



Design Space Exploration of Valid Generic Waveriders Using Single and Multi-Objective Optimization

Jade Nassif¹, Jimmy-John O.E. Hoste²

Abstract

Waveriders are typically designed through streamline-tracing methods starting from a pre-defined rear vehicle parametrization. It is possible that this inverse design method does not result in a valid waverider shape which makes it difficult to perform design space exploration in an optimization framework with specific objective functions. Instead, a direct methodology, such as the one introduced by Son *et al.* (Aerospace, Vol 9, Issue 7, 2022, 348), ensuring valid designs can be consistently obtained, is needed. This work builds further on the parametrization of Son *et al.* by considering various objective functions. The optimization is performed with a genetic algorithm together with a surrogate model for which the aerodynamic performance is evaluated through Computational Fluid Dynamics simulations.

Keywords: *CFD, waveriders, optimization, Kriging*

1. Introduction

Hypersonic vehicles have been a subject of interest in the past century with the first manned atmospheric hypersonic flight achieved in the X-15 program in 1961 [1]. They could enable access-to-space in single-stage-to-orbit (SSTO) space planes [2] or reduce significantly intercontinental flight times. Accomplishing this is a technological challenge, as exemplified by the National Aerospace Plane (NASP) program back in the late 1980s. Among the candidate vehicles for hypersonic flight, waveriders concepts [3, 4] offer increased flight range [5] by increasing the lift-to-drag ratio (L/D) via shock waves and were originally proposed by Nonweiler in 1959 [6]. Several waverider geometries have been experimentally studied and flight tested over the past two decades including the X-43 [7], the X-51 [8], the HTV-2 [9], various HIFiRE [10] configurations, HEXAFLY-INT [11] and the GHGV-2 [12]. Waveriders can be generated with either a direct method (parametrized geometry, e.g. [13, 14, 15, 16, 17]) or through inverse methods (streamline-tracing). Inverse-design methods are unable to assess *a priori* the desired aerodynamic performance as well as obtain geometrical information such as internal volume [17]. Therefore, Son *et al.* [17], building on the works of Kontogiannis *et al.* [16] and Takashima and Lewis [13], proposed a direct design method that parametrizes valid geometries which are finally obtained via the osculating cone method [18]. Regardless of the specific inverse-design of choice, the approach remains unchanged as detailed in [19, 17, 20].

Once the geometry is generated, the geometrical information can be assessed and the aero-thermodynamic performance can be obtained via impact methods [21, 22]. However, the general use of Computational Fluid Dynamics (CFD) has become more affordable and widespread. Inviscid simulations are particularly popular and these can be augmented through viscous corrections [23, 12, 24, 25, 26, 27] instead of solving the more costly turbulent problem via the Reynolds-Averaged Navier-Stokes (RANS) equations.

This work builds further on the work of Son *et al.* [17] by considering various single-and multi-objective functions to further explore potential waverider shapes. Mach 5 (M5) and Mach 8 (M8) flow conditions are selected at an altitude of 25 and 30 km, respectively. A genetic algorithm based optimization framework is considered which uses a surrogate aerodynamic model. The aerodynamic coefficients used to create this latter model are obtained via the commercial software Simcenter Star-CCM+ [28].

¹Bansbury, UK, Currently: Engineer at Haas F1 Team

²Destinus SA, Switzerland, Aerothermodynamics Engineer, jimmyjohn.hoste@destinus.ch

As part of this work, an open-source Python software for the generation of the hypersonic waveriders has been developed [29, 30]¹ with the aim to boost research in the field, often limited by the unavailability of geometry generation tools. Section 2 outlines the current computational framework followed by a discussion of results in 3 for several objective functions. Conclusions are drawn in 4.

2. Problem Description

Figure 1 presents the computational framework adopted in this work. An initial design space sampling (see 2.3.1) of the selected design parameters (see 2.1.3) is performed which is used to obtain a set of waverider geometries via an open-source generator (see 2.1.2). The computational mesh and setup for CFD analysis is subsequently performed (see 2.2.1) and the CFD simulations launched. Quantities of interest from the CFD runs are extracted (see 2.1.4) and surrogate models for each are build (see 2.3.2). These models are then used for single-objective (see 2.4.2) and multi-objective (see 2.4.3) shape optimisation studies with a genetic algorithm (see 2.4.1). Furthermore, a feature analysis of the design space inputs on the extracted quantities of interest is performed and discussed in the Results Section 3.1. A comprehensive overview is readily available in [30].

The Methods Section is further structured as follows. The direct parametrisation method is introduced in Sub-section 2.1. The CFD strategy and the associated surrogate modelling are detailed in 2.2 and 2.3, respectively. Finally, the optimisation algorithm is covered in 2.4.

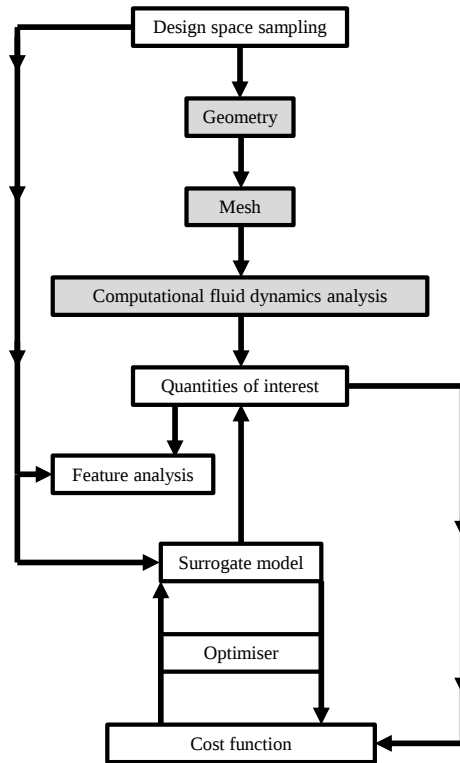


Fig 1. Schematic of the proposed waverider optimisation framework inspired by the work of Son *et al.* [17]. The grey boxes are part of the flow field analysis whereas other steps are needed to tackle the optimisation problem.

2.1. Waverider Parametrisation

2.1.1. Base Plane Parametrisation

In this work, the parametrisation by Son *et al.* [17] is adopted to parametrise the base plane of the waverider (see Figure 5 of [30] for overview of typical waverider nomenclature). The technique makes

¹https://github.com/jade5638/waverider_generator

use of four geometric parameters to define the upper surface curve (USC) and the shockwave curve (SC) of the waverider. This method consists of a small number of design variables while resulting in a wide variety of shapes (see Figure 7 of [17]), making it an efficient parametrisation method for optimisation. Figure 2 describes the parametrisation scheme accounting for symmetry. A local two-dimensional coordinate system is set up such that $y' = y + h$, $z' = z$ and w represents the half-width of the waverider.

