https://doi.org/10.1590/jatm.v18.1439 ORIGINAL PAPER

Effects of Non-Uniform Data Coverage on Deep Neural Network-Based Scramjet Performance Prediction

Ângelo de Carvalho Paulino[1,*], Pedro Paulo Batista de Araújo[2], Roberto Yuji Tanaka[2], Angelo Passaro[1,2]

1. Departamento de Ciência e Tecnologia Aeroespacial – Instituto Tecnológico de Aeronáutica – Divisão de Pós-Graduação e Pesquisa – São José dos Campos/SP – Brazil.
2. Departamento de Ciência e Tecnologia Aeroespacial – Instituto de Estudos Avançados – Divisão de Aerotermodinâmica e Hipersônica – São José dos Campos/SP – Brazil.

*Corresponding author: angeloacp@fab.mil.br

ABSTRACT

The development of hypersonic vehicles increasingly relies on deep learning (DL) models; however, their reliability depends on the coverage of training data. When training datasets exhibit uneven coverage of the operational domain, predictions may retain high statistical accuracy while losing physical meaning. This study examines how a deep neural network (DNN) responds to such uneven coverage in predicting thrust for supersonic ramjets (scramjets), whose thrust computation is based on reliable physical models accounting for Mach number and flight altitude. Two datasets generated through computational optimization sampling of operating conditions using these models are considered; the datasets differ in the uniformity of data coverage across operating conditions. Holding the network architecture, loss function, and optimizer fixed, the DNN trained on the larger, non-uniform dataset achieved lower error but exhibited thrust trends inconsistent with the envelope defined by the physical model used in the optimization. In contrast, the DNN trained on the smaller, uniformly covered dataset yielded slightly higher error, but preserved coherent altitude-conditioned thrust trends across the Mach-altitude domain. These results underscore that adequate coverage is a methodological requirement for physically consistent DL models, indicating that reliable artificial intelligence-based tools in aerospace design depend on careful dataset construction rather than architectural complexity.

Keywords: Machine learning; Neural nets; Data sampling; Supersonic combustion ramjet engines; Thrust; Performance prediction.

INTRODUCTION

The aerospace industry is increasingly leveraging deep learning (DL) techniques to enhance the design and performance predictions of hypersonic air-breathing vehicles (HVs) (Ai *et al*. 2021; Cui *et al*. 2018; McCall *et al.* 2018; Paulino and Passaro 2023; Wang and Ma 2024). These vehicles – capable of traveling faster than Mach 5, that is, five times the speed of sound –pose unique challenges because of their complex aerodynamic behavior and harsh operating environments (Russo *et al.* 2021; Scarlatella *et al.* 2024; Speier *et al*. 2017). A depiction of an HV is shown in Fig. 1. As the demand for efficient and reliable hypersonic travel grows, accurate predictive models become critical. Deep learning (DL) models have shown considerable promise in this area (Mao *et al.* 2021; Paulino and Passaro 2024; Wang and Ma 2024). By integrating domain-specific knowledge into the modeling process, researchers can enhance both the precision and the reliability of DL-based predictions (Ren *et al.* 2023).

Received: Oct. 20, 2025 | **Accepted:** Apr. 02, 2026
Peer Review History: Single Blind Peer Review.
Section editor: José Atílio Fritz F Rocco

Source: Elaborated by the authors.

Figure 1. Artistic representation of an HV.

Despite the potential of DL in aerospace applications, a major challenge arises from the difficulty of obtaining representative high-fidelity training data with adequate coverage and sampling density across relevant design and operating spaces (e.g., Mach and flight-condition ranges) (Brunton *et al.* 2021; Le Clainche *et al.* 2023; Mao *et al.* 2021). Furthermore, understanding the underlying physics of hypersonic flight – particularly the fluid-dynamic interactions between vehicle geometry and atmospheric conditions – is essential for developing accurate predictive models (Anderson 2003). The datasets used to train DL models must capture these complex relationships as faithfully as possible. However, available datasets often fail to represent all critical regimes associated with the governing physics, resulting in non-uniform coverage (including coverage gaps) across the operational envelope, which can compromise model reliability and lead to biased predictions that deviate from physical reality (Brunton *et al.* 2021; Le Clainche *et al.* 2023; Mao *et al.* 2021). Many scientific publications, despite their technical rigor, neglect to specify how data distribution is managed (Chen *et al.* 2024). Moreover, data uncertainties may further aggravate the problem (Paulino *et al.* 2020). Models trained on such datasets may perform well under specific conditions but struggle to generalize across broader Mach-altitude regimes (Dong *et al.* 2024; Ling *et al.* 2016; Zhang *et al.* 2024).

Thus, addressing data coverage and sampling density across Mach-altitude operating-condition cells is essential for developing robust and physically consistent predictive models in aerospace engineering. The case study presented here investigates how non-uniform coverage across Mach-altitude operating-condition cells, together with uneven sampling density, affect the performance of deep neural networks (DNNs) – specifically, a feed-forward multilayer perceptron – used to predict the aerodynamic performance of different compression-section configurations of a supersonic combustion ramjet (scramjet), a propulsion system widely associated with HVs, over a defined Mach-altitude grid. Specifically, the adopted strategy generates samples through repeated, independent metaheuristic (MH)-based optimization runs. This approach leverages the effectiveness of MHs in exploring complex search spaces where exact optimal solutions are computationally impractical (Doğan and Ölmez 2015).

The present work formulates and tests the following hypothesis: when network architecture and training protocol are held constant, a training set derived from optimization-driven sampling may exhibit globally low error metrics despite non-uniform coverage across Mach-altitude operating-condition cells, while degrading the physical consistency of thrust predictions in underrepresented Mach-altitude regimes. To examine this hypothesis, this study combines two well-established components: a feed-forward DNN architecture previously validated for scramjet compression ramps (Paulino and Passaro 2024) and the sea turtle optimization (STO) algorithm for sample generation (Araújo *et al.* 2024).

Two datasets are constructed for comparison: Dataset 1 (236,640 samples), with non-uniform Mach-altitude coverage, uneven sampling density, and missing operating-condition cells; and Dataset 2, which maintains uniform coverage of a predefined Mach-altitude grid, with an equal number of samples per operating-condition cell, albeit with a lower number of 99,000 samples. Using identical network topology, loss function, and optimizer for both datasets, this study (i) compares conventional performance indicators such as root mean square error (RMSE) (Chai and Draxler 2014), (ii) evaluates physical plausibility through first-principles consistency checks based on altitude-conditioned thrust envelopes, and (iii) analyzes error distributions across the Mach-altitude envelope.

