TU Delft

Delft University of Technology

Adaptive Dynamic Incremental Nonlinear Control Allocation for Aircraft With Innovative Control Effectors

Stam, N.; de Visser, C.C.

DOI
10.2514/6.2025-0082

Publication date
2025

Document Version
Final published version

Published in
AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2025

Citation (APA)
Stam, N., & de Visser, C. C. (2025). Adaptive Dynamic Incremental Nonlinear Control Allocation for Aircraft With Innovative Control Effectors. In *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2025* Article AIAA 2025-0082 (AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2025). American Institute of Aeronautics and Astronautics Inc. (AIAA). https://doi.org/10.2514/6.2025-0082

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

AIAA SciTech Forum
6-10 January 2025, Orlando, FL
AIAA SCITECH 2025 Forum

10.2514/6.2025-2804

Check for updates

Online Data-Driven Optimization of Aerodynamic Performance for an Unconventional Morphing Aircraft

Thymen Woldhuis*, Salvatore Asaro†, and Xuerui Wang‡
Delft University of Technology, Delft, 2629 HS, The Netherlands

In nature, birds can intelligently adapt their wing shapes to their environment. This paper aims to replicate this capability by designing an online data-driven aerodynamic performance optimization framework for an unconventional morphing aircraft. Compared to state-of-the-art methods, the proposed framework efficiently finds global optima with reduced computational load when addressing time-varying, nonlinear, and non-convex problems. It also demonstrates enhanced adaptability to unforeseen scenarios. In the event of a sudden actuator fault, the algorithm can automatically detect the fault, adapt the onboard data-driven model, and continue performing optimization and trimming tasks using the remaining healthy actuators. Additionally, the paper addresses the optimal number of actuators within a morphing surface, considering the tradeoff between optimization performance and the weight penalty. High-fidelity simulations demonstrate that through active morphing, the proposed framework achieves drag reductions of 1.9–4.9 % during cruise and up to 12.6 % at higher operational lift coefficients (due to heavier weight and lower speed), resulting in an overall drag reduction of 2.97 % over a typical flight cycle, which corresponds to fuel savings of approximately 150 kg/h. This research represents a significant advancement in sustainable aviation, contributing to reduced fuel consumption, lower emissions, and improved fault tolerance for next-generation aircraft.

Nomenclature

BOBYQA	Bound Optimization by Quadratic Approximation
CMA-ES	Covariance Matrix Adaptation Evolution Strategy
DIRECT	Dividing Rectangles
FTO	Fault Tolerant Optimization
MTOW	Maximum Takeoff Weight
PYSWARM	Particle Swarm Pattern Search Method
RANS	Reynolds-Averaged Navier–Stokes
VCCTEF	Variable Camber Continuous Trailing Edge Flap
VLM	Vortex Lattice Method

α	Angle of attack	$_i$	Iteration
C_D	Coefficient of drag	J	Cost
C_L	Coefficient of lift	k_1	Coefficient of lift deviation penalty
$C_{L_{\text{target}}}$	Target lift coefficient	k_2	Moment coefficient deviation penalty
C_M	Pitching moment coefficient	m	Number of actuators
$\boldsymbol{\delta}$	Flap deflections	Mach	Mach number
F_{weight}	Flap weight	$_{\text{opt}}$	Optimal
γ	Flight path angle	ψ	Engine installation angle
h	Altitude	Re	Reynolds number

*MSc Aerospace Engineering, Department of Control and Operations, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands, thymen.woldhuis@gmail.com.

†Postdoctoral Researcher, Department of Flight Performance and Propulsion, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands, s.asaro@tudelft.nl.

‡Tenured Assistant Professor, Department of Aerospace Structures and Materials & Department of Control and Operations, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands, X.Wang-6@tudelft.nl, Senior Member AIAA.

Copyright © 2025 by Delft University of Technology. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

ρ	Density	U	Airspeed
$\boldsymbol{s}$	Spline control parameters		
T	Thrust		

I. Introduction

In modern aircraft design, offline optimization is typically performed to achieve optimal aerodynamic performance at a specific point within the flight envelope, known as the "design point". Extensive offline optimization using analytical models is then employed to determine the optimal aircraft configurations at various points in the flight envelope. These offline solutions are stored in predefined lookup tables [1]. However, when the aircraft operates under different flight conditions, the aerodynamic efficiency tends to degrade. Moreover, deviations from the nominal configuration, such as those arising from manufacturing variability, icing, and structural damage, can alter the actual aerodynamic characteristics from the ideal digital models, leading to suboptimal performance, increased fuel consumption, and emissions.

Online optimization enables in-flight adjustments using actual aerodynamic data, rather than relying on static offline models. This concept is inspired by nature, where birds continuously adjust their wing shapes to optimize performance in response to changing conditions. Applying this adaptive approach to aviation allows for significant performance improvements during flight, enhancing aerodynamic efficiency across a wider range of conditions. By dynamically responding to environmental changes, online optimization offers potential benefits such as improved fuel efficiency, reduced emissions, and overall enhanced aircraft performance, providing a more effective alternative to traditional optimization methods based on idealized, pre-defined models.

With this motivation, a series of studies conducted by NASA Ames Research Center explored the online optimization of a Variable Camber Continuous Trailing Edge Flap (VCCTEF) system [2]. A real-time least-squares drag minimization method was developed, which proposed using a recursive least-squares algorithm to estimate the aircraft's aerodynamic coefficients and fit them to a polynomial model [3]. The optimal wing shape was then determined using a gradient-based optimization method, specifically Newton-Raphson. This approach resulted in 8.4 % drag reduction as compared to a clean wing for cruise [4]. Later wind tunnel experiments resulted in drag reductions of up to 9.4 % for off-design lift coefficients [1].

While promising, these studies have several limitations. First, the use of a polynomial-based aircraft model imposes a fixed structure that is valid only near the trimming points, limiting its adaptability. Achieving detailed accuracy in local regions while preserving global validity would necessitate a substantial increase in the polynomial's degree. Second, the optimization problem for aircraft performance can be nonlinear and non-convex. Gradient-based optimization strategies may converge to local minima or saddle points, exhibit sensitivity to initialization, and progress slowly in flat regions or near ill-conditioned points. Moreover, these methods struggle with non-smooth functions and can involve costly gradient or Hessian calculations, particularly in large-scale problems.

To address these limitations, an alternative approach was proposed for the Smart-X morphing wing [5] and tested in a wind tunnel experiment [6], achieving up to 19.8 % drag reduction. This study introduced the use of radial basis function neural networks to model the aircraft's aerodynamics in real time, followed by optimization using the covariance matrix adaptation evolution strategy (CMA-ES) [7]. While this approach improved the transition from local to global optimization, further enhancements in computational efficiency are necessary. Genetic optimization methods often require a large number of function evaluations and are typically beneficial only when the number of variables exceeds ten or when the search space is poorly defined. Furthermore, although neural networks have proven effective in capturing aircraft dynamics, they operate as black-box models and impose significant computational demands.

Up to this point, we have identified several research gaps from an algorithm development perspective. It has become evident that an online, global, and computationally efficient aerodynamic performance optimization framework is required. In addition to these algorithmic challenges, challenges also arise from a practical point of view.

The first challenge pertains to the hardware configuration on which the optimization is applied. The research conducted in [5, 6] focused solely on wing optimization, neglecting challenges posed by free-flight conditions, such as maintaining trim and reserving control authority for stability and maneuverability. In contrast, the studies in [8, 9] applied optimization to free-flying aircraft, but the configurations were conventional, i.e., tube-and-wing designs. Consequently, optimizing the wing's control surface settings in these studies had limited impact on mitigating the drag penalties caused by the large fuselage.

This research aims to develop an optimization framework for the Flying-V [10], an innovative wing-body aircraft

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

developed by TU Delft. The Flying-V has the potential to achieve a 25 % increase in efficiency compared to conventional tube-and-wing aircraft of similar size [10]. However, its implementation poses significant challenges. Due to the absence of conventional elevators and rudders, the control surfaces on the wing must perform multiple functions simultaneously, including three-axis trim, maneuverability, and performance optimization. Furthermore, Flying-V presents strong nonlinearity and non-convexity at high angles of attack, requiring an optimization strategy capable of addressing such complexity.

The second practical challenge lies in the control surfaces used to implement the optimization algorithm. In recent years, there has been an increasing trend toward expanding the number of trailing-edge control surfaces. From a theoretical perspective, as the number of control surfaces increases, the degrees of freedom for precisely manipulating aerodynamic behavior also increase. This is why a distributed mini-flap configuration was proposed in [11]. However, this configuration does not ensure smooth transitions between adjacent flaps. In contrast, a seamless active morphing wing concept was introduced in [5]. Nevertheless, this research does not address the optimal number of actuators within the morphing surface area, considering the tradeoff between performance and the weight penalty of additional actuators.

Although increasing the number of actuators provides more control degrees of freedom, it also raises the likelihood of actuator faults in practice, as verified in the wind tunnel experiment in [1]. However, when an actuator fault occurred in that study, it was manually excluded from the control space. The algorithm lacked capabilities for automatic fault detection, fault tolerance, and onboard model adaptation, while still achieving optimal performance in post-fault scenarios. These aspects are particularly crucial during real-world operations and will be addressed in this paper.

In addition, the performance of the optimization algorithm is highly dependent on the operational point and the aircraft's mass. While performance improvements of up to 9.4 % and 19.8 % were achieved in [1] and [6], respectively, these numbers may be overly optimistic when considering a typical flight cycle. This is because commercial aircraft spend the majority of their time in cruise conditions, where their shape is likely to be already near optimal. A more realistic assessment of optimization performance over a typical flight cycle will be evaluated in this paper.

In light of the gaps in the literature, this paper proposes a fault-tolerant, global, online, and computationally efficient aerodynamic performance optimization framework and applies it to a nonlinear, free-flying, flying-wing morphing aircraft. The framework can automatically identify faulty actuators, adapt the onboard model, and achieve both trimming and performance optimization simultaneously using the remaining healthy actuators. It offers improved computational efficiency, enhanced performance, and better adaptability in handling time-varying, nonlinear, and non-convex problems compared to state-of-the-art methods, as demonstrated by a quantitative comparative study in this paper. Moreover, the paper addresses the optimal number of actuators for a morphing control surface, considering the tradeoff between performance and weight penalty. Finally, high-fidelity simulations demonstrate the effectiveness of the proposed framework throughout the entire flight envelope and provide a more realistic estimation of drag reduction during a typical commercial aircraft flight cycle.