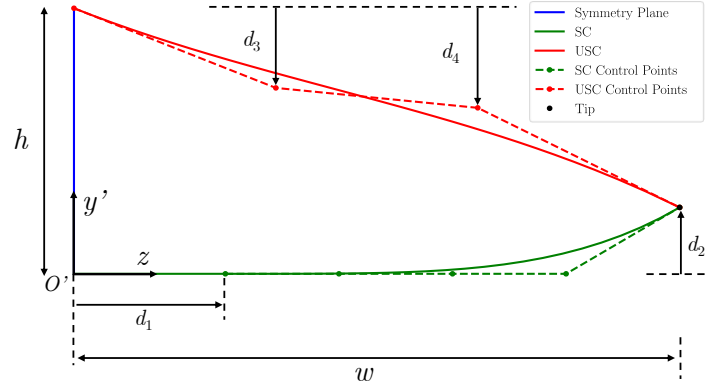


Fig 2. Parametrisation of the Base Plane

The SC is composed of a flat region, defined by the distance d_1 , and a curved region following it. The curved region is defined as a fourth order Bézier curve with five control points. These are equally spaced in z and lie on the line $y' = 0$, except for the rightmost whose vertical position is determined by the distance d_2 . Furthermore, the USC is defined as a third order Bézier curve with four control points. These are also equally spaced in z , while the vertical positions of the two internal points are determined by the distances d_3 and d_4 . These distances are then normalised as follows

$$\left\{ \begin{array}{lll} X1 = \frac{d_1}{w} & \text{with } d_1 \in [0, w[& \rightarrow X1 \in [0, 1[\\ X2 = \frac{d_2}{h} & \text{with } d_2 \in [0, h] & \rightarrow X2 \in [0, 1] \\ X3 = \frac{d_3}{(h - d_2)} & \text{with } d_3 \in [0, h - d_2] & \rightarrow X3 \in [0, 1] \\ X4 = \frac{d_4}{(h - d_2)} & \text{with } d_4 \in [0, h - d_2] & \rightarrow X4 \in [0, 1] \end{array} \right. \quad (1)$$

which gives the final four geometric parameters $[X1, X2, X3, X4]$.

Sobieczky's [18] osculating cone method (see also [31] for a recent review) is then applied on the Base Plane to obtain the remainder of the waverider geometry.

2.1.2. Waverider Generator

The open-source "Waverider Generator" [29] Python library complements ensures the reproducibility of the geometry generation by integrating the Osculating Cone Theory (OCT) and the aforementioned parametrisation. It was used to produce all geometries in this work. A comprehensive description of the input geometrical and flow parameters is provided in [30].

2.1.3. Design Space Inputs

Son *et al.* [17] made use of the geometric parameters X1 to X4 to define a design space for optimisation as

$$\mathbf{V} = \begin{cases} X1 \in [0, 1[\\ X2 \in [0, 1] \\ X3 \in [0, 1] \\ X4 \in [0, 1] \end{cases} \quad \text{with respect to GC : } \frac{X2}{(1 - X1)^4} < \frac{7}{64} \left(\frac{w}{h}\right)^4. \quad (2)$$

Here $\mathbf{V}$ is the design parameter vector and GC refers to a Geometric Constraint. The rest of the parameters including h , w , M_{design} , and the β shock angle are kept constant.

The Geometric Constraint (GC) in Equation (2) originating from OCT ensures that the inverse design method does not fail. Specifically, the OCT requires the intersection between any two osculating planes to be above the USC [17]. As such, the GC dynamically restricts the maximum value that X2 can take based on the value of X1.

The same design space will be used in this study, with the addition of the design Mach number M_{design} as a design variable. This is because the optimisation problems considered here are multi-point and aim to optimise the performance of the waverider across two flight conditions. The following paragraph details the rationale behind the inclusion of M_{design} in the design space.

Mach Number & Shockwave Considerations

A waverider designed for Mach 5 will not maintain the design shockwave shape at Mach 8 and vice versa. For this reason, in this optimisation framework, the influence of M_{design} on the shockwave shape is considered. It is an inherent part of the wide-speed problem. Figure 3 shows the shock angle at the symmetry plane, denoted as β_0 , against M_{design} for both flight conditions. Here, the design shock angle is $\beta = 15^\circ$. The curves are obtained by making use of the oblique shock relations ($\beta - \theta - M$ relation).

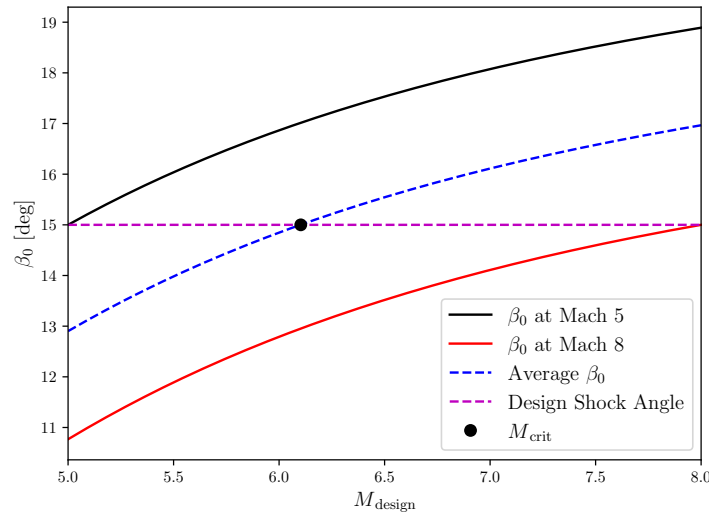


Fig 3. Shock angle at symmetry plane against M_{design}

At Mach 5, a waverider designed for that flight condition will have $\beta_0 = \beta = 15^\circ$. Similarly, at Mach 8, a waverider designed for that flight condition will display $\beta_0 = \beta = 15^\circ$. However, when the freestream Mach number $M_\infty \neq M_{\text{design}}$, the resulting β_0 is either higher (Mach 5) or lower (Mach 8) than the target. For simplicity and consistency across a range of waverider shapes, the change in the shockwave shape can therefore be quantified as:

$$\Delta\beta = |\beta_0 - \beta| \quad (3)$$

where β is the target shock angle. At $M_{\text{design}} \approx 6.1$, $\Delta\beta$ is equal for both flight conditions, and this can serve as an important point to compromise between the two design conditions. Furthermore,

it's important to note that M_{design} 's contribution's to the geometrical properties of the waverider is an increase in the deflection angle with an increase in M_{design} , as predicted by the $\beta - \theta - M$ relation. This translates to an increase in the thickness of the waverider for a specific shockwave shape. An example is shown in Figure 4, where this phenomenon is clearly visible:

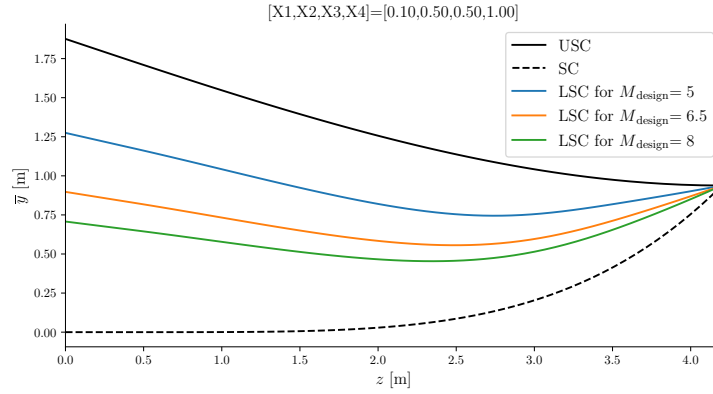


Fig 4. Effect of M_{design} on the lower surface of the waverider. View from the base plane. For $\beta=15^\circ$, $h = 1.876$ m and $w = 4.2$ m.