The main goal of this study is to demonstrate, in an application-driven context, the impact of uneven data coverage across Mach-altitude operating-condition cells on DNN predictions for a physically governed problem – specifically, scramjet thrust estimation. By isolating the distribution of input-output pairs across Mach-altitude operating-condition cells as the sole independent variable, while keeping network architecture and training protocol fixed, this work provides quantitative evidence of how non-uniform coverage and uneven sampling density can yield statistically accurate yet physically inconsistent predictions. The findings offer practical guidance for researchers employing DL-based surrogate models in hypersonic aerodynamics and other simulation-intensive, physics-governed domains.

Related work and theoretical framework

Training DNN models under uneven sampling

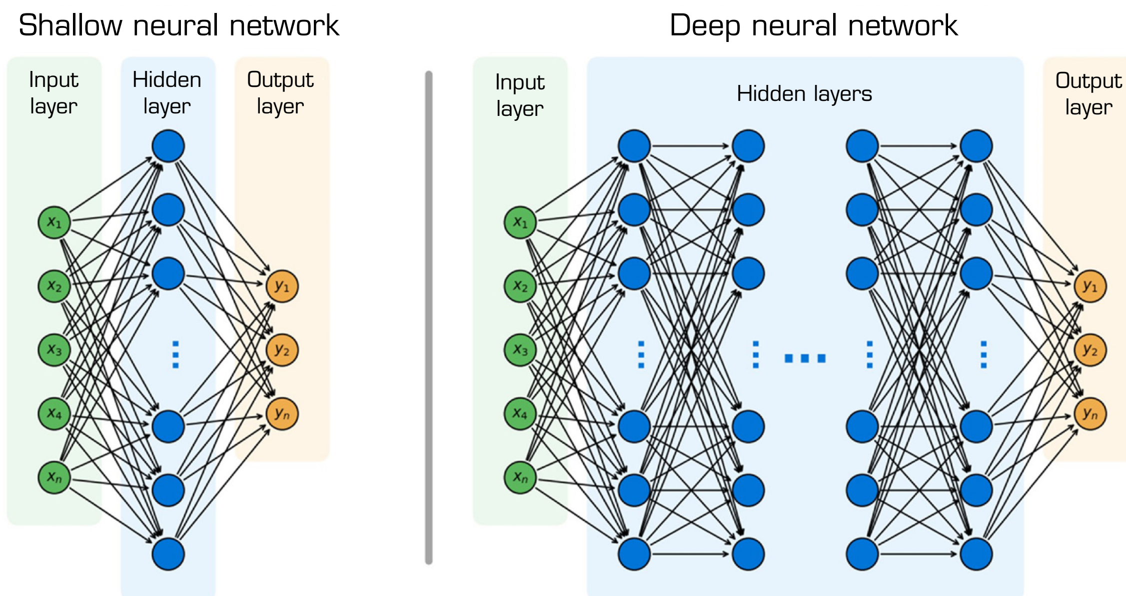
Machine learning (ML) provides a general computational framework for constructing artificial intelligence (AI) systems capable of operating in complex real-world environments (Goodfellow *et al.* 2016). Within ML, NNs constitute a central class of models inspired by biological information processing and widely used for approximating nonlinear relationships in high-dimensional problems (McCulloch and Pitts, 2021; Gond and Sengupta 2025; Paulino *et al.* 2019). The NNs are composed of interconnected artificial neurons organized in layers, where each neuron applies a nonlinear activation function to its inputs and propagates the result forward, enabling flexible function approximation (Haykin 2009; Schmidhuber 2015).

From an architectural perspective, NNs vary significantly in depth. Early network models typically employed one or a small number of hidden layers and are commonly referred to as shallow NNs. Although such architectures are theoretically capable of approximating arbitrary continuous functions, their practical effectiveness is often constrained by representational inefficiency, as complex mappings may require an impractically large number of neurons per layer (Bengio 2009; Delalleau and Bengio 2011). These limitations motivated the development of deep architectures, in which multiple layers enable hierarchical representation learning and improved scalability.

The DL builds upon this evolution by employing models with multiple levels of nonlinear representation learning, commonly instantiated as DNNs with several hidden layers (Goodfellow *et al.* 2016). These architectures include multilayer perceptrons, convolutional and recurrent NNs, as well as more recent paradigms such as transformer-based models (Vaswani *et al.* 2017) and deep reinforcement learning frameworks (Mnih *et al.* 2015). This hierarchical organization has been shown to improve representational efficiency and predictive performance in complex, high-dimensional, and highly nonlinear tasks, which has driven the widespread adoption of DL across diverse application domains (Brunton *et al.* 2021; Goodfellow *et al.* 2016; Haykin 2009; LeCun *et al.* 2015; Mao *et al.* 2021; Schmidhuber 2015). A comparative depiction between shallow and deep NN architectures can be found in Fig. 2.

In aerospace engineering, particularly in hypersonics, DL has enabled advances in data-driven modeling. Applications range from flight control and trajectory optimization (Bu *et al.* 2023; Lv *et al.* 2023; Shi *et al.* 2022; Shou *et al.* 2023; Wang *et al.* 2022) to flowfield prediction and aerodynamic load estimation (Beachy *et al.* 2021; Fujio and Ogawa, 2022; Ozbenli *et al.* 2020). Paulino and Passaro (2024) demonstrated that DNNs trained on large datasets generated via multi-objective optimization can approximate scramjet aerodynamic performance, highlighting the potential of such models. However, DL also revealed fundamental limitations related to data availability and representativeness (Mao *et al.* 2021; Wang and Ma 2024).

The predictive performance of DL data-driven surrogate models, including DNN-based regressors, is fundamentally conditioned by how training samples are distributed over the input (parameter) space, since the training loss is effectively discretized through finite sample sets in both the physical and parametric domains (Wang *et al.* 2024). The problem addressed in this work is not



Source: Elaborated by the authors.

Figure 2. Comparison between shallow and deep NN architectures. Shallow NNs comprise one or a small number of layers between the input and output layers, whereas deep NNs stack multiple hidden layers, enabling hierarchical feature extraction and multi-level nonlinear representations.

class imbalance in a discrete-label setting, but uneven sampling density and non- uniform coverage in a continuous input space. Here, the operational input space is discretized into Mach-altitude operating-condition cells, and the distribution of samples across these cells determines the extent of coverage and sampling density.

When sampling density is uneven, some regions of the parameter space are densely characterized while others remain sparsely sampled or entirely unobserved, leading to a loss of coverage across operating-condition cells. Although related effects have been described in other contexts, such as temporally non-uniform sampling of time-series data (Han *et al.* 2025), the underlying mechanism – the incomplete encoding of system variability due to uneven coverage – extends naturally to spatially defined parameter spaces employed in surrogate modeling and DNN-based regression. A primary consequence of uneven coverage is the degradation of predictive reliability in regions that are weakly represented or absent from the training data. In surrogate modeling and design-of-experiments literature, clustered sampling is known to leave portions of the parameter space underexplored, resulting in regions where local approximation is poorly supported (Garud *et al.* 2017). In particular, from a statistical learning perspective, this corresponds to non-uniform coverage of the training distribution across Mach-altitude operating-condition cells. Predictions in such regions rely increasingly on extrapolation, which is associated with elevated error, uncertainty, and reduced robustness (Hu *et al.* 2016).