The paper is structured as follows: The detailed algorithm design, on-board model, and optimizer are explained in Sec. II. The simulation results are then presented in Sec. III, followed by conclusions in Sec. IV.

II. Methodology

This section presents the methodology used to model, simulate and optimize the system. First, subsection II.A outlines the overall optimization architecture. Subsection II.B delves into the development of the aerodynamic model, detailing how it is constructed and what flight conditions are considered. A data-driven on-board surrogate model is created in subsection II.C. The on-board model will then be used to optimize the angle of attack and actuator inputs. Subsection II.D presents the optimizer design. Finally, subsection II.E explores strategies for managing actuator faults, ensuring system robustness and continued operation under faulty conditions.

A. Optimization architecture

The objective of the optimization is to find the best combination of the angle of attack (α) and actuation inputs ($\boldsymbol{\delta}$) that maximize aerodynamic efficiency while satisfying the longitudinal trimming requirement. This should be achieved throughout the entire flight envelope, bridging the simulation-to-reality gap, adapting to new conditions, and tolerating unforeseen scenarios including icing, actuator faults, and structural damages.

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

The optimization problem is formulated as follows:

$$\begin{aligned} \underset{\alpha,\boldsymbol{\delta}}{\text{minimize}} \quad & J(\alpha,\boldsymbol{\delta}) = C_D(\alpha,\boldsymbol{\delta}) + k_1(C_{L,\text{target}}(\alpha,\boldsymbol{\delta}) - C_L(\alpha,\boldsymbol{\delta}))^2 + k_2(C_M(\alpha,\boldsymbol{\delta}))^2 \\ \text{subject to} \quad & \alpha \in [\alpha_{\min}, \alpha_{\max}], \\ & \boldsymbol{\delta} \in [\boldsymbol{\delta}_{\min}, \boldsymbol{\delta}_{\max}], \\ & \dot{\boldsymbol{\delta}} \in [\dot{\boldsymbol{\delta}}_{\min}, \dot{\boldsymbol{\delta}}_{\max}], \\ & \boldsymbol{\delta} \in \left\{\boldsymbol{\delta} : [\boldsymbol{H}, -\boldsymbol{H}]^\top \boldsymbol{\delta} \le [\bar{\boldsymbol{\delta}}_{\text{adj}}^\top, \bar{\boldsymbol{\delta}}_{\text{adj}}^\top]^\top\right\} \end{aligned} \tag{1}$$

where C_L is the lift coefficient; C_M is the pitching moment coefficient that should be zero at trim; C_D is the drag coefficient to be minimized for reducing fuel consumption and emissions. k_1 and k_2 are user-defined weights. This optimization problem is nonlinear and non-convex, as will be shown in subsection II.C, the aerodynamic coefficients are nonlinear and non-convex functions of the optimization variables.

The morphing surface is subject to position and velocity constraints of ± 20 deg and 45 deg/s, respectively [12]. The position constraint is further tightened to ± 16 deg in optimization to reserve margins for maneuvers. This constraint can also be set asymmetrically to account for pitch moment trimming. The number of actuators in $\boldsymbol{\delta}$ is treated as a variable to be optimised, intending to answer the research question of what is best number of actuators within the morphing surface. The morphing surface is composed of composite and polymeric elastic materials, which impose physical constraints on the relative positional differences between adjacent actuators [13]. Without loss of generality, let the number of actuators within the morphing surface be denoted as m, resulting in relative position constraints given by $|\delta_{i+1} - \delta_i| \le \bar{\delta}_{\text{adj},i}$, for $i = 1, 2, \ldots, m-1$. This can be expressed in vector form as $[\boldsymbol{H}, -\boldsymbol{H}]^\top \boldsymbol{\delta} \le [\bar{\boldsymbol{\delta}}_{\text{adj}}^\top, \bar{\boldsymbol{\delta}}_{\text{adj}}^\top]^\top$, where $\boldsymbol{H} \in \mathbb{R}^{(m-1)\times m}$ with $H_{i,i} = 1$ and $H_{i,i+1} = -1$ for $i = 1, 2, \ldots, m-1$, and all other elements of $\boldsymbol{H}$ equal to zero [13]. Finally, The angle of attack α is constrained to $\alpha \in [0, 10]$ deg for stall and pitch break protection.

The optimization is performed under steady, trimmed flight conditions over a longer time scale. The dynamic effects of gusts and maneuvers are considered shorter-term and can be decoupled from this optimization problem using low-pass filtering on the sensor measurements, based on the time-scale separation principle. In addition, outliers in the sensor measurements should be detected and removed from the optimization loop.

Perpendicular to the flight path, the trimming condition can be expressed as follows:

$$-T\sin(\alpha + \psi) - L + W\cos\gamma = 0 \tag{2}$$

where T, L, and W represent thrust, lift, and weight, respectively; ψ denotes the engine installation angle, and γ is the flight path angle. Assume that $T\sin(\alpha+\psi)$ is small compared to L, which is generally valid for commercial aircraft. For aircraft with a higher thrust-to-weight ratio, this assumption can be removed by incorporating T into the optimization problem. Based on this assumption, the target lift coefficient, $C_{L_{\text{target}}}$, required to satisfy the trim condition, is calculated as follows:

$$C_{L_{\text{target}}} = \frac{2W\cos\gamma}{\rho U^2 S} \tag{3}$$

where $\rho U^2/2$ is the dynamic pressure and S is the wing surface area.

Figure 1 presents the optimization architecture in a schematic view. This paper considers the weight (W), Mach number (Mach), altitude (h), and flight path angle (γ) as flight condition definition variables. From this, the corresponding $C_{L_{\text{target}}}$ is calculated using Eq. (3). The computed $C_{L_{\text{target}}}$ acts as an input of the optimization loop, in which an optimizer minimizes the cost function based on a surrogate onboard model. The choice of the optimizer will be detailed in subsection II.D. To reduce computational loads of the optimizer and to ensure spanwise smoothness of the morphing surface, a spline function is created to map the real actuator motions to a virtual spline-curve. Once an optimal combination of α_i and spline coefficient vector $\boldsymbol{s}_i$ are found, they are then converted back to the physical actuation angles $\boldsymbol{\delta}_i$ to be used by the on-board surrogate model. In this paper, the on-board surrogate model is trained by datasets generated by real-world wind tunnel experiments and simulations based on Reynolds-Averaged Navier–Stokes (RANS) equations and Vortex Lattice Method (VLM). The aerodynamic coefficient estimated by the on-board model are then used by the cost function.

The optimization loop is run until the optimizer converges to an optimal α_{opt} and $\boldsymbol{\delta}_{\text{opt}}$, then the suggested morphing configuration is tested on the real aircraft and the on-board model is updated online.

In practical applications, the variables h, U, Mach number, ρ, and γ can be determined from measurements provided by the Pitot tube, air data computer, inertial navigation system, pressure altimeter, and Global Positioning System. The

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

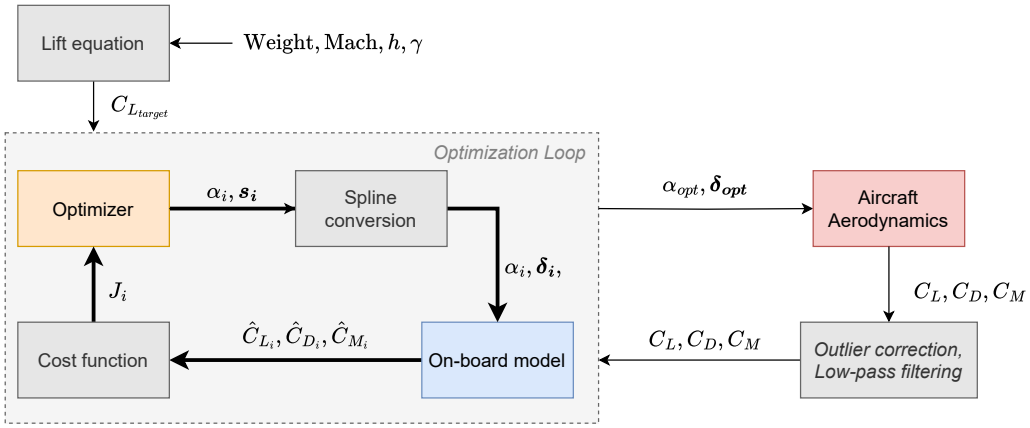


Fig. 1 Online shape optimization architecture.

wing surface area S is typically a known constant. The gross aircraft weight W can be estimated onboard based on fuel consumption and the known gross take-off weight, commonly recorded on commercial airliners. Additionally, W can be identified online by incrementally adjusting thrust and measuring the resultant accelerations. The aerodynamic coefficients C_L, C_D, and C_M can be estimated onboard using the three-axis force and moment dynamic equations, with linear and angular accelerations and rates obtained from the Inertial Measurement Unit; gross weight is estimated or identified as previously discussed; thrust can be inferred from engine parameters, including measurements of fuel flow rate, rotational speeds, and inlet and exhaust pressures and temperatures. The variables α and $\boldsymbol{\delta}$ are directly measurable.

B. Aerodynamic model of the Flying-V

The Flying-V aircraft has an unconventional configuration as presented in Fig. 2 with the morphing surface area, considered in this study, highlighted in yellow. Because of its tailless configuration, the morphing surfaces are responsible for multiple objectives including trimming, maneuver control, and online aerodynamic shape optimization. The morphing concept adopts the seamless Translation Induced Camber (TRIC) Smart-X Alpha in [5].

The Flying-V presents nonlinear behavior at high angles of attack and has transonic cruise speeds, leading to nonlinear, non-convex, and viscous behaviors. Therefore, high-fidelity simulations based on RANS and VLM are performed.

The RANS simulations are conducted with Ansys Fluent and the grid is created with Integrated Computer Engineering and Manufacturing for Computational Fluid Dynamics (ICEM CFD). After a mesh independence study, the number of elements used to discretize the domain is $12.1{\times}10^6$ which leads to maximum differences with respect to the Richardson extrapolated value of less than 10 % for C_D, C_L and C_M. The mesh independence study is conducted at different α and Mach numbers. Further details on the meshing and simulation strategies are described in [15]. The simulations are performed in the range of Mach = 0.2 to Mach = 0.7. This data is used as a basis for the zero-degree angle of attack coefficients and the coefficients with respect to α.