Presently, M_{design} is added to Son *et. al*'s [17] design space to give the final design space used in this study:

$$\mathbf{X} = \begin{cases} M_{\text{design}} & \in [5, 8] \\ X1 & \in [0, 1[\\ X2 & \in [0, 1] \\ X3 & \in [0, 1] \\ X4 & \in [0, 1] \end{cases} \quad \text{with respect to} \quad \begin{cases} \text{GC : } \frac{X2}{(1-X1)^4} < \frac{7}{64} \left(\frac{w}{h}\right)^4 \\ \beta = 15^\circ \\ h = 1.876 \text{ m} \\ w = 4.2 \text{ m} \end{cases} \quad (4)$$

where $\mathbf{X}$ is the vector of design variables. Including the design Mach number in this parameter set does allow to consider waverider geometries optimized for another Mach number than the ones of interest (5 and 8 in our case) and therefore increases the pool of potential shapes. The above set of parameters leads to a vehicle length $l = h / \tan(\beta) \approx 7$ m which is chosen to be similar to the X-51 [8] that is 7.5 m long.

2.1.4. Design Space Outputs

Having established the inputs of the design space, many outputs of interest can be obtained, some of which are used in the optimisation problems. A database containing these outputs for 99 points across the design space was created and stored in this project's GitHub Repository ². The full sampling strategy is outlined later in Sub-Section 2.3. Table 1 lists the main outputs considered in this study and the methods by which these were obtained. Note that, for convenience, all forces are those acting on only half the geometry as a symmetry plane boundary condition is applied in the CFD setup. Moreover the Lift and Drag are obtained at angle of attack α of 0° representing cruise condition and this is the case throughout the optimisation. Furthermore, the subscripts "M5" and "M8" will be used hereafter to denote forces at the M5 and M8 flow conditions respectively. On the other hand, geometric properties such as V , S_{wet} and v_{eff} are obtained on the entire geometry through *cadquery* [32], which is a Python library for handling and creating CAD models.

2.2. Performance Estimation

As shown in Table 1, the aerodynamic performance is obtained via CFD. This section describes the CFD setup used to solve the inviscid flow around a waverider with the Euler Equations. In this case, the flow is solved in a steady-state to represent cruising condition and minimise computational cost.

²https://github.com/jade5638/jade_nassif_thesis

Table 1. Outputs of the design space

Symbol	Units	Output Description	Method
D	N	Inviscid Drag force	Inviscid CFD
L	N	Inviscid Lift force	Inviscid CFD
(L/D)	[-]	Inviscid Lift to Drag ratio	Inviscid CFD
D^{visc}	N	Viscous Drag force	Viscous Drag Corrections
V	m ³	Internal Volume	cadquery [32]
S_{wet}	m ²	Wetted Surface Area	cadquery [32]
v_{eff}	[-]	Volumetric Efficiency	$V^{2/3}/S_{\text{wet}}$

The commercial CFD solver Simcenter STAR-CCM+ 19.02.009-R8 [28] was used to conduct all CFD simulations in this work, owing to its compact user interface which handles all stages of the simulation from meshing to post-processing. This feature, along with the use of Java macros, provides a significant advantage in terms of workflow automation.

2.2.1. Computational Domain and Mesh

The same computational domain, illustrated in [30], was kept for all inviscid CFD simulations. As is commonly done in the literature [17, 33, 34], the wake behind the waverider is not included in the domain due to the lack of viscosity which prevents an accurate representation of the wake and can lead to convergence issues. Furthermore, only half the body is included making use of the symmetry. The mesh is unstructured and generated using Polyhedral elements with the build-in STAR-CCM+ automated mesher. Further details about the metrics used to obtain the base grids can be found in [30]. The shockwave is the main feature of the flow and is driving the lift and the drag characteristics. As such, Adaptive Mesh Refinement (AMR) was included in the CFD setup to ensure the sharp gradients around the shock are well captured and reasonable estimates of lift and drag are reached. The criterion chosen is shown in Equation (5):

$$C = \|\nabla M(x, y, z)\| \times s = \sqrt{\left(\frac{\partial M}{\partial x}\right)^2 + \left(\frac{\partial M}{\partial y}\right)^2 + \left(\frac{\partial M}{\partial z}\right)^2} \times s \quad (5)$$

where $M(x, y, z)$ is the local Mach number represented as a scalar field across the domain and s is the Adaptation Cell Size (ACS), calculated as "twice the maximum distance between a cell centroid and any of the cell vertices" in Star-CCM+ [28]. The refinement can take place when $C \notin [0, 0.1]$, with a maximum refinement level of 2. Furthermore, the minimum Face Validity was set at 0.9 to prevent bad quality cells in the refined regions. Additionally, only cells with an ACS greater than 0.001 m were eligible for refinement. AMR was set to take place every 100 iterations to allow the solution on the current mesh to reach a realistic steady-state convergence before being refined. A number of refinements n_{ref} is also predefined to serve as a minimum number of iterations before considering the convergence criterion. For example, if $n_{\text{ref}} = 2$, then a minimum of $2 \times 100 = 200$ iterations is set before checking for convergence. Figure 5 showcases this AMR setup after 3 refinements at $\alpha = 0^\circ$ for the M5 flow condition. The criterion is suitable for this type of flow as it successfully refines the region of the shock and the overflow to the upper surface. A grid convergence study is discussed in [30] for both the M5 and M8 cases.

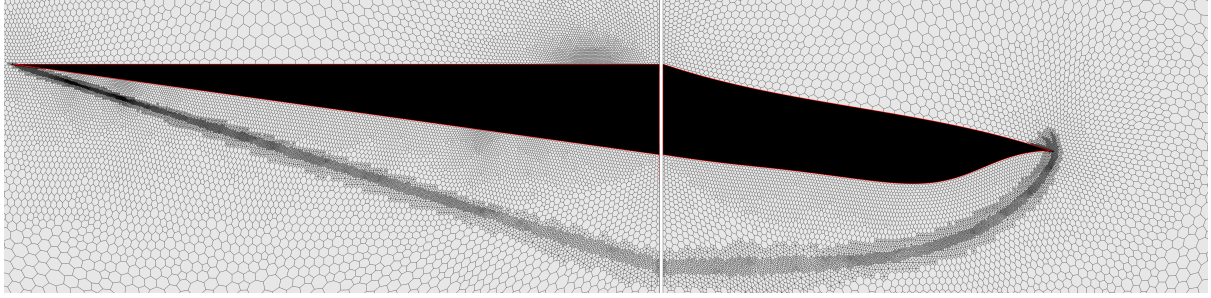


Fig 5. AMR on the symmetry plane (left) and back plane (right) with 3 refinement steps.

2.2.2. Numerical Solver Configuration

Inviscid simulations were used to obtain the aerodynamic coefficients. The Coupled Implicit solver was selected where the equations get solved simultaneously through pseudo-time marching [28, 35]. Moreover, the Advection Upstream Splitting Method (AUSM) + Flux Vector Splitting (FVS) [36] scheme was chosen to evaluate inviscid fluxes. In addition to this, the Courant–Friedrichs–Lewy (CFL) number was determined through the Automatic Control method built into Star-CCM+ [28]. Furthermore, the Incomplete Lower Upper (ILU) [37] relaxation scheme was selected for the Algebraic Multigrid (AMG) Linear Solver. A third order Monotonic Upstream-centered Scheme for Conservation Laws (MUSCL) scheme with central differencing was selected in conjunction with the Venkatakrisnan [38] slope limiter. The scheme results in 3rd order accuracy except around regions with strong gradients, where the limiter reduces the accuracy to 2nd order [35]. Given the hypersonic nature of the flow, the air cannot be assumed to be an ideal gas [39, 35]. For this reason, a Real Gas model known as Equilibrium Air [40], recommended by Cross *et al.* [35] and the Star-CCM+ user guide [28] was chosen.