Beyond global effects, insufficient sampling density suppresses the ability of data-driven models to resolve localized, high-frequency, or rare features of the response surface. Similar limitations have been reported in diverse fields, including geochemical exploration (Hosseini-Dinani *et al.* 2019), non-uniformly sampled nuclear magnetic resonance spectroscopy (Pedersen *et al.* 2021), and probabilistic risk assessment of rare events (Choi and Song 2026). Conversely, excessive local sampling density introduces redundancy and computational inefficiency without commensurate gains in predictive performance, as over-sampling clustered regions inflates computational cost while contributing limited new information (Yuchi *et al.* 2023). Taken together, both insufficient and excessive local sampling impose intrinsic information constraints that cannot be fully mitigated by increased model complexity alone. In high-dimensional, nonlinear, and computationally intensive problems – such as scramjet thrust estimation – these considerations motivate the use of algorithmic solution-generation strategies that improve effective coverage while respecting physical and operational constraints. Such strategies include MH frameworks (Doering *et al.* 2019; Krishna *et al.* 2022; Talbi 2009).

Metaheuristics (MH)-based sampling

MHs are a widely adopted class of stochastic global optimization methods. These high-level, general-purpose iterative methodologies are designed to guide the search for good solutions in computationally hard optimization problems (Talbi 2009). They are commonly employed when exact methods are computationally prohibitive or fail to provide solutions within acceptable time frames (Doering *et al.* 2019; Drake *et al.* 2020). Although MHs do not guarantee convergence to a global optimum, they are widely used due to their ability to deliver near-optimal, high-quality solutions within reasonable computational times across a broad range of problem domains (Houssein *et al.* 2024; Talbi 2009). Unlike problem-specific heuristics, MHs are conceived as domain-independent frameworks, even though their practical implementation must be adapted to the specific problem and instance under consideration (Doering *et al.* 2019; Talbi 2009). Their performance depends on balancing exploration and exploitation, mitigating premature convergence to local optima, and effectively navigating the instance-specific structure of the optimization landscape (Doering *et al.* 2019; Houssein *et al.* 2024; Talbi 2009).

Metaheuristics can be broadly classified into population-based methods, which evolve a set of candidate solutions, and single-solution methods, which iteratively improve a single existing solution. An alternative classification distinguishes nature-inspired approaches, such as genetic algorithms and particle swarm optimization (PSO), from non-nature-inspired methods, such as Tabu search and simulated annealing (Doğan and Ölmez 2015; Du and Swamy 2016; Oliveira *et al.* 2020; Sörensen 2015). Additional taxonomies and classification schemes are discussed in Houssein *et al.* (2024).

In population-based MHs, a set of candidate solutions is maintained and evolved over successive generations, enabling the simultaneous exploration of multiple regions of the search space. PSO (Kennedy and Eberhart 1995), for example, guides a swarm of particles – each representing a candidate solution – based on their individual best positions and the global best position identified by the swarm, promoting information sharing and reducing the likelihood of premature convergence. PSO has been successfully applied to a wide range of optimization problems, including feature selection, portfolio optimization, and hybrid optimization frameworks (Almaiah *et al.* 2024; Doering *et al.* 2019; Talbi 2009).

Single-solution MHs, in contrast, emphasize intensification within a narrower region of the search space while still generating sequences of candidate solutions through systematic neighborhood exploration. The vortex search algorithm (Doğan and Ölmez 2015), for instance, starts from a broad exploratory phase and progressively contracts the search region toward more promising areas, producing samples increasingly concentrated around locally optimal solutions. The capability of some MHs to accept inferior solutions under specific conditions further enables transitions between attraction basins and enhances exploration (Talbi 2009).

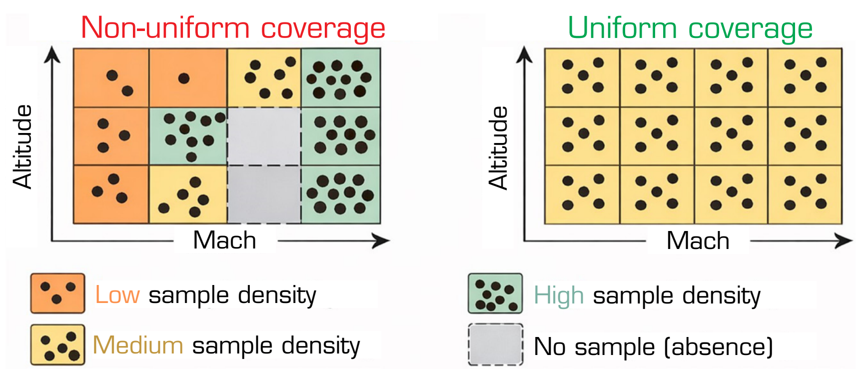
An application of MH-based optimization in hypersonic propulsion is presented by Araújo *et al.* (2024), who investigated the multi-objective optimization of HVs, specifically scramjet engines, using MH algorithms coupled with an analytical performance model validated against computational fluid dynamics (CFD) data. The optimization problem sought to maximize thrust while minimizing drag under strict aerodynamic and operational constraints. Unstart avoidance was enforced through a Gaussian penalization strategy based on the Korkegi limit (Korkegi 1975) in which candidate solutions were penalized whenever the shock-induced pressure-gradient criterion exceeded a semi-empirical threshold (Araújo *et al.* 2024). Three alternative formulations of the multi-objective function were evaluated. The results supported the reliability of the analytical model for preliminary scramjet design and indicated that the proposed optimization framework can reduce computationally intensive simulations during early-stage development.

Building upon this prior work, the present study adopts the same STO MH employed in Araújo *et al.* (2024). Here, STO is used as a solution-generation mechanism to construct two datasets through repeated optimization-driven exploration of the same analytical model, under identical feasibility constraints and penalization strategies. The resulting datasets intentionally include a broad range of outcomes, since the supervised learning task targets predictions over the full response surface rather than only feasible optima. The Mach-altitude operating-condition cells used for dataset construction, including the definition of the Mach and altitude grids, are detailed in the Methodology section.

The two datasets differ in how these Mach-altitude cells are populated during solution generation. Dataset 1 is constructed by enforcing non-uniformity across operating-condition cells, resulting in deliberately non-uniform coverage across the Mach-altitude grid. In contrast, Dataset 2 is explicitly constructed to ensure uniform coverage with equal sample counts per cell.

Although identical MH algorithms, physical modeling assumptions, and optimization formulations are employed, the resulting datasets correspond to independent realizations of the stochastic search process. As a result, both high- and low-performing designs may be included. It is important to note that this construction can only enforce uniformity across Mach–altitude operating-condition cells and does not imply uniformity across the full geometric input space.