To capture the aerodynamic mapping from actuation inputs to the coefficients, RANS simulations can also be performed. However, this would be computational inefficient because we treat the number of actuators as a variable in this paper, leading to a large number of possible compositions. To enhance computational efficiency, the contribution of the morphing surface is captured by VLM simulations. Low fidelity tools as the VLM cannot predict this type of behavior. However, if properly calibrated, VLM can predict the change in forces and moment induced by the morphing surface for longitudinal trimming purposes, for limited deflections as considered in this study. The VLM simulations are conducted with OpenVSP [16]. A total of 24 elements are used to discretize the aircraft along the chord and 72 elements along the span. The winglets of the Flying-V are not included in this study. A Python script was used to run the OpenVSP simulations for any actuator configuration at any flight condition.

The VLM simulations are calibrated with wind tunnel experiments on a half model of the Flying-V at freestream velocity of 25 m/s. Further details on the experiments can be found in [15]. The wind tunnel model is equipped with 3 control surfaces which span in the same area as the morphing surface. Each control surface is tested individually allowing to determine the impact of each of them on C_D, C_L and C_M. The simulation data is then calibrated by

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

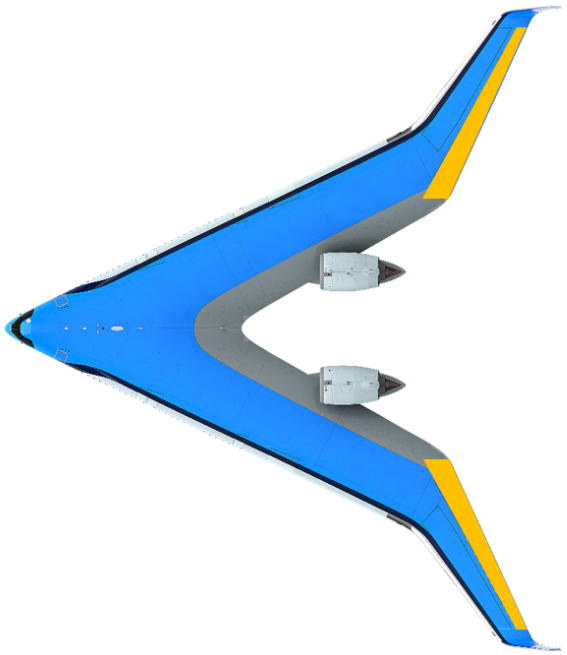


Fig. 2 The Flying-V aircraft with morphing surface area indicated in yellow (modified from [14]).

multiplying the C_{L_δ}, $C_{D_{\delta^2}}$, and C_{M_δ} coefficients by the corrections factors a, b, and c, as shown in the Appendix. The construction of the aerodynamic model is summarized in Fig. 3.

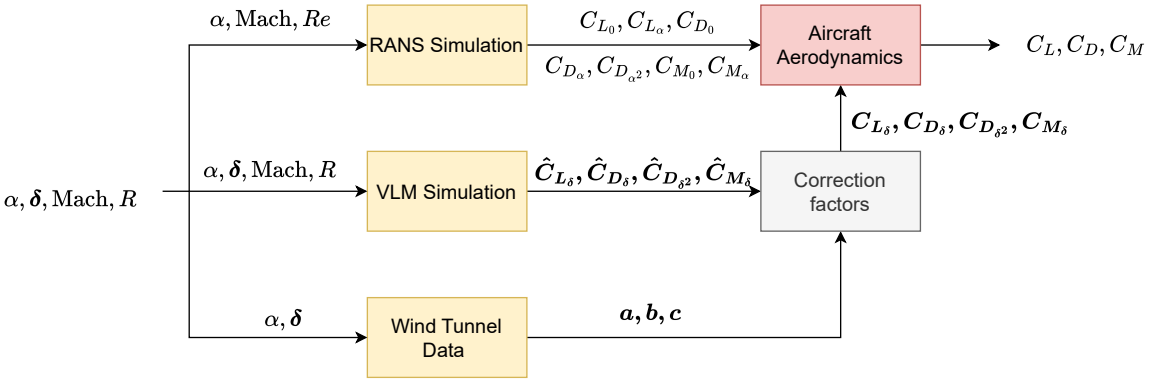


Fig. 3 The construction of the aerodynamic model of the Flying-V.

As mentioned before, the ideal number of actuators on the wing is set as a variable to be optimized and will be explored in subsection III.A. To ensure that morphing along the span is smooth and to reduce the number of input variables for the optimizer, the virtual shape function can be used to map the real actuator deflections to the coefficients of virtual shape functions. The state-of-the-art practice is to adopt Chebyshev polynomials [17] as virtual shape functions [3, 18]. However, the parameters used to tune the Chebyshev polynomial are coupled and changing one parameter has a global influence on the shape of the polynomial. This hinders the state-of-the-art approach from precisely manipulating the aerodynamic properties at specific local locations. In view of this gap, this research adopts the univariate spline function as the base of virtual shape functions, which allows precise local manipulation and also ensures smooth transient in between adjacent actuators. Figure 4 shows an example of actuator deflections governed by the univariate spline curve. A tradeoff was made regarding the number of spline control coordinates by increasing the number of control points and calculating the mean squared error (MSE) between 1000 random functions and their spline

approximations. It was observed that after five control points, the improvement in approximation accuracy became negligible. Therefore, the number of spline parameters was set to five.

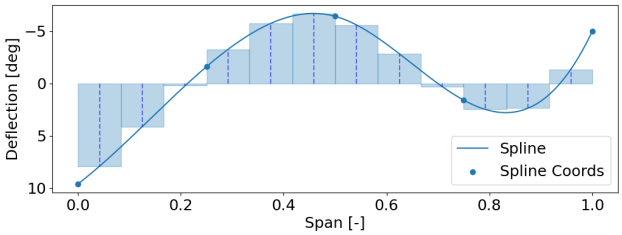


Fig. 4 Example of an univariate spline curve smoothly connects 12 morphing actuators.

The flight conditions considered are given in Table 2 and are based on the available data from Ref. [15]. The flight path angle is estimated using an earlier study on the Flying-V [19]. The fuel weight is assumed to decrease linearly over the course of the flight. Consequently, the weight for each flight phase is calculated by subtracting the estimated fuel consumption from the MTOW. The $C_{L_{\text{target}}}$ is calculated using Eq. (3) and the Flying-V parameters in Table 1.

Table 1 Aircraft modelling parameters of the Flying-V.

Variable	Value	Unit
Wing surface	898	m^2
Mean aerodynamic chord	18	m
Center of gravity	28.1	m
Maximum Takeoff Weight (MTOW)	278000	kg
Fuel weight	109000	kg

Table 2 Flight conditions considered for a typical Flying-V flight profile.

	h (m)	ρ (kg/m)	U (m/s)	Mach	Re (10e6)	γ (deg)	mass (kg)	$C_{L_{\text{target}}}$
Take-off	0	1.225	68.06	0.2	83.9	15.00	278000	**1.03**
Initial Climb	3250	0.886	131.03	0.4	124	4.50	277539	**0.40**
Climb to cruise	6500	0.624	172.90	0.55	123	2.50	275816	**0.32**
Cruise *	9750	0.426	210.39	0.7	110	0.00	223500	**0.26**
Descent	6500	0.624	172.90	0.55	123	-2.50	172508	**0.20**
Approach	3250	0.886	131.03	0.4	124	-4.50	171357	**0.25**
Landing	0	1.225	68.06	0.2	83.9	-3.00	169000	**0.65**

C. On-board data-driven model

To accelerate online computation, a data-driven global on-board surrogate model is learned from the aerodynamic datasets. Various approaches are available, including polynomial-based, spline-based, and neural network models [20], each offering distinct advantages. To determine the most suitable system identification model for this study, a thorough analysis of the requirements is essential. First, the model must be global in scope, capable of capturing nonlinear behavior. Second, it must be adaptive, allowing for updates as new data becomes available. Finally, it must be computationally efficient, with minimal evaluation costs.

*This Flying-V can cruise at the higher Mach number. In this research, Mach number of 0.7 is obtained because this is the highest Mach number at which accurate data is still available.

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

Polynomial models are frequently employed for system identification due to their simplicity. However, their accuracy declines when applied to nonlinear systems. Additionally, polynomial models require data collected across a wide range of global variations to accurately represent the system. In contrast, neural networks are capable of modeling both global and complex behavior. Several studies have utilized neural networks to develop on-board models [21, 22]. Despite this, neural networks are computationally demanding and often rely on global basis functions. Moreover, they lack transparency and are prone to numerical instabilities [23].

Multivariate splines, by contrast, have been shown to effectively generate a global aircraft model [20, 24], and in the case of the Flying-V, under specific test conditions, they have even outperformed other identification methods [25]. Spline-based models are composed of piecewise functions, allowing different sections of the model to be fitted independently without interference. This characteristic is particularly advantageous for global models that require high accuracy in local regions, especially around trimming points.

The independent input parameters for the on-board model are angle of attack (α), control surface deflections ($\boldsymbol{\delta}$), and Mach number (Ma). Altitude is coupled with the Mach number, such that the appropriate altitude is selected based on Table 2 for a given Mach number. If all $\boldsymbol{\delta}$ values were used as variables in the multivariate spline, the model's complexity would increase rapidly. Therefore, the on-board model employs two separate methods. First, for each Mach number (Mach = 0.2, 0.3, 0.4, 0.5, 0.6, 0.7), a cubic univariate spline is fitted for $C_L - \alpha$, $C_D - \alpha$, and $C_M - \alpha$. Linear interpolation is then used to estimate the splines for intermediate Mach numbers. To account for the influence of the actuators, estimates for C_{L_δ}, C_{D_δ}, $C_{D_{\delta^2}}$, and C_{M_δ} are generated based on the aerodynamic data. This strategy mirrors the approach used to generate the aerodynamic data of the onboard model; nevertheless, in a real-world scenario, the aerodynamic data can be directly obtained during flight.

A cubic spline is defined by a cubic polynomial on each interval $[x_i, x_{i+1}]$, such that:

$$S(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i, \quad \text{for} \quad x_i \leq x < x_{i+1} \tag{4}$$

where a_i, b_i, c_i, d_i are the coefficients of the cubic polynomial on the i-th interval. For the spline to be smooth, it must be continuous on three conditions. Firstly, the spline function must be continuous at each knot to ensure the function remains gap-free: $S(x_i^-) = S(x_i^+),\ \forall i$. Secondly, the first derivative of the spline must be continuous at each knot to ensure no abrupt changes of the slope: $S'(x_i^-) = S'(x_i^+),\ \forall i$. Lastly, the second derivative must be continuous at each knot to ensure smooth curve transitions: $S''(x_i^-) = S''(x_i^+),\ \forall i$.