2.2.3. Convergence

The lift to drag ratio L/D was chosen as a convergence criterion. This is because lift and drag are the two quantities which are obtained from the CFD setup and used in the optimisation problems, described in Sub-section 2.4. The solution is considered converged if

$$|(L/D)_{\max} - (L/D)_{\min}| < \epsilon \quad (6)$$

Where $(L/D)_{\max}$ and $(L/D)_{\min}$ are the maximum and minimum values of L/D over 40 sequential samples, and $\epsilon = 1 \times 10^{-6}$. A maximum number of iterations was also set at 1000. In the event that convergence is not reached, the behavior of L/D is observed. If the quantity seems to oscillate about a certain value but does not settle enough to reach the established criterion, then the solution is still deemed valid and the last value of L/D is recorded.

2.2.4. Flow Conditions

The multi-objective optimisation problems consider two flow conditions:

1. Mach 5 at 25 km altitude (M5)
2. Mach 8 at 30 km altitude (M8)

The two flow conditions are summarised in Table 2:

Table 2. Flow conditions adopted in optimisation work

	M5 at 25 km alt.	M8 at 30 km alt.
p_{∞} [Pa]	2506.0	1169.0
U_{∞} [m.s ⁻¹]	1490.0	2416.0
T_{∞} [K]	221.65	226.65
ρ_{∞} [kg.m ⁻³]	3.940×10^{-2}	1.797×10^{-2}

2.3. Surrogate Modelling

The first step in building a surrogate model is to determine a set of sample points to accurately represent the design space and which will be used to train the model. As such, 2.3.1 focuses on the sampling methodology while 2.3.2 focuses on the training and validation of the surrogate models used in optimisation through a Kriging approach. The Surrogate Modeling Toolbox (SMT) Python library [41] was chosen to conduct all the steps outlined in these sections, as it provides a wide range of tools and options to construct surrogate models.

2.3.1. Sampling

The Latin Hypercube Sampling method [42] was selected to sample the design space due to its numerous advantages over classic random sampling (see detailed discussion in 2.5.1 of [30]). In the work by Son *et al.* [17], which considers the same design space with the exception of the design Mach number, a final set of 60 points were used to train the surrogate models for volume, C_D and C_L . However, the latter two displayed poor accuracy in some regions, which could be partly due to an insufficient number of points. For this reason, and given the addition of M_{design} in the design space, a target was set to sample 100 points.

Due to the Geometric Constraint (see Section 2.1) which dynamically restricts X2, 500 samples were initially obtained via Latin Hypercube Sampling and these were then filtered to only keep the ones which satisfy the constraint, resulting in 183 valid points. The K-Means clustering algorithm, as implemented in the Scikit-learn Python package [43], was then applied to select 100 representative points from this set. Each sampling point i , representing a waverider, will be referred to as Waverider i . Waverider 63 was omitted from the set due to meshing issues, leading to 99 final sample points. Slices of the sampling across the design space are shown in Figure 16 of [30].

2.3.2. Kriging Surrogate Model

An ordinary Kriging model was built for each of the following quantities: V (Internal Volume), S_{wet} (Wetted Area), L_{M5} (Lift at M5), L_{M8} (Lift at M8), D_{M5} (Drag at M5) and D_{M8} (Drag at M8). The samples obtained via Latin Hypercube Sampling (see Section 2.3.1) were first separated into a testing (10%) and a training (90%) dataset. The KRG class of the SMT Python package [41] was then used to train the surrogate models, each instance of this class representing an individual surrogate model. The models were then evaluated on testing data with very satisfactory predictive behavior. For further details, the reader is referred to [30].

2.4. Shape Optimisation

This section outlines the optimisation approach taken and the different cases considered. The implementation of the Genetic Algorithm (GA) used in the optimisation is first covered in 2.4.1. Single objective unconstrained test cases are then presented in 2.4.2, the objectives of these being to:

- evaluate the performance of the surrogate model and optimisation algorithm on simple cases.
- identify redundant optimisation objectives.

Five multi-objective optimisation cases are finally presented in Nassif [30] of which one is discussed in 2.4.3. Note that across the optimisation cases, the cost function may be normalised with respect to a reference value for easier comparison across these cases. These values ($\bar{D}_{M5}, \bar{D}_{M8}, \bar{L}_{M5}, \bar{L}_{M8}$) are obtained as the averages across the database of samples and are summarised in Table 10 of [30].

2.4.1. Genetic Algorithm

Genetic algorithms (GA) [44], which mimic the process of natural selection, evolve a set of solutions over many generations. This makes them suitable for multi-objective problems with conflicting objectives, where a set of optimal trade-off solutions is more relevant than a single optimal solution. At every generation, "genetic" operations are applied on "parent" solutions to obtain new "off-spring" solutions based on their performance (fitness value). This allows for a thorough exploration of the design space. The GA of the Python library PyGAD [45] was used in this work, owing to its flexibility and extensive documentation. It is worth noting that PyGAD maximises the solution, and so this may be reflected in the way the optimisation problems are tackled and presented hereafter.

Overall Setup GAs are inherently random due to their stochastic processes. As such, the optimisation was run more than once to ensure robustness and consistency in the results. In multi-objective problems, after each run, the set of Pareto-optimal (non-dominated) solutions from the last generation was recorded. Once all the runs were performed, the set of non-dominated solutions across all runs was computed to give the final Pareto Front. Further details about exact parameter settings are provided in Table 5 of [30].

Constraint Evaluation Given that GAs cannot directly evaluate constraints, it was decided to indirectly account for broken constraints via a penalty approach with details found in [30]. Furthermore, to favor geometrically valid solutions, it was decided to attribute a fitness value of 0 whenever the Geometric Constraint (GC) is broken. This would prioritise valid solutions and minimise the number of non-valid solutions present at every generation.

2.4.2. Single-objective Test Cases

A total of six single objective optimisation test cases were explored as part of the original studies as explained in Table 6 of [30]. In this work, two of the objectives (see Table 3) are considered for detailed discussions.

Table 3. Test cases considered

Test Case	Objective	Cost Function
1	Maximise Volume	$C = V$
2	Maximise L_{M5}	$C = L_{M5}/\bar{L}_{M5}$

2.4.3. Multi-objective Optimisation Cases

In the original work [30], five multi-objective optimisation cases were considered. In order to limit the scope of the current discussions, only one is retained as given by Table 4. It is representative of the general capabilities of the approach and paints a global picture of the achievable optimal designs. The reader is referred to Section 2.6.3 of [30] for a complete overview.