A schematic comparison between non-uniform and uniform Mach-altitude coverage in the training distribution is presented in Fig. 3. The left panel illustrates non-uniform coverage, characterized by variable sampling density across operating-condition cells, including absent cells (gray dashed regions). The right panel shows uniform coverage, where all Mach-altitude operating-condition cells are represented and populated with equal sampling density. Black dots denote individual samples. Uniformity refers to the distribution of samples across Mach-altitude operating-condition cells rather than to the total number of samples. In this work, coverage is understood as a global property of the distribution across Mach-altitude operating-condition cells, encompassing both the presence or absence of cells and eventual sampling density differences among them, whereas sampling density denotes the number of samples within a given cell. Accordingly, the present work examines how training with non-uniform Mach-altitude coverage induced by optimization-driven solution generation affects the physical consistency and generalization behavior of DNN predictions of scramjet aerodynamic performance across the Mach-altitude envelope.



Source: Elaborated by the authors.

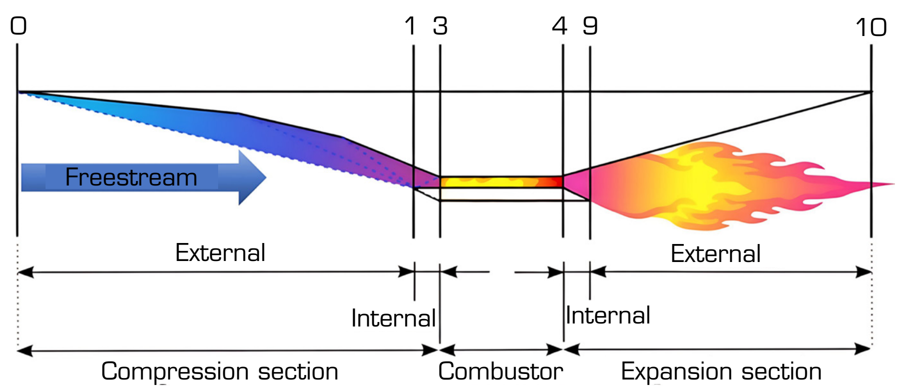
Figure 3. Schematic comparison of Mach-altitude coverage in the training distribution. Left: non-uniform coverage with heterogeneous sampling density and unpopulated cells (gray dashed regions). Right: uniform coverage with equal sample counts per cell. Black dots denote individual samples; uniformity refers to the distribution across Mach-altitude cells.

Supersonic combustion overview

The efficacy of designing HVs relies on the implementation of weight-saving, low-drag, and engine-integrated airframes (Takashima and Lewis 1999). These airframes should be such that the outer surfaces of the vehicle act as integral components of the propulsion system, which is the scramjet. The development must be addressed from a systems point of view, ensuring that not only performance goals are met, but also safety, security, maintainability, operational flexibility, reliability, and sustainability (Sziroczak and Smith 2016). Also, as a system, it must have real-time monitoring and control of its performance for it to be safe and practical for operation (Idris *et al.* 2019), requiring a high degree of system integration (Long and Hanzevack 1995; Takashima and Lewis 1999). It is worth mentioning that a cost-effective development of hypersonic vehicles demands analytical/numerical and experimental capabilities that can accurately predict hypersonic flowfields, solving the continuity equation, the momentum equation, the energy equation simultaneously (Bertin and Cummings 2003). As the need for accuracy in predicting aerothermodynamic performance increases, HV designs increasingly depend on computational methods (Eyi *et al.* 2019).

The accurate modeling of the airframe's aerodynamics and the combustor's flow conditions is imperative to predict a high-speed vehicle's performance, especially regarding the optimization of its range, payload, trajectory, and other characteristics (O'Brien *et al.* 2001). Thus, there is a necessity for design tools to provide rapid, accurate calculations of complex fluid flows (Long and Hanzevack 1995). In general, the flight envelope involves varying freestream conditions and altitude during ascent flight, which implies a complex optimization problem subject to several constraints, from design and operation perspectives (Brahmachary *et al.* 2022), regarding each part of the aircraft – especially the scramjet. In this work, the operational envelope is examined through a predefined discretization of Mach number and altitude, consistent with the Mach-altitude operating-condition cells adopted in the dataset construction and evaluation.

The scramjet, which uses atmospheric oxygen to burn the fuel in the combustion chamber, has great potential to reduce costs related to space access (Ogawa and Boyce 2012a), offering a higher specific impulse in flight conditions above Mach 5 (Anderson 2006). It is fundamentally divided into three main components: the inlet (external and internal compression sections), the combustor (combustion chamber), and the outlet (internal and external expansion sections) (Heiser *et al.* 1994). A depiction of each section of the scramjet is presented in Fig. 4.



Source: Elaborated by the authors.

Figure 4. Typical scramjet sections.

The inlet encompasses the stages from Station 0 (scramjet leading-edge) up to Station 3 (combustion chamber entrance), including the cowl leading-edge at Station 1. The inlet's primary function is to compress atmospheric air from the freestream, simultaneously reducing the airflow velocity and increasing its thermodynamic properties, aiming to provide adequate conditions for spontaneous combustion of the fuel (hydrogen). In hypersonic flight, the freestream hypersonic airflow is decelerated to supersonic speed at the entrance of the combustor (Station 3) through an adiabatic compression process. This process utilizes oblique shock waves (external compression, Stations 0-1) and reflected shock waves (internal compression, Stations 1-3) to achieve the necessary static temperature at Station 3 (T_3) and Mach number at Station 3 (M_3). For example, when burning hydrogen, the required T_3 of 1071.46 K for the incoming airflow is high enough to spontaneously ignite the fuel by heating it from its injection temperature (about 300 K) up to its auto-ignition temperature (845.15 K). Optimized inlet design relies on geometrical conditions such as "shock on-lip" and "shock on-corner" to ensure that all incident oblique shock waves converge to the cowl leading-edge and that the reflected shock wave converges to the combustion chamber entrance, maximizing the captured mass airflow (Araújo *et al.* 2021; Heiser *et al.* 1994).

The combustion chamber (Stations 3 to 4) is where the supersonic combustion takes place. Due to the high-speed flight, the airflow velocity must remain supersonic within the combustor. T_3 and M_3, established by the inlet, are considered the most important conditions for scramjet design. After the fuel burns with the supersonic airflow, the high-energy combustion products flow into the expansion section (Stations 4 to 10), which includes the cowl trailing-edge at Station 9 and the scramjet trailing-edge at Station 10. The role of the expansion section is to expand these products to generate the necessary thrust. Obtaining a positive net thrust (F_{net}) is essential for accelerating the scramjet to higher hypersonic velocities.