To fit the on-board model, the system must be excited, for which several methods are discussed in the literature. Ref. [3] proposed applying bounded randomized inputs to the model, after which the on-board model is trained. Ref. [1] introduced the sweep excitation method, which deflects all control surfaces in an ordered sweeping motion, offering greater computational efficiency and time savings compared to the random excitation method. Additionally, Ref. [18] proposed a wandering phase method, which employs pseudo-random sample points using a Sobol sequence [26]. The principle behind all excitation methods is that they are performed once during a test flight, after which the model is fine-tuned and adapted during real flights. If the model remains accurate following the previous flight, further excitations may not be necessary. This study adopts the sweep method due to its computational efficiency, while acknowledging that any of the aforementioned methods could be suitable for the proposed optimization framework. Assuming a maximum actuator velocity of 45 deg/s [12], with the actuators starting and ending at the zero position, it would take 1.42 seconds per actuator to perform the sweep excitation method. This assumes that the sampling rate of the aerodynamic coefficients is significantly higher than the excitation frequency of the actuators.

Using the sweep method, a dataset was generated from the aerodynamic model, to which the on-board model was fitted. Figure 5 illustrates the influence of Mach number and angle of attack on the aerodynamic coefficients. Markers in the figure indicate points from the dataset to which a spline is fitted. The moment coefficient has been calculated using the center of gravity from Table 1. The nonlinear and non-convex behavior is clearly visible in the C_M-α graph, where the unstable pitch break phenomenon appears at high angle of attack. In addition, the pitch break margin also decreases as Mach number increases. Nonlinearities also emerge in the lift and drag coefficients. Figure 6 depicts the influence of actuator deflections on the aircraft when Mach number is 0.7. Upward deflection of the actuators decreases the lift coefficient but increases the moment coefficient, while downward deflection has the opposite effect. The drag coefficient increases symmetrically with both upward and downward deflection, as $C_{D_\delta} \approx 0$. It is noteworthy that, for clarity of presentation, only a subset of the data is shown in figures 5 and 6. The actual dataset used is considerably more extensive than what is displayed.

A crucial requirement for an online optimizer is that the on-board model must be capable of adapting during flight. This implies that trustworthy fresh data points should update the model, even if they sometimes contradict the existing model. To facilitate this, a *forgetting mechanism* is incorporated into the on-board model. If a new data point is within a

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

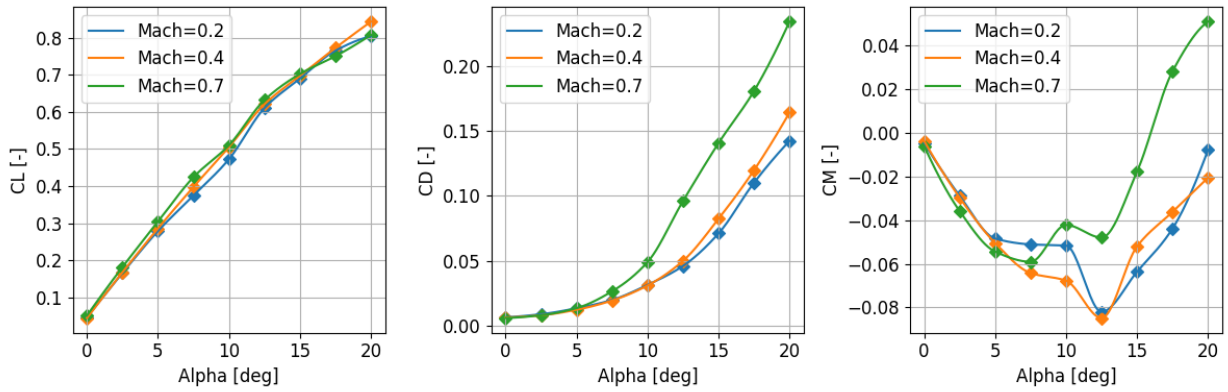


Fig. 5 Impacts of Mach number on lift, drag, and pitching moment coefficients.

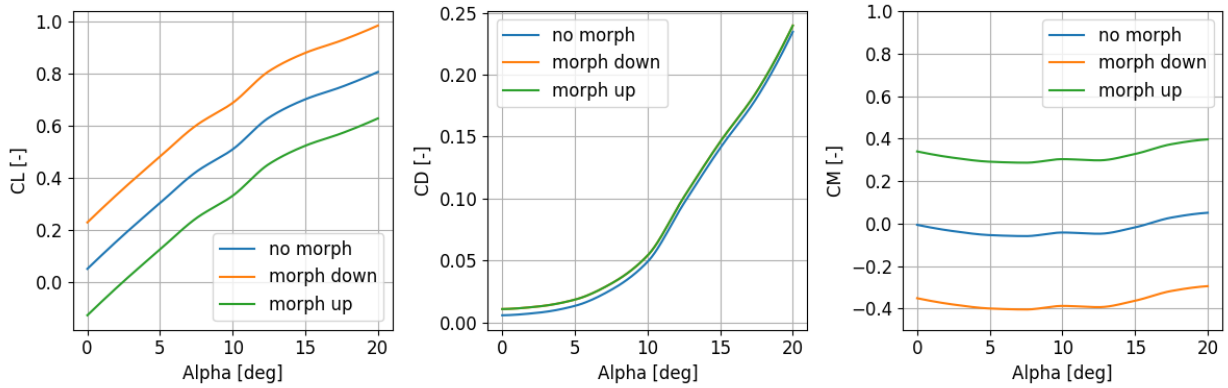


Fig. 6 Influence of actuator deflections on aerodynamic coefficients at $\delta = -16$ deg, $\delta = 0$ deg, and $\delta = 16$ deg.

small distance of an existing one, the older point is discarded, and the spline is refitted. This adaptation is instantaneous for each available, reliable data point. It does not need to wait for the collection of a data batch, as is typically required by neural networks or polynomial models. Figure 7 illustrates the effectiveness of updating the model from an initial configuration. The aerodynamic coefficients, C_L, C_D, and C_M, are plotted against the angle of attack. The model is then updated using new, randomly generated data points, which consist of vectors of actuator deflections and angle of attack. These figures demonstrate that the model can effectively adjust its structure based on the incorporation of new data.

D. Optimizer

To solve the problem formulated in subsection II.A, an optimizer is designed. Optimizers can generally be classified as either local or global. From a theoretical point of view, the Flying-V aerodynamic model is nonlinear and non-convex. To confirm this, we first applied a local gradient-based optimization algorithm with random initial conditions, which indeed reveals various local optima. As a result, local optimization is not suitable, and a global optimizer is required. The nonlinear and non-convex property of the optimization problem is further revealed in Fig. 8a. When there is only one optimization variable, namely the angle of attack, the cost function already shows two local minima. Moreover, the cost function becomes more complicated when the uniformed morphing along the span is added as an additional optimization variable. Furthermore, when distributed morphing present, the cost function is expected of becoming more complicated, preserving the nonlinear and non-convex properties, motivating the choice of gradient-free global optimization.

Numerous global optimization strategies exist, each with distinct strengths and weaknesses. Selecting the most appropriate depends on the specific characteristics and behavior of the cost function. As previously mentioned, the on-board model is non-convex and, being spline-based, is a smooth function. Reference [27] studied various optimization

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

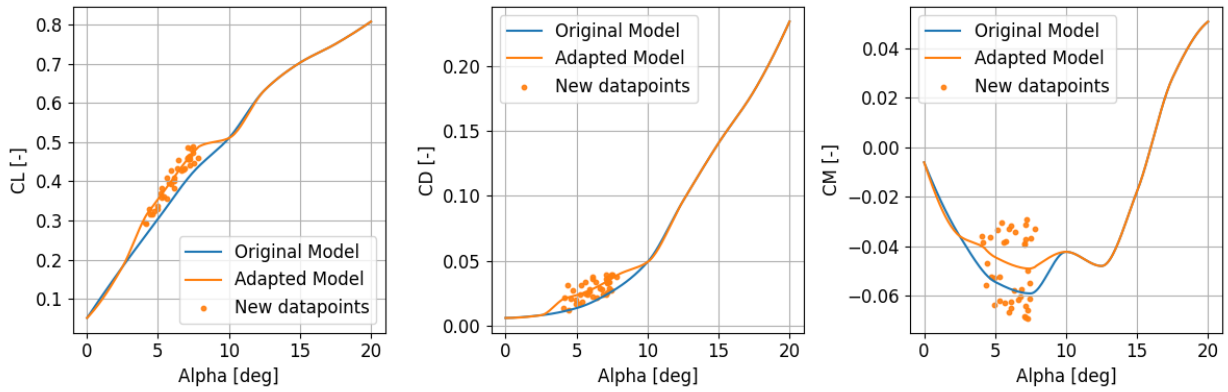
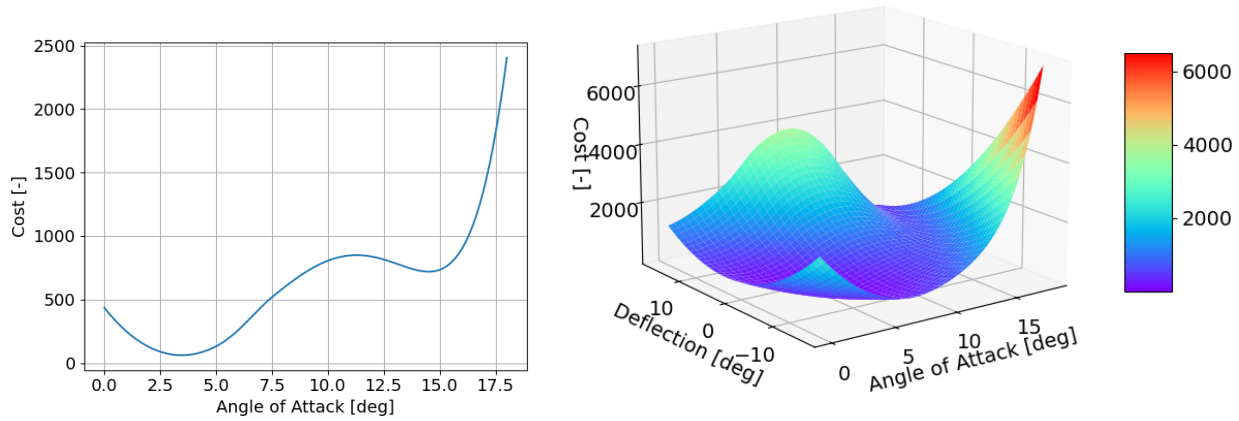


Fig. 7 On-board model adaptation to 40 new datapoints at Mach = 0.7.