Table 4. Description of the Multi-objective Cases

Case	Objectives	Constraints
1	Maximise Volume Minimise Inviscid Drag	None

Case 1

Case 1 may be presented as:

$$\text{Maximise } \begin{cases} V \\ 1/D_{\text{avg}} \end{cases}$$

where:

$$D_{\text{avg}} = \frac{0.5D_{M5} + 0.5D_{M8}}{0.5\bar{D}_{M5} + 0.5\bar{D}_{M8}} \quad (7)$$

3. Results and Discussion

3.1. Feature Analysis of Design Space Sampling

Son *et al.* [17] made use of Analysis of Variance (ANOVA) to determine the influence of the inputs on the outputs of the design space. However, ANOVA assumes independence between the design variables.

As discussed in [30], interdependencies do exist between the design variables, and this can lead to inaccuracies in the outcome of ANOVA. In this work, another approach was taken by training Random Forest Models (RFMs) and extracting the Feature Importances. In a RFM, the importance of a feature directly reflects how crucial this feature is in correctly predicting the output, as it is calculated based on the feature's contribution to minimising error across the ensemble of decision trees [46]. Moreover, RFMs do not assume independence and are able to model complex relationships, making them a more suitable and flexible alternative to ANOVA. These models were trained solely for this purpose via the Scikit-learn Python package [43] with 1000 trees.

The feature importances are shown in Figure 6 for M_5 . The M_8 results do provide the same insight and are therefore not discussed here.

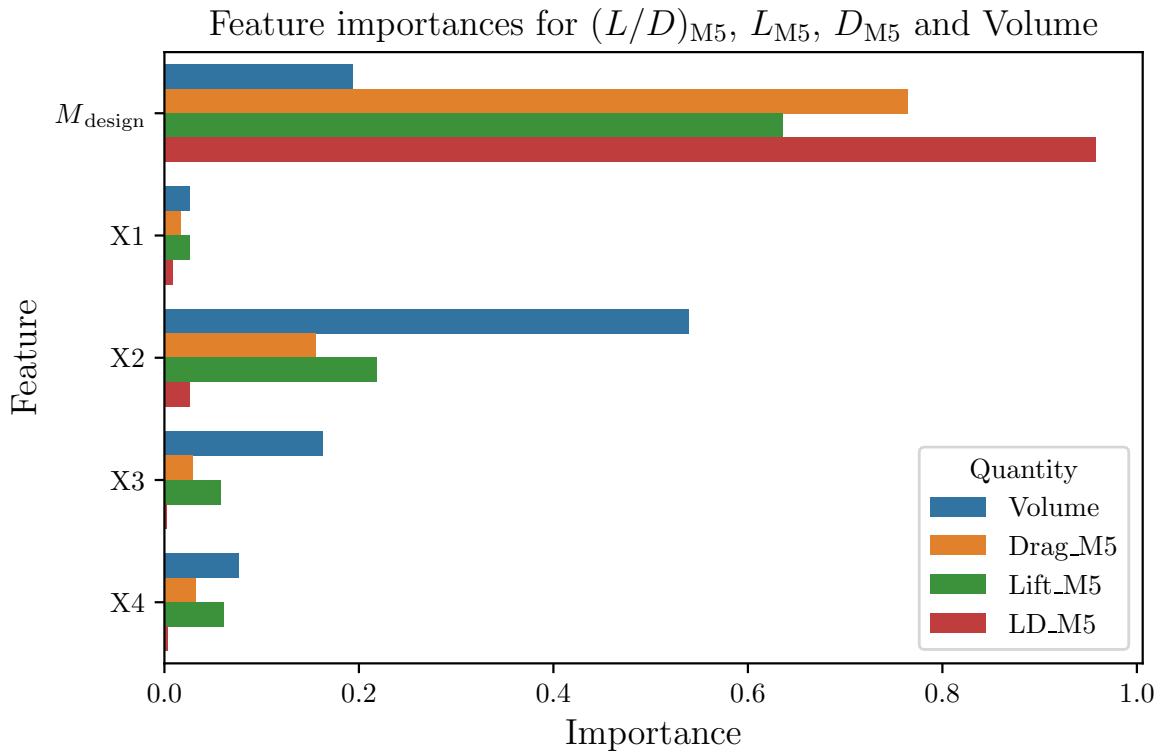


Fig 6. Feature importances for for $(L/D)_{M5}$, L_{M5} , D_{M5} and Volume

The $(L/D)_{M5}$ is clearly dominated by M_{design} with the other variables playing little to no importance in determining the ratio. Looking more closely as the available sample points, M_{design} generally leads to a decrease in $(L/D)_{M5}$ (see also Figure 22a of [30]). The relationship is non-linear and appears to “cap” the maximum lift to drag ratio achievable as a function of M_{design} . Similarly, L_{M5} and D_{M5} are mainly driven by M_{design} . From Figure 7, an increase in M_{design} generally leads to an increase in L_{M5} and the same is shown in Figure 22c of [30] for D_{M5} . However, the geometric parameters have gained more importance than with $(L/D)_{M5}$, especially in the prediction of lift. Furthermore, in the paper by Son *et al.* [17, Figure 10], X3 has significantly more influence on lift and drag compared to X4. Here, their importances are similar, and this may be a result of one, or a combination of (a) Larger number of samples, (b) Introduction of M_{design} to the design space, (c) Use of a RFM instead of ANOVA. When it comes to Volume, X2 has the most influence on the output and is in line with the findings of Son *et al.* [17]. This is because X2 controls the curvature of the SC, which determines the location of the center of the local osculating cone and therefore influences the flowfield the most [17]. As seen in Figure 7, a higher value of X2 tends to result in a higher volume. This is in line with the findings of Son *et al.*

[17] where the maximum volume waveriders found in the optimisation neared $X_2 = 1$. Moreover, X_3 is more important than X_4 , which is the opposite in [17]. Once again, this may be attributed to one of the reasons mentioned above.

Overall, the introduction of M_{design} to the design space reveals the major role it plays in determining key aerodynamic properties like L/D , lift and drag. This is because, for a combination X_1 to X_4 (i.e specific shockwave and upper surface shapes), M_{design} determines the upper and lower bounds of volume. Indeed, as mentioned in Section 2.1.3, an increase in M_{design} results in an increase in the deflection angle, which increases overall volume. With this in mind, the trade-off relationships between lift/drag and volume, identified in the previous section, explain why lift and drag tend to increase with M_{design} .

Furthermore, as seen in Figure 7 for L_{M5} , as M_{design} increases, the range of lift also appears to increase (i.e the difference between the maximum and minimum values). The same is seen for the D_{M5} in Figure 22c of [30]. For a given M_{design} , the minimum volume waverider is obtained when:

$$\mathbf{X} = [M_{\text{design}}, X_1, X_2, X_3, X_4] = [M_{\text{design}}, 0, 0, 1, 1]$$

which corresponds to a flat SC with the lowest possible USC. On the other hand, the maximum volume waverider is obtained when:

$$\mathbf{X} = [M_{\text{design}}, X_1, X_2, X_3, X_4] = [M_{\text{design}}, X_{1\text{crit}}, 1, 0, 0]$$

which corresponds to a SC with maximum curvature, flat USC and maximum flat section in the SC. The minimum and maximum volumes were recorded for three values of M_{design} and this is shown in Table 5. ΔV denotes the different between the maximum and minimum. Table 5 highlights the increase in ΔV

Table 5. Maximum and minimum values of volume for three design mach numbers

M_{design}	V_{min} [m ³]	V_{max} [m ³]	ΔV [m ³]
5.0	2.54	13.31	10.68
6.5	4.11	21.02	16.91
8.0	4.91	24.90	19.99

with M_{design} . Therefore, this explains why the plots of lift and drag against M_{design} appear to "widen" with an increase in design Mach number.