To prevent detrimental operational issues specific to high-speed flight, such as flow blockage, the Mach number of the combustion products at the combustor exit (M_4) must remain supersonic (e.g., $M_4 \geq 1.1$) to avoid thermal choking. Furthermore, managing the shock boundary layer interactions, often assessed using criteria such as the Korkegi limit (Korkegi 1975), is critical in the combustor area to prevent flow separation and "unstart" conditions, especially for flight Mach numbers lower than about 7 when burning hydrogen (Araújo *et al.* 2021; Heiser *et al.* 1994).

This effect, combined with the rise in back pressure from fuel injection, limits the amount of fuel that can be added without causing adverse pressure gradients and causing inlet instability (Anderson 2006). Thus, to improve engine efficiency, drag must be minimized and thrust maximized by optimizing the engine configuration (Araújo *et al.* 2021; 2024; Geraci *et al.* 2019; Ogawa and Boyce 2012b; Smart 1999). F_{net} is a key parameter in scramjet optimization, as it accounts for the stream thrust and the aerodynamic forces acting on the engine/vehicle surfaces (drag). It is expressed as:

$$F_{net} = F_{un} - D_p - D_v \tag{1}$$

where F_{un}, D_p, and D_v denote the uninstalled thrust, the pressure drag, and the viscous drag, respectively. The uninstalled thrust is obtained from a one-dimensional momentum balance between the inlet and nozzle exit control surfaces, including the pressure forces acting on these surfaces (Heiser *et al.* 1994):

$$F_{un} = \dot{m}_a (V_e - V_\infty) + (p_e - p_\infty)\, A_e, \ \text{if} \frac{\dot{m}_f}{\dot{m}_a} \ll 1 \tag{2}$$

where $\dot{m}_a$ is the captured air mass flow rate, $\dot{m}_f$ is the fuel mass flow rate, and A_e is the nozzle exit area. The variables V and p denote the flow velocity and static pressure, respectively, with subscripts ∞ and e indicating freestream and nozzle exit conditions (Araújo *et al.* 2021; 2024). The condition $\dot{m}_f \cdot \dot{m}_a^{-1} \ll 1$ implies that the fuel mass flow rate is small compared with the air mass flow rate, allowing the momentum balance to be expressed in terms of $\dot{m}_a$ only.

In general, the captured air mass flow rate is defined by the surface integral of mass flux across the capture plane, as seen in Eq. 3:

$$\dot{m}_a = \int_{A_0} \rho \, (\vec{V} \cdot \vec{n}) \, dA \tag{3}$$

where A_0 is the effective inlet capture area in the freestream, ρ is the local density, $\vec{V}$ is the local velocity vector, and $\vec{n}$ is the outward unit normal to the surface. Under the assumptions of uniform freestream properties across A_0 and negligible spillage, Eq. 3 reduces to the one-dimensional expression below (Eq. 4):

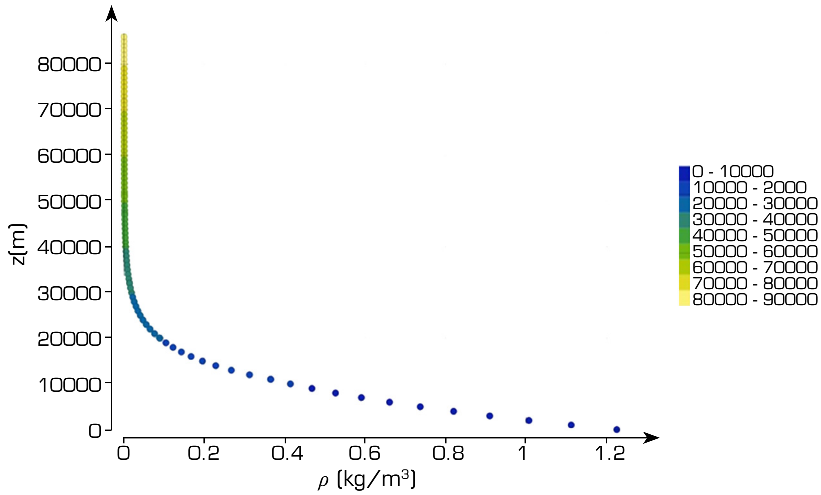
$$\dot{m}_a = \rho_\infty V_\infty A_0 \tag{4}$$

where ρ_∞ and V_∞ denote the freestream density and velocity, respectively.

Based on the U.S. Standard Atmosphere (NASA 1976), ρ_∞ decreases approximately exponentially with altitude (Fig. 5), directly influencing the uninstalled thrust F_{un} through its effect on $\dot{m}_a$ (Eq. 4).

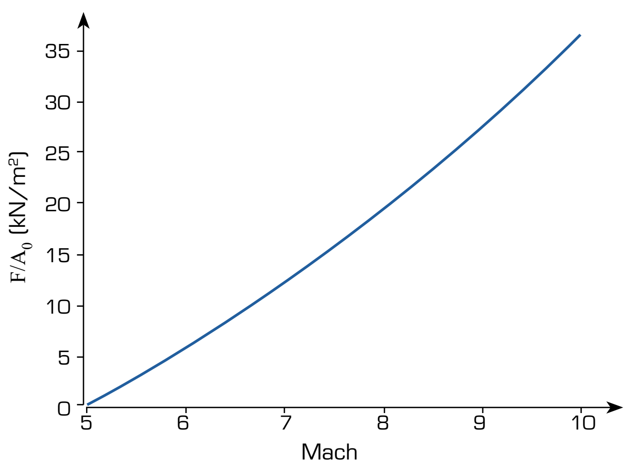
At a fixed altitude, the stream thrust increases nearly exponentially with Mach number (Araújo *et al.* 2021). Thus, an increase in Mach number or a reduction in altitude is expected to enhance the net thrust parameter, typically exhibiting an exponential-like behavior. As illustrated in Fig. 6, the ideal scramjet thrust per unit air capture area at an altitude of 30 km exhibits an exponential-like growth with increasing flight Mach number. While the thrust remains modest at lower hypersonic speeds, it rises sharply beyond Mach 7 and approaches values exceeding 35 kN·m^{-2} near Mach 10 for a five-ramp inlet configuration. This trend emphasizes the strong nonlinear dependence of scramjet performance on flight Mach number.

These physical relationships form the foundation of the consistency checks applied to the learned maps in this work. Across the Mach-altitude envelope, a physically plausible surrogate must reproduce two main trends: (i) a non-decreasing F_{net} with



Source: Elaborated by the authors.

Figure 5. Variation of the air density with altitude – U.S. Standard Atmosphere (NASA 1976).



Source: Adapted from Araújo *et al.* (2021).