(a) Cost function against the angle of attack. **(b) Cost function against the angle of attack and uniformed morphing.**

Fig. 8 Cost function behavior when one or two independent optimization variables present.

algorithms for non-convex and smooth cost functions. Based on this study, the tradeoff will consider on Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [7], Bound Optimization by Quadratic Approximation (BOBYQA) [28], Dividing RECTangles (DIRECT) [29], and the particle swarm pattern search method (PYSWARM) [30] as viable optimization methods.

The optimizers are compared based on performance, the number of required evaluations, and sensitivity. The results of this comparison are shown in Fig. 9a. CMA-ES and BOBYQA converge to the same optimum; however, BOBYQA achieves this with fewer required function evaluations. In contrast, DIRECT and PYSWARM, which also show highly variable results, both converge to a lower optimal value.

When updating the on-board model, more complex behavior emerges due to the potential for nonlinear and non-convex behavior in local regions. To prevent significant deviations from the optimum, the optimizers must return results with low variability. By using an initial coarse on-board model and introducing random data points, the variability of the solutions can be analyzed. As illustrated in Fig. 9b, CMA-ES and BOBYQA performed similarly. Despite that, BOBYQA converges with much fewer function evaluations. Therefore, this study considers BOBYQA the most suitable optimizer for this application.

BOBYQA is a derivative-free optimization algorithm designed for solving smooth, bound-constrained, nonlinear optimization problems. It belongs to the class of trust-region algorithms, meaning it iteratively constructs local quadratic approximations of the objective function within a defined “trust region” around the current solution. The BOBYQA algorithm initializes using samples of the cost function, usually $n + 2$ (where n is the number of function variables) and

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

then approximates the objective function ($f(x)$) using a quadratic model ($Q(x)$), which can be written as:

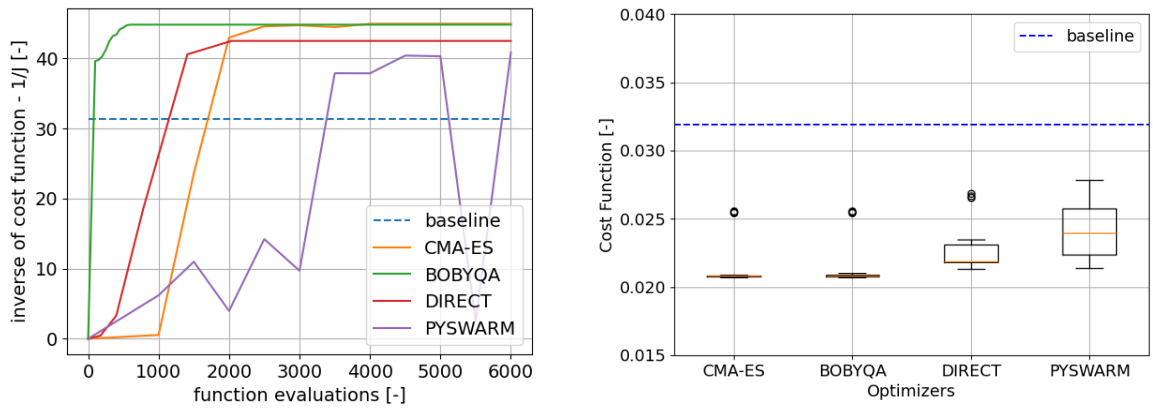
$$Q(x) = c + g^\top (x - x_0) + \frac{1}{2}(x - x_0)^\top H (x - x_0) \tag{5}$$

where: x_0 is the current point (the center of the trust region); g is the gradient approximation; H is the Hessian approximation; c is a constant term. The algorithm minimizes the quadratic model within a trust region defined by the radius Δ_k. The sub-problem to solve is:

$$\min Q(x) \quad \text{subject to} \quad \|x - x_k\| \leq \Delta_k \tag{6}$$

where Δ_k is the radius of the trust region, and x_k is the current best estimate. If the solution to the sub-problem improves the objective function sufficiently, the trust region is expanded; otherwise, it is shrunk. After each iteration, the algorithm uses new function evaluations to update the quadratic model $Q(x)$ around the new current best estimate.

BOBYQA is particularly well-suited for problems involving noisy data, nonlinear smooth functions, bound constraints, and low to medium dimensionality. While heuristic methods perform well in cases with a large number of optimization variables and a broad search space, this specific use case involves only six optimization variables and a well-defined search space. Moreover, because BOBYQA employs a systematic optimization approach, it typically achieves faster convergence compared to heuristic methods. Using the numerical results from [28], it can be estimated that BOBYQA requires $O(n \cdot \log(n))$ function evaluations, where n is the number of optimization variable. Regarding computational efficiency, the paper reported computation times scaling as $O(n^2)$. Since the optimization problem in this study involves less than 10 independent variables, the time constraints imposed by this scaling are not expected to significantly impact performance.



(a) Optimizer performance against the number of function evaluations.

(b) Variability in optimal cost for different optimizers when adapting the on-board model with new datapoints.

Fig. 9 Comparison of various global optimizers in terms of performance, efficiency and sensitivity.

E. Fault tolerance of actuator faults

The aim of fault tolerance is to detect any possible actuator failure based on fresh data and adjust the remaining actuators so that the system remains trimmed and optimized for minimal drag.

Figure 10 shows how the system identifies and optimizes in presence of an actuator failure. A checking function is designed to judge if the aircraft is actually trimmed. If an unforeseen actuator failure presence, a mismatch between the onboard model and reality would appear, leading to a derivation from the real optimum.

If the aircraft is not trimmed correctly, despite the optimizer still thinking the aircraft is in an optimum based on existing knowledge, the proposed algorithm would first identify the faulty actuator through a new model excitation sequence. If the actuator is stuck, then the identified C_{L_δ} would be approximately zero. Once the system has identified the faulty actuator, the splines will be adjusted to incorporate this fixed point, ensuring that the faulty actuator remains in the same position in the model while ensuring spanwise smoothness. The algorithm will then determine the optimal

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

positions for the remaining actuators, taking into account the position of the faulty actuator. Figure 11 illustrates how the splines are modified to accommodate the faulty actuator. The dashed line represents the faulty actuator location, and it is evident how the splines consistently pass through this fixed point while allowing the other actuators to remain variable.

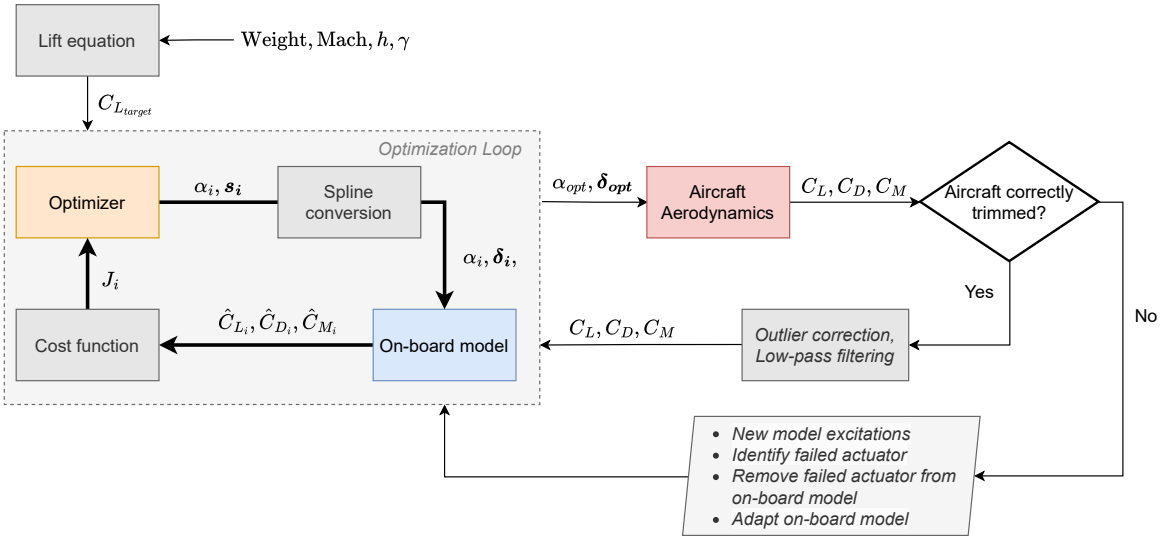


Fig. 10 The optimization architecture that can identify and tolerate actuator faults.

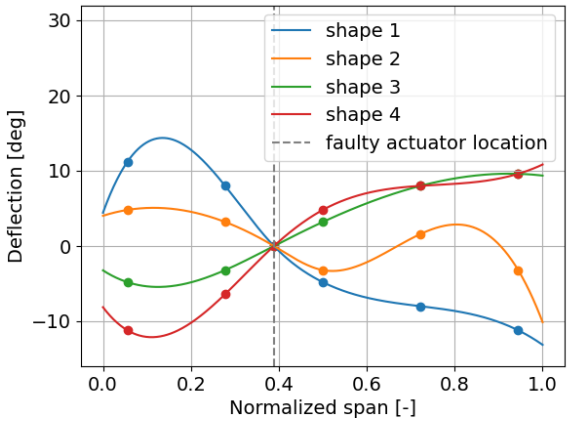


Fig. 11 Illustration of actuator deflections based on the spline curve accounting for a faulty actuator.

III. Results

In this section, the effectiveness of the proposed optimization architecture is presented. Subsection III.A determines the optimal number of actuators considering the tradeoff between performance and weight penalty. Subsection III.B presents the optimization effectiveness on various flight conditions. Subsection III.C presents the fault tolerance results. Subsection III.D evaluates the overall performance improvement in one typical flight profile.

A. Ideal number of morphing actuators

In conventional aircraft design, each wing typically has only one control surface, namely the aileron, responsible for roll control. However, as aircraft design evolves, we are seeing an increasing number of control surfaces on aircraft

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

wings. The more control surfaces available, the greater the degrees of freedom for control, which allows for more precise manipulation of local aerodynamic properties and potential improvements in aerodynamic performance. Recently, Boeing has expressed interest in distributed trailing-edge control surfaces [11]. However, adding more control surfaces also results in a weight penalty. This subsection aims to quantify this tradeoff.

A previous study on continuous trailing-edge actuators estimated the actuator weight to be approximately 20 kg, based on calculations for both the servo-actuator weight and the structural weight [31]. However, since the tradeoff is highly dependent on the actuator weight, the efficiency benefits provided by the actuators relative to the added weight are shown in Fig. 12 for three different actuator weight scenarios.