3.2. Single-Objective Optimisation Cases

Figure 7 presents the optimal solutions for the single-objective cases introduced in 2.4.2 which aim at maximising the volume and the L_{M5} . A scatter plot of the originally sampled design space used in creating the surrogate model and the surrogate predicted optimal solution is also provided for each objective function. The maximum volume case results in values for X_2 of 1 and a design Mach number of 8. The behaviour is in line with the dominant identification of both these parameters on the Volume as indicated in the feature analysis (Sub-section 3.1). X_1 has reached its critical value and the values of X_3 and X_4 do not play a role any more as a flat USC is obtained. The maximum L_{M5} solution is also one where the design Mach number is 8 but unlike the other optimal solution, both X_1 and X_2 do play a role. The intersection of the USC and LCS is closer to the LSC value at the symmetry plane resulting in a more rounded USC. A posteriori CFD analysis of the optimal shapes did indicate relative errors in surrogate models predicted values below 1 % (see Table 16 of [30] for a quantitative overview). Overall, the optimisation setup's ability to successfully reach both known and unknown solutions with high accuracy has been demonstrated by these test cases.

3.3. Multi-Objective Optimisation Case

The multi-objective case discussed here aims at maximising the Volume while minimising the drag. In this case, both Mach 5 and Mach 8 are considered relevant as defined in 2.4.3. Figure 8 presents the resulting solutions as a Pareto front. The Drag Fitness (inverse of average drag cost function) as a function of Volume indicates the conflicting objectives and a suite of shapes are found which favor

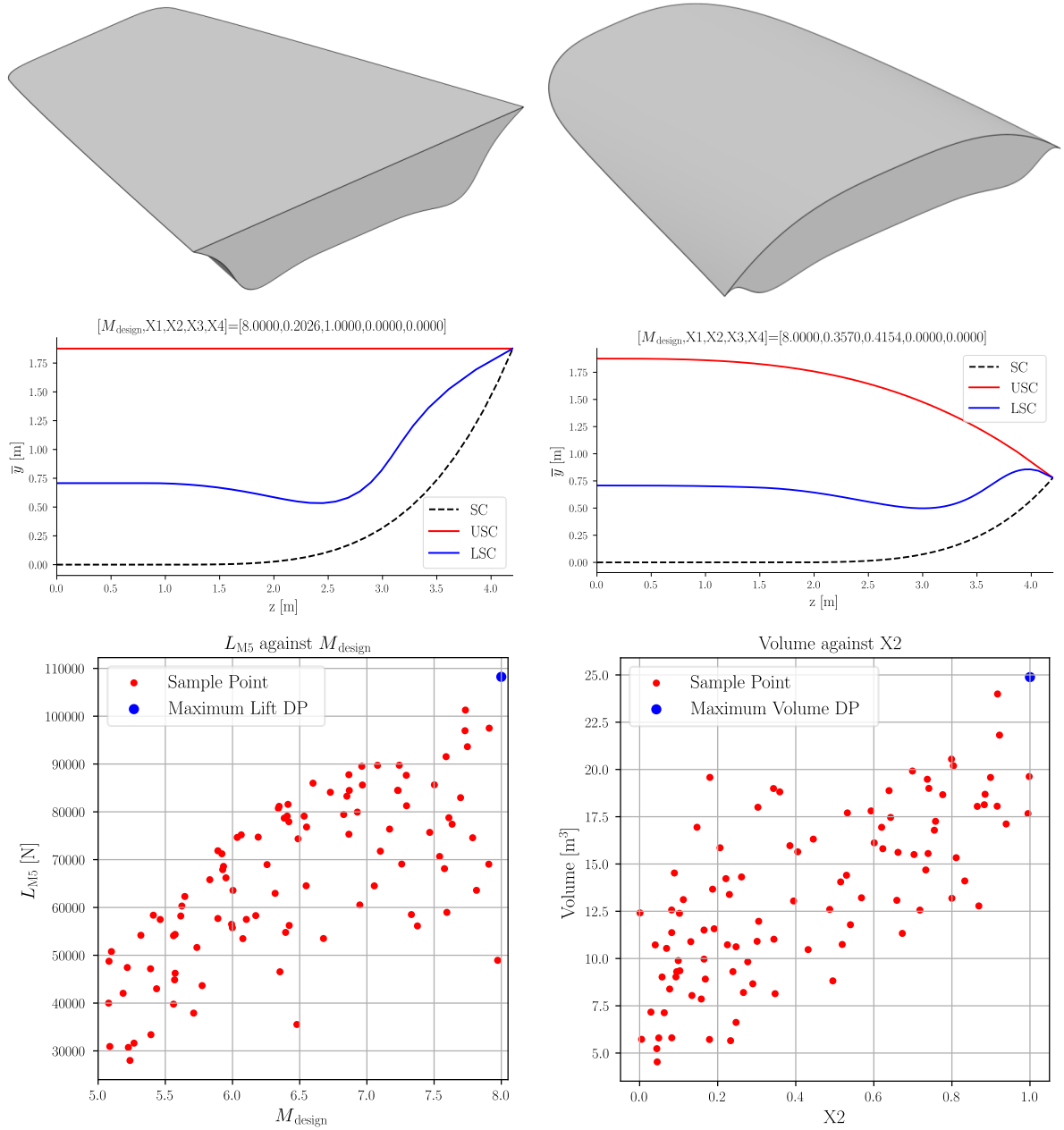


Fig 7. Visualisation of the single-objective optimisation results with maximum volume (left) and maximum lift (right) as objectives. 3D view of two optima reached in the test cases (top) along with corresponding plots of the base plane (middle). Graph of the lift as a function of the design Mach number (left) and graph of the volume as a function of the X2 design parameter for all samples.

one objective over the other. In the Pareto Front obtained by Son et al. [17], Figure 12], the trade-off relationship is non-linear. Here, the relationship between the two appears linear with a first linear region that spans from the minimum volume to a volume of 12.5 m³ while the second covers the remaining range till the maximum volume. These will be referred to as the first and second intervals hereafter. The role of M_{design} in shaping the waveriders is shown in the center of Figure 8. It can be seen that in the first interval, M_{design} remains more or less constant around 5. Then, it oscillates slightly until it starts stably increasing throughout the second interval. The two parts of the Pareto Front are visible through

multiple outputs of the design space as discussed in [30]. Finally, the behaviour of X_1 and X_2 across the Pareto Front is shown the bottom representation of Figure 8. During the first interval, both variables are zero, which corresponds to waveriders with a flat Shockwave Curve (SC). Then, a buffer/transition zone takes place until the variables stabilise into the second interval. In this second part, X_2 tends towards its maximum value of 1 while X_1 stabilises around X_{1crit} . When X_2 reaches its maximum value, X_{1crit} is the maximum value X_1 can take according to the GC. For a given M_{design} , this configuration also represents the maximum volume waverider, as has been shown in 3.1. Along with the previous discussions, this further confirms the findings that:

- The Pareto Front is divided into two main linear regions. In the first region, the increase in drag is less steep than in the second.
- In the first region, $M_{design} \approx 5$ and a flat SC is favored. Furthermore, $(L/D)_{M5}$ and v_{eff} do not vary much [30, Figure 25].
- In the second region, the maximum volume configuration for a given M_{design} is favored. Additionally, M_{design} increase rapidly while $(L/D)_{M5}$ decreases rapidly [30, Figure 25].