Figure 6. Exponential-like increase of the ideal scramjet thrust per unit air capture area with flight Mach number at 30 km altitude, reaching values above 35 kN·m^{-2} near Mach 10 for a vehicle inlet with five ramps.

increasing M_0 at fixed altitude, and (ii) an increasing F_{net} with rising ρ_∞ (i.e., decreasing altitude) at fixed M_0, typically exhibiting an exponential-like dependence. Deviations from these trends – such as inversions of the altitude effect or non-monotonic Mach curves far from operability boundaries – indicate that a given model violates the underlying physics. Such inconsistencies can arise when the training data exhibit non-uniform coverage and uneven sampling density across Mach-altitude operating- condition cells, constraining the model's ability to learn physically consistent mappings.

METHODOLOGY

To systematically assess the impact of uneven sampling on the predictive performance of DNNs, both quantitative and qualitative analyses were performed using two different datasets generated through optimization-driven sampling. The first was generated with enforced non-uniformity across Mach-altitude operating-condition cells (Dataset 1), while the second was explicitly constructed to ensure uniformity (Dataset 2). The same physical modeling assumptions and optimization framework were applied when generating the datasets. However, they differ in the coverage and sampling density of the input-output pairs presented to the DNN, as a consequence of how operating conditions were imposed during the solution-generation process. As explained above, in this work, sampling density refers to the number of samples available within an observed Mach-altitude operating-condition cell, and non-uniform coverage refers to the uneven representation of Mach-altitude cells in the dataset, including the eventual absence of specific cells and differences in sampling density between cells. The two datasets differ not only in their coverage of the Mach-altitude grid, but also in the specific geometric configurations they contain. Thus, Dataset 2 cannot be constructed as a subsample of Dataset 1, because all configurations are generated stochastically within each prescribed cell. The following sections describe the construction of the datasets. Quantitative summaries and visual analyses are included to support the discussion.

Datasets

The datasets analyzed in this study are derived from the optimization framework described in Araújo *et al.* (2024). The framework generates candidate scramjet compression-section configurations by relating F_{net} to a set of input parameters representing both operational flight conditions and geometric characteristics. The input variables, detailed in Table 1, include geometric descriptors (such as the height of the combustion chamber and ramp shock intensities) and freestream conditions (temperature, pressure, and Mach number). Thus, each sample corresponds to one scramjet configuration evaluated under a specific combination of operating conditions, as defined by the imposed constraints and analytical performance model. It is important to emphasize that the datasets intentionally contain both high- and low-performing configurations, including cases with negative F_{net} and/or blockage-risk indicators (Korkegi < 1), since the supervised learning task (in particular, the DNN training) aims at approximating the full response surface rather than only feasible optima.

Table 1. Description of input parameters.

Feature	Description	Role
h_3	Combustion chamber height	Input
y_0	1st ramp shock intensity	Input
y_1	2nd ramp shock intensity	Input
y_2	3rd ramp shock intensity	Input
T_0	Freestream temperature	Input
P_0	Freestream pressure	Input
M_0	Freestream Mach number	Input
$^*F_{net}$	*Scramjet's net thrust	Output

Source: Elaborated by the authors. *Target feature.

Dataset 1 covers a wide range of flight conditions, with Mach numbers ranging from 6 to 10 and altitudes from 25 to 35 km. The relationship between freestream pressure (p_0) and temperature (T_0) results from their dependence on flight altitude (Z). However, Z is not directly included as an input parameter, unlike M_0. In practice, is imposed during dataset generation, and its effect is encoded in the input variables (p_0, T_0) according to (NASA 1976). Table 2 summarizes the distribution of samples per Mach-altitude operating condition in Dataset 1.

Dataset 2 extends the altitude interval from 25-35 km (Dataset 1) to 20-40 km (Dataset 2). A systematic Mach-altitude grid was defined with five discrete Mach values ($M_0 \in \{6, 7, 8, 9, 10\}$) and five discrete altitude values ($Z \in \{20, 25, 30, 35, 40\}$ km), resulting in 25 operating-condition cells. For instance ($M_0 = 7$, $Z = 30$) is one cell. For Dataset 2, sampling was performed to ensure complete coverage of the Mach-altitude grid, with an equal number of samples per cell, as shown in Table 3, while Dataset 1 misses a number of cells (with a corresponding sampling density of zero). Both datasets serve as the basis for assessing the influence of coverage and sampling density on the training and predictive performance of the DNN models.

The differences in the sample distributions between Dataset 1 and Dataset 2 across Mach-altitude operating-condition cells are visually illustrated in Figs. 7–10. Fig. 7 shows the histogram of samples' Mach numbers in Dataset 1, highlighting disparities in

Table 2. Number of samples per Mach-altitude operating-condition cell – Dataset 1.

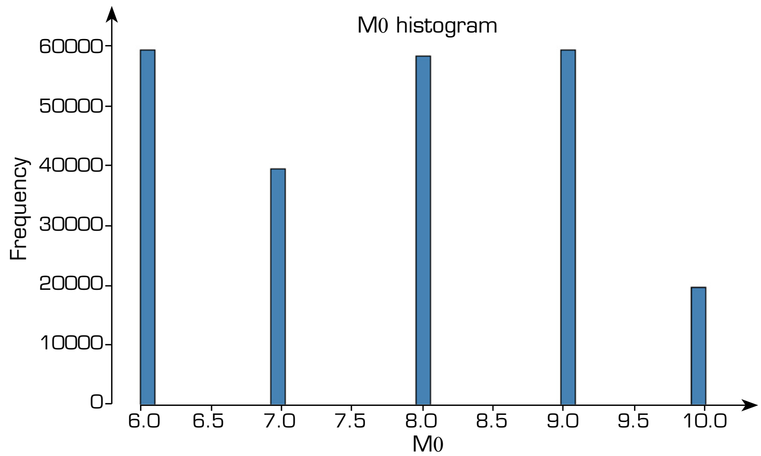
Mach – M_0	Altitude – Z (km)	Samples	Samples/Mach	Total samples
6	25	19,800		
6	30	19,800	59,400	
6	35	19,800		
7	25	19,800		
7	30	19,800	39,600	
8	25	19,800		
8	30	19,800	58,440	236,640
8	35	18,840		
9	25	19,800		
9	30	19,800	59,400	
9	35	19,800		
10	30	19,800	19,800	

Source: Elaborated by the authors.

Table 3. Number of samples per Mach-altitude operating-condition cell – Dataset 2.

