. thesis, University of Bremen, 2012.

Ridolfi, *Space Systems Conceptual Design Analysis Methods for Engineering-Team Support*, Ph.D. thesis, Politecnico di Torino-Delft University of Technology, 2013.

Rodrigo, H. R., Victor, M., and Murray, K., "IXV GNC Verification from Inspection to Flight Demonstration," *6th International Conference on Astrodynamics tools and Techniques (ICATT)*, 2016.

Rufolo, M., "SPACE-RIDER the Reusable Orbital Re-entry Vehicle for Europe," *2nd CESMA Hypersonic flight Symposium, Rome, Italy*, Vol. 2, 2016.

Santilli, F. and Sudars, M., "IXV PHASE C2/D/E1, TPM ARCHITECTURE, DRL/DRD: SYS050," Thales Alenia Space, Technical internal document, 2014.

Schmit, L. A., "Structural Design by Systematic Synthesis," *Proc. of the Second ASCE Conference on Electronic Computation*, 1960, pp. 105–122.

Schmit, L. A. and Thornton, W. A., *Synthesis of an Air-Foil at Supersonic Mach Number*, National Aeronautics and Space Administration, 1965.

SpaceflightInsider, "Different vehicles for different purposes: Orion and the Commercial Crew contenders," http://www.spaceflightinsider.com/organizations/nasa/different-vehicles-different-purposes-orion-commercial-crew-contenders/, [Online; accessed 16-May-2016], 2016.

SPARTA Development Core Team, *SPARTA, Users Manual 8 Sep 2015 Version*, Sandia National Laboratories, USA, 2014.

Steenhuizen, E., Rocca, "Decomposition and Coordination Multidisciplinary Optimization Strategies Flight Performance and Propulsion Group Advanced Design Methods," 2015.

Sudars, Martin, *Code for Rapid Aerodynamic Database Synthesis, Graphical User Interface utilization GUIDE*, Thales Alenia Space, Italy, 2016a.

Sudars, Martin, *Flight Mechanics Simulation Tool, Graphical User Interface utilization GUIDE*, Thales Alenia Space, Italy, 2016b.

Tröltzsch, A., Siggel, M., Kopp, A., and Schwanekamp, T., "Multidisciplinary Analysis of the DLR Spaceliner Concept by Different Optimization Techniques," *Minisymposium on Structural and Multidisciplinary Optimization at the World Congress on Computational Mechanics 2014, Barcelona, Spain*, 2014, p. 1.

Tsuchiya, T., Takenaka, Y., and Taguchi, H., "Multidisciplinary Design Optimization for Hypersonic Experimental Vehicle," *AIAA journal*, Vol. 45, No. 7, 2007, pp. 1655–1662.

Van Eekelen, T., Lachaud, J. R., Martin, A., and Cozmuta, I., "Ablation Test-Case Series# 3. Numerical Simulation of Ablative-Material Response: Code and Model Comparisons," *7th European Workshop on Thermal Protection Systems and Hot Structures*, 2013, pp. 1–18.

Xu, D., Bil, C., and Cai, G., "A CDF framework for aerospace engineering education," *Journal of Aerospace Operations*, Vol. 4, No. 1-2, 2016, pp. 67–84.

Yokoyama, N., "Trajectory Optimization of Space Plane Using Genetic Algorithm Combined with Gradient Method," *23rd International Congress of Aeronautical Sciences, Toronto, Canada*, Vol. 513, 2002, pp. 1–10.

Yokoyama, N., "Multidisciplinary Optimization of Space Plane Modeled as Rigid Body," Vol. 44, 2004, pp. 1–10.

Yokoyama, N., Suzuki, S., and Tsuchiya, T., "Study on trajectory optimization of space plane ascending to ISS," *10th AIAA/NAL-NASDA-ISAS International Space Planes and Hypersonic Systems and Technologies Conference*, 2001, pp. 18–29.

Yokoyama, N., Suzuki, S., Tsuchiya, T., Taguchi, H., and Kanda, T., "Multidisciplinary Design Optimization of Space Plane Considering Rigid Body Characteristics," *Journal of Spacecraft and Rockets*, Vol. 44, No. 1, 2007, pp. 121–131.