Furthermore, this division in the Pareto Front shows that, in the first section, the increase in volume is provided by a change in the shape of the USC - given that the SC is flat and M_{design} is constant. On the other hand, in the second section the increase in volume is provided purely by an increase in M_{design} . With regards to the Pareto Front, this indicates that the flat SC configuration can provide a sufficient increase in volume early on and maintain a low drag. However, at some point a steep increase in drag becomes inevitable if an increase in volume is to be achieved and so the solution then shifts to the second section of the Pareto Front with the increase in M_{design} .

A clear drawback of the introduction of M_{design} into the design space can be identified, which is the lack of variety in the shapes obtained. Indeed, only two "types" of waveriders were presently obtained. Son *et al.* [17, Figure 12] fixed the design Mach number and performed optimisation only through the geometric parameters $X_1...X_4$. Their resulting shapes progressively change through the Pareto Front. Introducing M_{design} may prevent such a diverse set of solutions from being reached. This is likely because an increase in volume can also be obtained via M_{design} . Whereas, when M_{design} is constant, the increase in Volume can only be achieved via the manipulation of the geometric parameters, yielding a Pareto Front like Son *et. al's* [17].

4. Conclusions

An approach for the aerodynamic shape optimisation of hypersonic waveriders, based on the reference framework by Son *et al.* [17], was further explored in this work. The approach made use of a geometric parametrization (X_1, X_2, X_3, X_4) of the back plane in conjunction with the osculating cone theory providing valid geometries. To perform design space exploration studies, inviscid CFD simulations were used for performance estimation, and an accurate Kriging surrogate model was trained for outputs of interest. A genetic algorithm was then coupled to the surrogate models and used to perform optimisation.

Feature Importances analysis via Random Forest Models indicate that the design Mach number is the driving factor for L/D , lift and drag. Volume remains primarily controlled by X_2 due to the curvature of the shockwave playing the biggest role in the volume of the vehicle. Single-objective optimisation test cases were then performed to assess the accuracy of the surrogate and performance of the overall optimisation setup. These confirmed the major role of the design Mach number in shaping the optimal waveriders for their respective objective function. A multi-objective multi-point case with conflictive objectives of maximising volume while minimising drag was presented. The Pareto Front of solutions consists of two linear regions. In the first region, the optimisation favors solutions with a flat shockwave shape and a fixed design Mach number. In the second region, M_{design} begins increasing while favoring solutions with a flat upper surface, maximum curvature and a maximum flat shock region ($X_1 = X_{1crit}$). Overall, this work presents a robust, and open-access, framework that can benefit the high-speed research community for design space exploration or trade-off studies. Various points for future work can be considered. This includes the use of viscous simulations to tune the viscous correction coefficients adopted in this work which might depend on shape families. The design-space exploration tasks

should also include static aerodynamic stability considerations to make it more practical. Similar design space explorations could be made for vehicles with different size ranges which could potentially identify similarities.

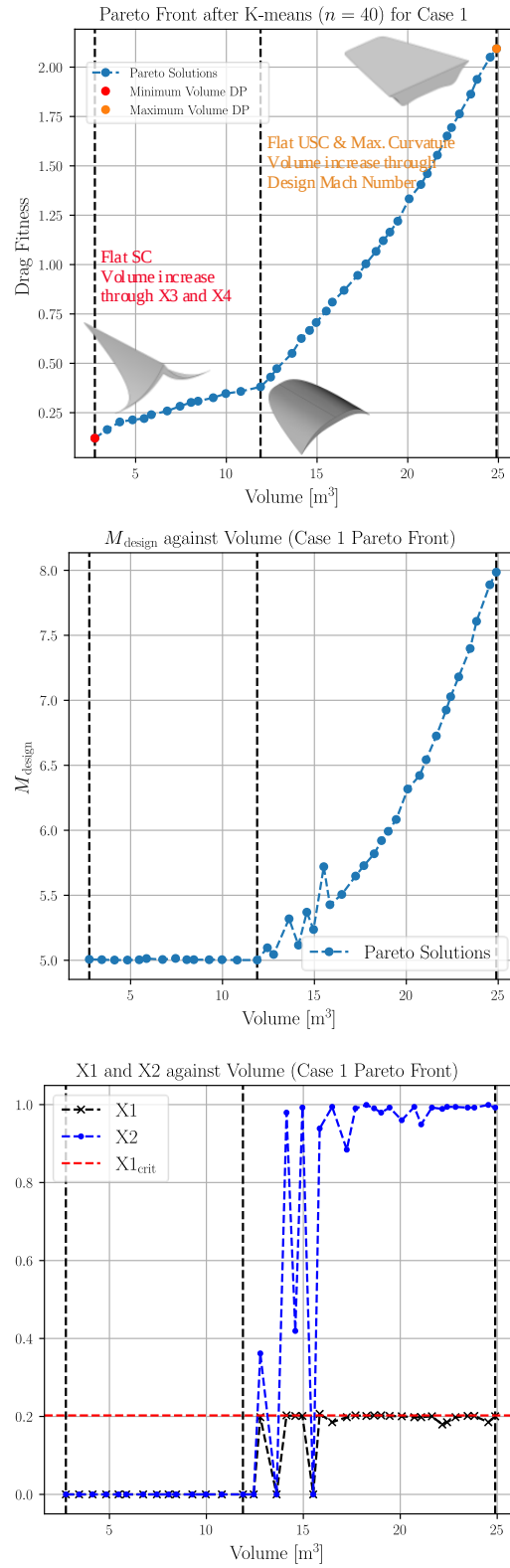


Fig 8. Pareto front as function of the volume and the drag fitness (top). The displayed typical waverider shapes indicate a swap between two types of geometries as suggested by the curve comprised of two nearly linear regions with distinct slopes. Pareto front as a function of the volume and the design Mach number (middle). Design parameters as functions of the volume indicating two distinct geometries as the results of the optimisation.