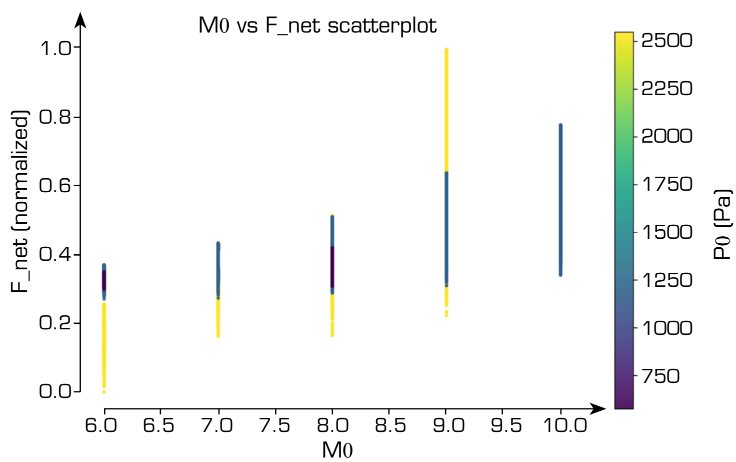
Mach – M_0	Altitude – Z (km)	Samples	Samples/Mach	Total samples
6	20	3,960		
6	25	3,960		
6	30	3,960	19,800	
6	35	3,960		
6	40	3,960		
7	20	3,960		
7	25	3,960		
7	30	3,960	19,800	
7	35	3,960		
7	40	3,960		
8	20	3,960		
8	25	3,960		
8	30	3,960	19,800	99,000
8	35	3,960		
8	40	3,960		
9	20	3,960		
9	25	3,960		
9	30	3,960	19,800	
9	35	3,960		
9	40	3,960		
10	20	3,960		
10	25	3,960		
10	30	3,960	19,800	
10	35	3,960		
10	40	3,960		

Source: Elaborated by the authors.



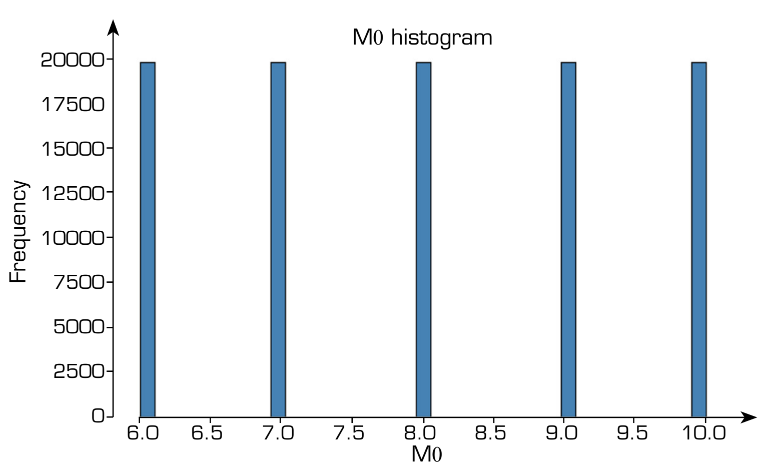
Source: Elaborated by the authors.

Figure 7. Histogram of M_0 values for Dataset 1, showing uneven sampling coverage across Mach numbers.



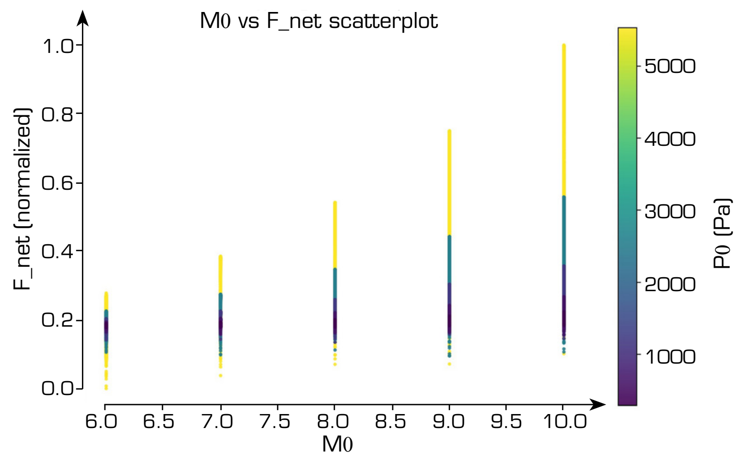
Source: Elaborated by the authors.

Figure 8. Scatterplot of the normalized values of F_{net} as a function of M_0 for Dataset 1, evidencing a non-uniform behavior. Each point indicates a sample across Mach-altitude operating-condition cells. The color-coded scale shows atmospheric pressure (Pa).



Source: Elaborated by the authors.

Figure 9. Histogram of M_0 values for Dataset 2, indicating uniform sampling density across Mach numbers.



Source: Elaborated by the authors.

Figure 10. Scatterplot of the normalized values of F_{net} as a function of M_0 for Dataset 2. Each point indicates a sample across Mach-altitude operating-condition cells. The color-coded scale shows atmospheric pressure (Pa).

sample frequency across the underlying Mach-altitude cells. Fig. 8 presents a scatterplot of samples' normalized F_{net} as a function of M_0, color-coded by p_0, with a non-uniform behavior. For Dataset 2, Fig. 9 shows the histogram of samples' Mach numbers with uniform bar heights, and Fig. 10 provides a scatterplot of samples' normalized F_{net} as a function of Mach number, also color-coded by p_0. It should be noted that p_0 decreases monotonically with Z over the altitude interval considered, as shown in previous section.

Characterization of sampling coverage and datasets construction

Analyzing the previous subsection, it is clear that Dataset 1 reveals clear asymmetries across Mach number and altitude distributions, manifested as uneven sampling density and non-uniform coverage in the Mach-altitude parameter space (if necessary, refer to Fig. 3 for clarification). Mach numbers 6, 8, and 9 are each associated with three altitude conditions. In contrast, Mach 7 and 10 are represented by only two and one altitude condition, respectively. The sample counts per condition also vary, resulting in aggregate disparities – such as 59,400 samples for Mach 6 and 9, compared to only 19,800 for Mach 10 – yielding a 3:1 ratio between the most and least sampled regimes. These disparities, summarized in Table 2, are visually confirmed by the histogram in Fig. 7, which also shows that Mach 6, 8, and 9 each contain approximately 59,000 samples, while Mach 7 and 10 are substantially underrepresented. Several Mach–altitude cells are absent, including combinations at Z = 35 km for Mach 7 and combinations for Mach 10 outside Z = 30 km. In addition, Dataset 1 does not include operating conditions at 20 km and 40 km, which are explicitly included in Dataset 2.

Such factors constrain the information available to train the DNN using Dataset 1. The implications for the learning process can be stated as follows: systematic biases may arise as a consequence of uneven sampling density and non-uniform coverage, such that models trained on Dataset 1 become disproportionately influenced by overrepresented regimes. This effect can reduce predictive reliability in sparsely sampled conditions. Accordingly, the uncertainty of performance estimates can vary across the Mach regimes represented in the data. Predictions for the least represented conditions (e.g., Mach 7 and 10) are more error-prone.

To provide a controlled contrast with Dataset 1, Dataset 2 was designed to achieve uniform coverage across Mach number and altitude (i.e., across operating-condition cells). The sampling procedure involved the use of the previously defined Mach-altitude grid described in the previous subsection (25 operating-condition cells). Each combination was populated with 3,960 samples. This yields a total of 99,000 data points and ensures an equitable distribution of 19,800 samples per Mach number (Table 3).