It can be seen from Fig. 12 that as the number of actuators increases, the drag coefficient initially decreases significantly, before exhibiting a moderate upward trend. The optimal number of actuators is 9 when the single actuator weight is $F_{\text{weight}} = 10$ kg and changed to 6 for $F_{\text{weight}} = 20$ kg and $F_{\text{weight}} = 40$ kg. Considering the slopes in Fig. 12, the optimal number of actuators can be relaxed to 8 when $F_{\text{weight}} = 10$ kg and to 5 when $F_{\text{weight}} = 40$ kg without experiencing significant performance loss. In general, as the weight of each individual actuator increases, the optimal number of actuators decreases, while the drag coefficient increases. This is explained by the dependency of $C_{L_{\text{target}}}$ on aircraft weight (Eq. (3)). Namely, when the aircraft becomes heavier, higher angle of attack is required to trim the aircraft, which results in increased lift-induced drag.

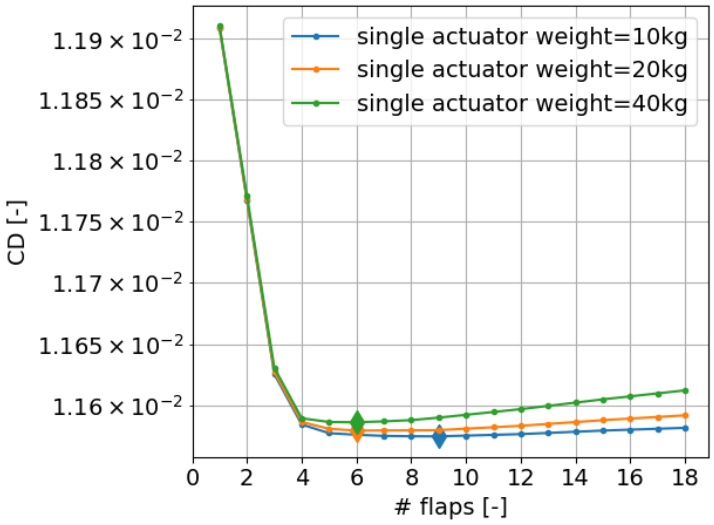


Fig. 12 Drag reduction effectiveness against the number of actuators (diamonds marks the minima).

B. Optimization performance and computational efficiency

This subsection presents the performance and computational efficiency of the proposed optimization architecture. First, at one specific cruise condition, three approaches are compared: 1) the baseline, which is the practical state-of-the-art that only uses one single flap/actuator on each wing for trim; 2) the gradient-based optimizer based on the Nelder-Mead method; 3) the proposed global optimization method BOBYQA. The latter two approaches both use a wing with 9 actuators, and a actuator weight of 10 kg. For a fair comparison, the flight condition and control surface area for the three methods are identical.

The on-board model is initially configured with a sweeping model excitation to get the aerodynamic coefficients at Mach = 0.7 and $h = 9750$ m. Then, the optimization is run on the on-board surrogate model. The number of iterations is limited to 1300 or until the minimal tolerance is met. Since the minimum tolerance is met in both cases, running the same experiment for more iterations will not change the final values. As seen from Fig. 13, both Nelder-Mead and BOBYQA present an improvement as compared to the baseline. However, the gradient-based Nelder-Mead optimization method is not capable of converging to the global optimal solution and is prone to be trapped in local optima. The spikes in the BOBYQA method are the exploration phases of the approximation regions.

The comparison with the baseline is further detailed in Fig. 14. Since this baseline approach only has two inputs (α,

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

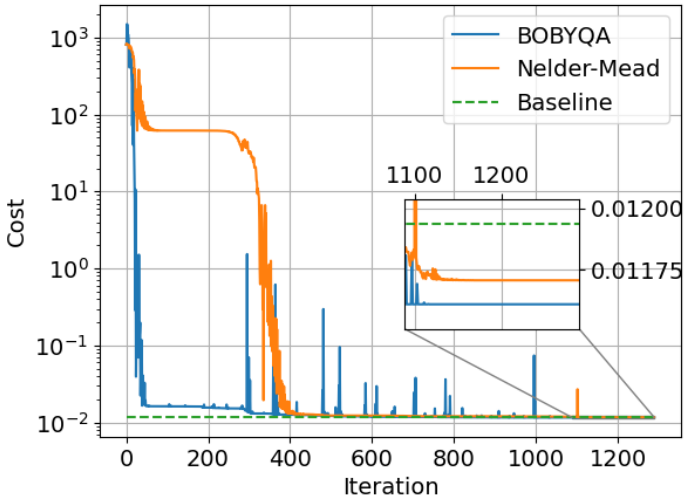


Fig. 13 Optimization performance comparisons at a cruise condition.

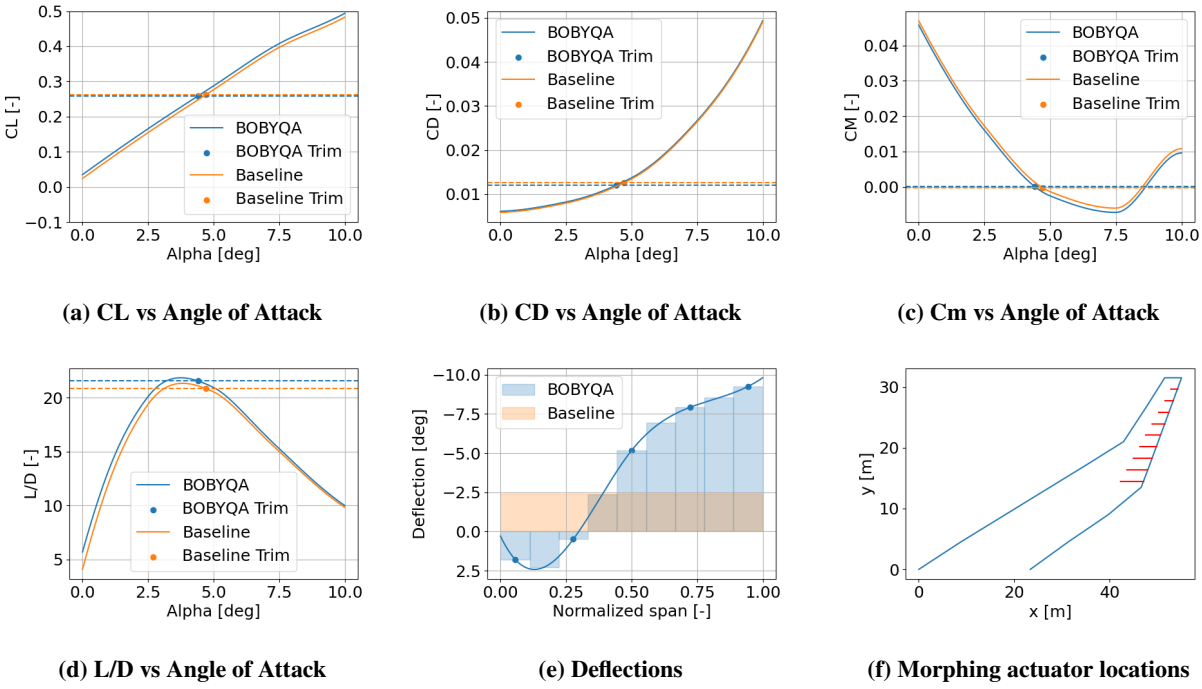
and one single actuators) to satisfy both lift and moment trim, there is only one feasible solution. It can be seen that the optimized solution has a higher L/D than the simple trim solution, while keeping both the moment and lift coefficient at the intended value. At the flight condition of $C_{L_{\text{target}}} = 0.26$ and Mach = 0.7, the proposed approach can increase the aerodynamic efficiency (L/D) by 2.98 %. This is by virtue of the distributed actuators configuration that can modify the aerodynamic profile regionally. Specifically, the outboard actuators have a larger moment arm to the center of gravity, while having a smaller influence on lift and drag. The optimizer can therefore use more of the outboard actuators to trim without over-deflecting the inner actuators, which minimizes the drag. In addition, having the inner actuators be able to deflect downwards increases the lift and therefore lowers the angle of attack, which is also beneficial for reducing lift-induced drag.

After achieving a 2.98 % of aerodynamic efficiency enhancement at $C_{L_{\text{target}}} = 0.26$ and Mach = 0.7, the effectiveness of the proposed approach on various flight conditions is then evaluated in Fig. 15. The target lift coefficient is an indication of flight conditions and is related to aircraft weight, fuel burn, and flight phases. It can be seen from Fig. 15a that the optimized solution outperforms the baseline at all the tested conditions. Moreover, the optimizer gains more as the target lift coefficient increases, reaching an increase 8.19% at $C_{L_{\text{target}}} = 0.4$. When going to even higher target lift coefficients, the L/D ratio increase even reaches 12.6%. The main reason for this is that at lower target lift coefficients the system is easier to be trim with just one actuator because the ideal shape, as can be seen in Fig. 15b, is more similar to a horizontal straight line. Conversely, at higher target lift coefficients, a single-flap system loses excessive lift in its attempt to maintain a zero C_M, necessitating a higher angle of attack, which in turn increases drag.

Figure 16 illustrates the increase in L/D ratio and the corresponding actuator configurations for different Mach numbers. The L/D ratio increases at higher Mach numbers, even in the baseline case. The optimization benefits remain relatively consistent across the range of Mach numbers. It is important to note that under identical flight conditions, the $C_{L_{\text{target}}}$ increases at lower Mach numbers due to the reduction in speed. As shown in Fig. 15, a higher $C_{L_{\text{target}}}$ would lead to greater increases in the L/D ratio. However, to ensure a meaningful comparison across different Mach numbers, the $C_{L_{\text{target}}}$ is maintained at a constant value. Figure 16b indicates that the actuators require less upward morphing at higher Mach numbers because of the increase in control effectiveness.

The optimization computations were performed on a laptop equipped with an Intel Core i7-8750H processor (6 cores, 12 logical processors, base clock 2.20 GHz), 16 GB DDR4 RAM, and a 512 GB NVMe SSD. The system ran Microsoft Windows 11 with Python 3.9.18. Using this setup, the BOBYQA optimization converged in an average CPU time of 11.63 seconds and an average wall clock time of 11.79 seconds, as measured across multiple runs for different Mach numbers and lift coefficients as found in the typical flight profile.