References

- [1] JD Anderson. *Hypersonic and high temperature gas dynamics, Second Edition*. AIAA, 2006.
- [2] C Bruno and PA Czysz. *Future spacecraft propulsion systems: enabling technologies for space exploration*. Springer Science & Business Media, 2009.
- [3] Dietrich Küchemann. *The Aerodynamic Design of Aircraft*. American Institute of Aeronautics and Astronautics, Inc., Reston, Virginia, 9 2012.
- [4] J Anderson Jr and M Lewis. Hypersonic waveriders-Where do we stand? In *31st Aerospace Sciences Meeting*, page 399, 1993.
- [5] MJ Lewis. Hypersonic missions and vehicle configurations. *Encyclopedia of Aerospace Engineering*, 2010.
- [6] T. R. F. Nonweiler. Aerodynamic Problems of Manned Space Vehicles. *The Journal of the Royal Aeronautical Society*, 63(585):521–528, 9 1959.
- [7] C McClinton. X-43-scramjet power breaks the hypersonic barrier: Dryden lectureship in research for 2006. In *44th AIAA aerospace sciences meeting and exhibit*, page 1, 2006.
- [8] J Hank, J Murphy, and R Mutzman. The x-51a scramjet engine flight demonstration program. In *15th AIAA international space planes and hypersonic systems and technologies conference*, page 2540, 2008.
- [9] S Walker, J Sherk, D Shell, R Schena, J Bergmann, and J Gladbach. The darpa/af falcon program: The hypersonic technology vehicle# 2 (htv-2) flight demonstration phase. In *15th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, page 2539, 2008.
- [10] K Bowcutt, A Paull, D Dolvin, and M Smart. Hifire: An international collaboration to advance the science and technology of hypersonic flight. In *Proceedings of the 28th International Congress of the Aeronautical Sciences*, pages 2012–998. ICAS Secretariat RA Leiden, The Netherlands, 2012.
- [11] S Di Benedetto, MP Di Donato, A Rispoli, S Cardone, J Riehmer, Johan Steelant, and L Vecchione. Hexafly-int project: Design of a high speed flight experiment. *Proceedings of ISHF*, 2017.
- [12] J Autenrieb and N Fezans. Flight control design for a hypersonic waverider configuration: A non-linear model following control approach. *CEAS Space Journal*, pages 1–24, 2024.
- [13] N Takashima and MJ Lewis. Wedge-cone waverider configuration for engine-airframe interaction. *Journal of Aircraft*, 32(5):1142–1144, 9 1995.
- [14] RP Starkey and MJ Lewis. Simple analytical model for parametric studies of hypersonic waveriders. *Journal of Spacecraft and Rockets*, 36(4):516–523, 1999.
- [15] J Shi, L Zhang, B Jiang, and B Ai. Aerodynamic force and heating optimization of htv-2 typed vehicle. In *21st AIAA International Space Planes and Hypersonics Technologies Conference*, page 2374, 2017.
- [16] Konstantinos Kontogiannis, András Sóbester, and Nigel Taylor. Efficient Parameterization of Waverider Geometries. *Journal of Aircraft*, 54(3):890–901, 5 2017.
- [17] Jiwon Son, Chankyu Son, and Kwanjung Yee. A Novel Direct Optimization Framework for Hypersonic Waverider Inverse Design Methods. *Aerospace*, 9(7):348, 6 2022.
- [18] Helmut Sobieczky, F Dougherty, and Kevin Jones. Hypersonic Waverider Design from Given Shock Waves. 5 1990.
- [19] Feng Ding, Jun Liu, Chi-bing Shen, Zhen Liu, Shao-hua Chen, and Xiang Fu. An overview of research on waverider design methodology. *Acta Astronautica*, 140:190–205, 11 2017.

- [20] Zhen-tao Zhao, Wei Huang, Li Yan, and Yan-guang Yang. An overview of research on wide-speed range waverider configuration. *Progress in Aerospace Sciences*, 113:100606, 2 2020.
- [21] Kevin G. Bowcutt. Multidisciplinary Optimization of Airbreathing Hypersonic Vehicles. *Journal of Propulsion and Power*, 17(6):1184–1190, 11 2001.
- [22] Romain Wuilbercq. *Multi-disciplinary modelling of future space-access vehicles*. PhD thesis, University of Strathclyde, UK, 2015.
- [23] N Viola, P Roncioni, O Gori, and R Fusaro. Aerodynamic Characterization of Hypersonic Transportation Systems and Its Impact on Mission Analysis. *Energies*, 14(12):3580, 6 2021.
- [24] Chris Bliamis, Constantinos Menelaou, and Kyros Yakinthos. Implementation of various-fidelity methods for viscous effects modeling on the design of a waverider. *Aerospace Science and Technology*, 133:108141, 2 2023.
- [25] Alberto Simon Felix. Adjoint-Based Shape Optimisation of Hypersonic Vehicles. Master’s thesis, Cranfield University, Cranfield, UK, 2023.
- [26] Marc Famada Vizcaino. Aerodynamic shape optimisation of hypersonic waveriders using CFD and adjoint-based method. Master’s thesis, Cranfield University, Cranfield, UK, 2023.
- [27] Jesse R. Maxwell. Efficient Design of Viscous Waveriders with CFD Verification and Off-Design Performance Analysis. In *53rd AIAA/SAE/ASEE Joint Propulsion Conference*, Reston, Virginia, 7 2017. American Institute of Aeronautics and Astronautics.
- [28] Siemens Digital Industries Software. Simcenter STAR-CCM+ 2402 (19.02.009-R8), 2024.
- [29] Jade Nassif. Waverider Generator 2.0.1, 6 2024.
- [30] Jade Nassif. Multi-Objective Multi-Point Optimization of a Hypersonic waverider. Master’s thesis, Cranfield University, Cranfield, UK, 2024.
- [31] Chuanzhen Liu and Peng Bai. Waverider Design Using Osculating Method. *AIAA Journal*, 62(5):1643–1661, 5 2024.
- [32] David Cowden and James Cook. cadquery 2.4.0, 1 2024.
- [33] Feng Qu, Tianyu Wang, Chaoyu Liu, Junjie Fu, and Junqiang Bai. Aerodynamic shape optimization of the vortex-shock integrated waverider over a wide speed range. *Aerospace Science and Technology*, 143:108696, 12 2023.
- [34] Wen Liu, Chen-An Zhang, Fa-Min Wang, and Zheng-Yin Ye. Design and Optimization Method for Hypersonic Quasi-Waverider. *AIAA Journal*, 58(5):2132–2146, 5 2020.
- [35] Peter G Cross and Michael R West. Simulation of hypersonic flowfields using Star-CCM+. *China Lake, CA, USA: Naval Air Warfare Center Weapons Division*, 2019.
- [36] Meng-Sing Liou. A Sequel to AUSM: AUSM+. *Journal of Computational Physics*, 129(2):364–382, 12 1996.
- [37] J. A. Meijerink and H. A. van der Vorst. An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-Matrix. *Mathematics of Computation*, 31(137):148, 1 1977.
- [38] V. Venkatakrishnan. On the accuracy of limiters and convergence to steady state solutions. In *31st Aerospace Sciences Meeting*, Reston, Virginia, 1 1993. American Institute of Aeronautics and Astronautics.
- [39] D G Fletcher. Fundamentals of hypersonic flow-aerothermodynamics. *Critical Technologies for Hypersonic Vehicle Development (Von Karman Inst. for Fluid Dynamics, Rhode-St-Genese, 2005)*, pages 1–3, 2004.

- [40] Roop N Gupta, Kam-Pui Lee, Richard A Thompson, and Jerrold M Yos. Calculations and curve fits of thermodynamic and transport properties for equilibrium air to 30000 K. Technical report, 1991.
- [41] P Saves, R Lafage, N Bartoli, Y Diouane, J Bussemaker, T Lefebvre, J T Hwang, J Morlier, and J R R A Martins. SMT 2.0: A Surrogate Modeling Toolbox with a focus on Hierarchical and Mixed Variables Gaussian Processes. *Advances in Engineering Software*, 188:103571, 2024.
- [42] M. D. McKay, R. J. Beckman, and W. J. Conover. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 21(2):239, 5 1979.
- [43] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.
- [44] John H Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.
- [45] Ahmed Fawzy Gad. Pygad: An intuitive genetic algorithm python library. *Multimedia Tools and Applications*, pages 1–14, 2023.
- [46] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.