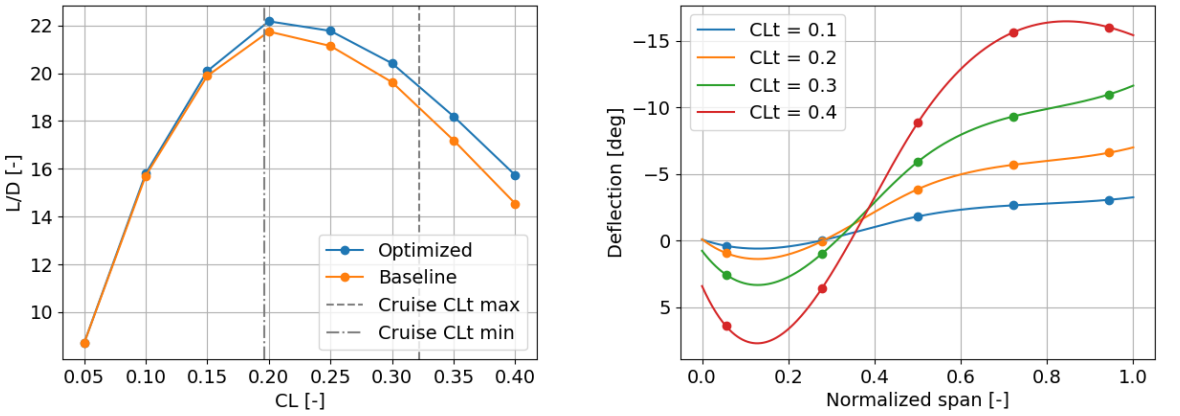
Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804



(a) CL vs Angle of Attack **(b) CD vs Angle of Attack** **(c) Cm vs Angle of Attack**

(d) L/D vs Angle of Attack **(e) Deflections** **(f) Morphing actuator locations**

Fig. 14 Optimization results compared to the baseline for $C_{L_{\text{target}}} = 0.26$, Mach = 0.7



(a) Optimization for various $C_{L_{\text{target}}}$ during cruise. **(b) Optimal shape for different $C_{L_{\text{target}}}$ during cruise.**

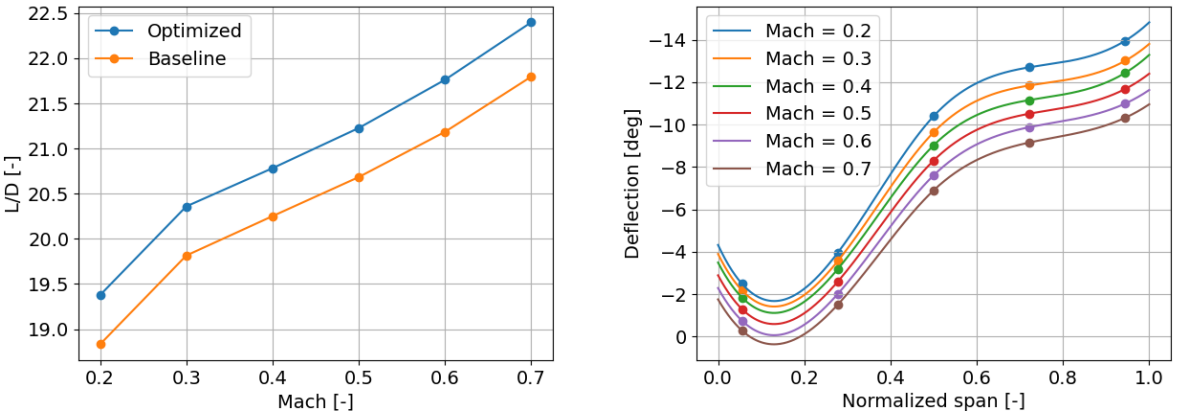
Fig. 15 Optimization performed for various target lift coefficients.

C. Fault tolerant optimization

In the presence of an unforeseen actuator fault, the system should still be able to optimize and trim the aircraft. This subsection presents the results of a case study where the system is optimized on the original on-board model while there is actually an unforeseen actuator fault present. The faulty actuator is simulated by setting the C_{L_δ}, C_{M_δ}, and $C_{D_{\delta^2}}$ in the original dataset to zero.

Figure 17 compares the results with and without fault-tolerant optimization. It shows that because of the lack of knowledge of the fault, the spline curve for the system without fault tolerance does not go through the failed actuator's actual position, leading to an untrimmed and non-optimal solution as highlighted in Fig. 17c. By contrast, the algorithm proposed in subsection III.C is able to automatically identify which actuator is stuck as well as the jammed position,

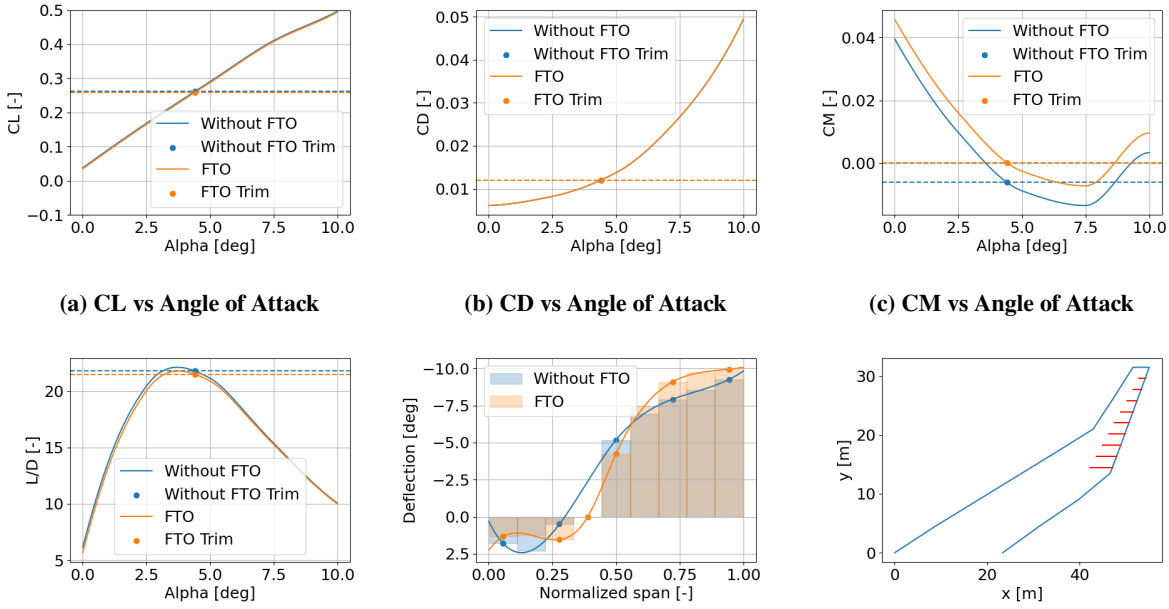
Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804



(a) Optimization for different Mach numbers. **(b) Optimal shape for different Mach numbers.**

Fig. 16 Optimization performed for various Mach numbers when $C_{L_{\text{target}}} = 0.26$.

thus still fulfills the optimization and trim requirements effectively.



(a) CL vs Angle of Attack **(b) CD vs Angle of Attack** **(c) CM vs Angle of Attack**

(d) L/D vs Angle of Attack **(e) Deflections** **(f) Morphing actuator locations**

Fig. 17 Optimized results for fault tolerant optimization (FTO) when $C_{L_{\text{target}}} = 0.26$, Mach = 0.7

D. Effectiveness on a typical flight cycle

In subsection III.B, the effectiveness of the proposed method on various flight conditions has been demonstrated. Depending on the flight condition, the algorithm can achieve up to 12.6% of increase in aerodynamic efficiency (L/D) and up to 12.6% of drag reduction (C_D). Nevertheless, during one typical flight cycle, an aircraft experiences different durations in different flight conditions. For example, for most of the time, the aircraft is cruising. This subsection aims to evaluate the effectiveness of the proposed method during one typical flight cycle.

A typical flight profile is considered based on data provided in Ref. [32] for the Airbus A350-9000. The results are

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

presented in Table 3. The time per flight phase has been calculated using rate of climb/descent data from Ref. [32]. Without loss of generality, this research considers a ten-hour flight. The drag reductions is calculated for every flight phase, and a time-weighted average is calculated, resulting in a drag reduction of 2.97% in one typical flight cycle.

For jet aircraft, fuel consumption is proportional to the thrust required, so if drag decreases, thrust and fuel consumption decrease proportionally. An estimation on the fuel savings can be made using the data from Table 3. Assuming the Airbus A350-9000 consumes around 5000 kg/h of fuel, this would result in 150 kg/h of fuel saving in one typical flight cycle. For the typical flight mentioned in Table 3, this leads to a total of 1500 kg fuel savings during cruise, and 72.5 kg in other flight phases.

Table 3 Full flight savings for a typical flight profile with 10 hour cruise. Flight profile is based on [32].

	Time [min]	Mach	$C_{L_{\text{target}}}$ estimate	C_D baseline	C_D optimized	ΔC_D %
Take-off	1.00	0.2	1.03	0.1658	0.1654	0.24
Initial Climb	1.67	0.4	0.40	0.0225	0.0217	3.61
Climb to cruise	9.97	0.55	0.32	0.0161	0.0155	3.35
Cruise	600.00	0.7	0.26	0.0124	0.0120	2.98
Descent	10.57	0.55	0.20	0.0096	0.0094	2.13
Approach	6.67	0.4	0.25	0.0121	0.0118	2.49
Landing	1.00	0.2	0.65	0.0722	0.0675	6.44
Total	**630.88**	**-**	**-**	**-**	**-**	**2.97**

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

IV. Conclusions

This paper proposes a global online data-driven aerodynamic performance optimization framework, which can efficiently and smoothly model nonlinearities, adapt to changes on-board, and automatically detect and adapt to actuator faults.

Quantitive comparison results based on a free-flying Flying-V aircraft model confirm that the adopted data-driven Bound Optimization by Quadratic Approximation algorithm outperforms gradient-based local as well as other considered global optimization methods. Moreover, a multivariable spline-based on-board model effectively captures the nonlinearities of the Flying-V model and adapts to new data. Furthermore, the framework shows effectiveness in determining the optimal number of flaps considering the tradeoff between aerodynamic performance and weight penalty.

By optimizing the angle of attack and actuator positions, the proposed online optimization framework can achieve a drag reduction of 1.9 % to 4.9 % during cruise. At higher target lift coefficients (caused by heavier weight and lower speed), the drag reduction can reach up to 12.6 %. This framework has shown effectiveness throughout the entire flight envelope. For a typical flight cycle of ten hours, it can achieve a 2.97 % of drag reduction and which is approximately 150 kg/h of fuel. Last but not least, in the presence of unforeseen actuator faults, the proposed algorithm is able to automatically identify, adapt, trim, and optimize.

One limitation of this research is that high-lift devices used during takeoff and landing are not modeled. In future work, these can be incorporated into the framework either as unknown optimization variables or as known inputs.

In conclusion, the proposed online aerodynamic performance optimization framework demonstrates significant drag reduction and fault-tolerance capabilities. This advancement represents a crucial step towards more sustainable aviation, contributing to reduced fuel consumption and lower emissions in future aircraft designs.

Appendix

Table 4 presents the coefficients in relation to the actuator deflection from the wind tunnel data and VLM data and presents the correction factors for each section. Wind tunnel data was gathered for a three-actuator layout [15], where the controllable surface is divided into three individual sections. Any actuator of the VLM simulations that fall within the geometric bounds of a wind tunnel actuator configuration will be corrected in the same manner.

Table 4 Correction factors for control effectiveness.

Coefficient	Wind tunnel	VLM	Correction Factors
$C_{L_{\delta_1}}$	2.59e-3	5.02e-3	0.52
$C_{L_{\delta_2}}$	1.88e-3	2.13e-3	0.88
$C_{L_{\delta_3}}$	1.18e-3	1.22e-3	0.97
$C_{D_{\delta_1^2}}$	1.32e-5	1.05e-5	1.25
$C_{D_{\delta_2^2}}$	6.75e-6	3.99e-6	1.69
$C_{D_{\delta_3^2}}$	4.89e-6	2.70e-6	1.81
$C_{M_{\delta_1}}$	-2.80e-3	-6.21e-3	0.45
$C_{M_{\delta_2}}$	-2.41e-3	-3.73e-3	0.65
$C_{M_{\delta_3}}$	-1.63e-3	-2.65e-3	0.62

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804

Acknowledgments

The authors would like to acknowledge Professor Roelof Vos for sharing the Flying-V aerodynamic model. The authors also would like to thank Professor Erik-Jan van Kampen and Professor Coen de Visser for their suggestions.

References

[1] Nguyen, N., Cramer, N., Hashemi, K., Ting, E., Drew, M., Wise, R., Boskovic, J., Precup, N., Mundt, T., and Livne, E., "Real-time adaptive drag minimization wind tunnel investigation of a flexible wing with variable camber continuous trailing edge flap system," *AIAA Aviation 2019 Forum*, 2019. https://doi.org/10.2514/6.2019-3156.

[2] Nguyen, N., "Elastically Shaped Future Air Vehicle Concept," Tech. rep., NASA Innovative Partnerships Program Office, October 2010. https://ntrs.nasa.gov/citations/20110023698.

[3] Ferrier, Y., Nguyen, N., and Ting, E., "Real-time adaptive least-squares drag minimization for performance adaptive aeroelastic wing," *34th AIAA Applied Aerodynamics Conference*, 2016. https://doi.org/10.2514/6.2016-3567.

[4] Ting, E., Chaparro, D., Nguyen, N., and Fujiwara, G. E. C., "Optimization of Variable-Camber Continuous Trailing-Edge Flap Configuration for Drag Reduction," *Journal of Aircraft*, Vol. 55, No. 6, 2018, p. 2217–2239. https://doi.org/10.2514/1.c034810.

[5] Mkhoyan, T., Thakrar, N. R., Breuker, R. D., and Sodja, J., "Design and development of a seamless smart morphing wing using distributed trailing edge camber morphing for active control," *AIAA Scitech 2021 Forum*, 2021. https://doi.org/10.2514/6.2021-0477.

[6] Mkhoyan, T., Ruland, O., De Breuker, R., and Wang, X., "On-Line Black-Box Aerodynamic Performance Optimization for a Morphing Wing With Distributed Sensing and Control," *IEEE Transactions on Control Systems Technology*, Vol. 31, No. 3, 2023, pp. 1063–1077. https://doi.org/10.1109/TCST.2022.3210164.

[7] Hansen, N., and Ostermeier, A., "Completely derandomized self-adaptation in evolution strategies." *Evolutionary computation*, Vol. 9, No. 2, 2001, pp. 159–195. https://doi.org/10.1162/106365601750190398.

[8] Vassberg, J. C., DeHaan, M. A., Rivers, S. M., and Wahls, R. A., "Development of a common research model for applied CFD validation studies," *AIAA Applied Aerodynamics Conference*, 2008. https://doi.org/10.2514/6.2008-6919.

[9] Nguyen, N., and Xiong, J., "Real-time drag optimization of aspect ratio 13.5 common research model with distributed flap system," *AIAA Scitech 2021 Forum*, 2021. https://doi.org/10.2514/6.2021-0069.

[10] Faggiano, F., Vos, R., Baan, M., and Van Dijk, R., "Aerodynamic design of a flying V aircraft," *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017. https://doi.org/10.2514/6.2017-3589.

[11] Xiong, J., Nguyen, N. T., and Bartels, R. E., "Numerical Simulation of An Aspect Ratio 13.5 Common Research Model with Trailing Edge Mini-Flaps," *AIAA Aviation 2023 Forum*, 2023. https://doi.org/10.2514/6.2023-4525.

[12] Hanke, C. R., and Nordwall, D. R., "The Simulation of a Large Jet Transport Aircraft. Volume II: Modeling Data," *NASA Contractor Report*, Vol. CR-114494, 1970. https://ntrs.nasa.gov/citations/19730001300.

[13] Wang, X., Mkhoyan, T., Mkhoyan, I., and De Breuker, R., "Seamless active morphing wing simultaneous gust and maneuver load alleviation," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 9, 2021, pp. 1649–1662. https://doi.org/10.2514/1.G005870.

[14] TU Delft, Faculty of Aerospace Engineering, "Flying-V Project," https://www.tudelft.nl/en/ae/flying-v, 2022. [Accessed: May 16, 2024].

[15] Asaro, S., and Vos, R., "Synthesis of the Aerodynamic Model of a Flying Wing Aircraft," *AIAA Scitech 2025 Forum*, 2025.

[16] McDonald, R. A., and Gloudemans, J. R., "Open vehicle sketch pad: An open source parametric geometry and analysis tool for conceptual aircraft design," *AIAA Science and Technology Forum and Exposition, AIAA Scitech Forum 2022*, 2022. https://doi.org/10.2514/6.2022-0004.

[17] Gomroki, M. M., Topputo, F., Bernelli-Zazzera, F., and Tekinalp, O., "Solving constrained optimal control problems using state-dependent factorization and chebyshev polynomials," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 3, 2018, pp. 618–631. https://doi.org/10.2514/1.G002392.

[18] Ruland, O., Mkhoyan, T., De Breuker, R., and Wang, X., "Black-Box Online Aerodynamic Performance Optimization for a Seamless Wing with Distributed Morphing," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 3, 2023, pp. 560–570. https://doi.org/10.2514/1.G006573.

[19] de Zoeten, G. J., Varriale, C., and Vos, R., "Flight Performance Evaluation of the Flying-V," *AIAA Aviation 2023 Forum*, 2023. https://doi.org/10.2514/6.2023-3484.

[20] Tol, H. J., De Visser, C. C., Van Kampen, E., and Chu, Q. P., "Nonlinear multivariate spline-based control allocation for high-performance aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1840–1862. https://doi.org/10.2514/1.G000065.

[21] Calise, A. J., and Rysdyk, R. T., "Nonlinear Adaptive Flight Control Using Neural Networks," *IEEE Control Systems*, Vol. 18, No. 6, 1998, pp. 14–25. https://doi.org/10.1109/37.736008.

[22] Calise, A. J., Lee, S., and Sharma, M., "Development of a Reconfigurable Flight Control Law for Tailless Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 5, 2001, p. 896–902. https://doi.org/10.2514/2.4825.

[23] Babuška, R., and Verbruggen, H., "Neuro-fuzzy methods for nonlinear system identification," *Annual Reviews in Control*, Vol. 27 I, No. 1, 2003, pp. 73–85. https://doi.org/10.1016/S1367-5788(03)00009-9.

[24] Sun, L. G., de Visser, C. C., Chu, Q. P., and Mulder, J. A., "Online Aerodynamic Model Identification Using a Recursive Sequential Method for Multivariate Splines," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 5, 2013, pp. 1278–1288. https://doi.org/10.2514/1.60375.

[25] Ruiz-García, A., Vos, R., and de Visser, C. C., "Aerodynamic Model Identification of the Flying V from Wind Tunnel Data," *AIAA Aviation 2020 Forum*, 2020. https://doi.org/10.2514/6.2020-2739.

[26] Sobol, I., "On the distribution of points in a cube and the approximate evaluation of integrals," *USSR Computational Mathematics and Mathematical Physics*, Vol. 7, No. 4, 1967, pp. 86–112. https://doi.org/10.1016/0041-5553(67)90144-9.

[27] Rios, L. M., and Sahinidis, N. V., "Derivative-free optimization: A review of algorithms and comparison of software implementations," *Journal of Global Optimization*, Vol. 56, No. 3, 2013, pp. 1247–1293. https://doi.org/10.1007/s10898-012-9951-y.

[28] Powell, M. J. D., "The BOBYQA Algorithm for Bound Constrained Optimization without Derivatives," Tech. rep., University of Cambridge, Department of Applied Mathematics and Theoretical Physics, January 2009. https://www.damtp.cam.ac.uk/user/na/NA_papers/NA2009_06.pdf.

[29] Jones, D. R., Perttunen, C. D., and Stuckman, B. E., "Lipschitzian optimization without the Lipschitz constant," *Journal of Optimization Theory and Applications*, Vol. 79, No. 1, 1993, pp. 157–181. https://doi.org/10.1007/BF00941892.

[30] Vaz, A. I. F., and Vicente, L. N., "A particle swarm pattern search method for bound constrained global optimization," *Journal of Global Optimization*, Vol. 39, No. 2, 2007, pp. 197–219. https://doi.org/10.1007/s10898-007-9133-5.

[31] Xiong, J., Bartels, R. E., and Nguyen, N., "Aerodynamic optimization of mach 0.745 transonic truss-braced wing aircraft with variable-camber continuous trailing-edge flap," *AIAA Scitech 2021 Forum*, 2021. https://doi.org/10.2514/6.2021-0337.

[32] EUROCONTROL, "Aircraft Performance Database: Airbus A350-900 (ICAO: A359)," https://contentzone.eurocontrol.int/aircraftperformance/details.aspx?ICAO=A359, 2018. [Accessed: September 18, 2024].

Downloaded by Technische Universiteit Delft on January 6, 2025 | http://arc.aiaa.org | DOI: 10.2514/6.2025-2804