For both datasets, samples were generated independently using the STO MH implemented in the LOF-SYSTEM optimization framework (Saba 2017). This framework, which has been applied in several academic studies (Araújo *et al.* 2024; De Lima Filho *et al.* 2022; Santana *et al.* 2022; Silva *et al.* 2021; Soares *et al.* 2022), provides a set of established single-solution and population-based MH algorithms. Sampling was performed within each Mach-altitude operating-condition cell, under the same physical modeling assumptions and constrained multi-objective formulation described in (Araújo *et al.* 2024). All cells in the prescribed grid were populated with a predefined number of samples, ensuring uniform coverage and consistent sampling density across the discrete Mach-altitude grid for the operating conditions considered.

The uniform coverage of Dataset 2 is shown in Fig. 9, where all Mach numbers are equally represented in the histogram. Figure 10 complements this view by showing the distribution of normalized F_{net} across Mach number, also color-coded by p_0. The observed trends indicate that the normalized F_{net} increases with Mach number and also increases as altitude decreases, consistent with the compressible-flow behavior embedded in the analytical performance model over the operating conditions considered. At lower altitudes, higher air density and static pressure lead to greater air mass flow rates through the inlet, enhancing overall thrust generation. Figure 10 further reveals a strong nonlinear dependence of F_{net} on static pressure, particularly evident in the highest-pressure data points.

The analysis above establishes the coverage properties used in subsequent DNN training experiments. The uniform distribution of samples across all Mach-altitude operating-condition cells reduces potential weighting effects across operating conditions in the evaluation of DNN generalization. It is important to note that, while Dataset 2 enforces uniform coverage across Mach-altitude operating-condition cells by construction, it does not imply uniformity across the full geometric input space. By characterizing both datasets and the methodology used to construct a dataset with uniform coverage across Mach-altitude operating-condition cells, this section establishes a comprehensive framework for assessing how coverage and sampling-density properties influence DL-based scramjet performance modeling.

Deep neural network (DNN) training

This study adopts, without modification, the optimal DNN architecture proposed by Paulino and Passaro (2024). In that work, the architecture was determined through a systematic process that began with simple models available in the PyTorch library (Paszke *et al.* 2019). The complexity was then progressively increased by adding hidden layers and tuning the number of neurons, until no further significant performance improvements were observed.

Here, the training procedure used the mean squared error (MSE) loss function (Goodfellow *et al.* 2016), Eq. 5. As for model performance, it was evaluated using the RMSE (Chai and Draxler 2014), Eq. 6, and the coefficient of determination (R^2), Eq. 7. Both metrics are widely used in regression tasks (Ispir *et al.* 2023; Staerk *et al.* 2024). The equations are:

$$\mathrm{MSE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(\mathrm{y}_{\mathrm{pred,i}} - \mathrm{y}_{\mathrm{ref,i}}\right)^2 \tag{5}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(\mathrm{y}_{\mathrm{pred,i}} - \mathrm{y}_{\mathrm{ref,i}}\right)^2} \tag{6}$$

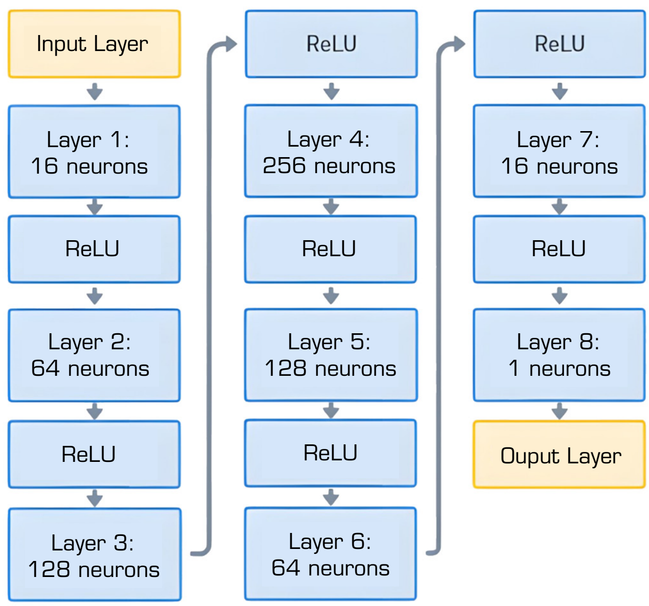
$$\mathrm{R}^2 = 1 - \frac{\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(\mathrm{y}_{\mathrm{pred,i}} - \mathrm{y}_{\mathrm{ref,i}}\right)^2}{\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(\mathrm{y}_{\mathrm{ref,i}} - \bar{\mathrm{y}}_{\mathrm{ref}}\right)^2} \tag{7}$$

where N is the number of samples, $\mathrm{y}_{\mathrm{pred,i}}$ is the value predicted by the DNN for the i-th sample, $\mathrm{y}_{\mathrm{ref,i}}$ is the corresponding reference value provided by the analytical model used for dataset generation, and $\bar{\mathrm{y}}_{\mathrm{ref}}$ is the mean of all reference values. Higher R^2 values indicate that the model explains a larger proportion of the variance in the reference data (Brar and Singh 2024; Ispir *et al.* 2023).

In the present implementation, model parameters were optimized using the AdamW optimizer (Loshchilov and Hutter 2019). Training was performed on a graphics processor unit (GPU). The rectified linear unit activation function was employed, as recommended in the literature (Goodfellow *et al.* 2016). A maximum of 20,000 epochs was allowed, with early stopping applied if a minimum improvement threshold of 10^{-4} was not achieved within 200 epochs. The batch size was set to 500. A grid search (Bergstra *et al.* 2012) over the discrete set {0.0002, 0.0005, 0.001, 0.002} was applied to tune the learning rate (*lr*), selected for its simplicity and suitability for exploring a small, manually defined hyperparameter space. Training was terminated if R^2 exceeded 0.995. Literature reports that low loss values and high R^2 (typically above 90-95%) are generally regarded as indicative of good regression performance (Fujio and Ogawa 2022; Ispir *et al.* 2023; Kontolati *et al.* 2022; Mao *et al.* 2021; Thuerey *et al.* 2020).

Architecture and mapping

In the present study, the chosen DNN architecture is shown in Fig. 11. The first layer (input) receives a matrix in which the number of rows corresponds to the number of samples presented to the network, and the number of columns matches the selected input parameters, which also define the number of input neurons. The last layer (output) produces a scalar prediction of the target variable F_{net}.



Source: Elaborated by the authors.

Figure 11. Architecture of the DNNs used for the experiments.

Accordingly, the DNN defines a nonlinear regression mapping from the seven-dimensional input feature space to the predicted scalar net thrust output. This mapping can be written as Eq. 8:

$$\hat{F}_{net} = f_\theta(h_3, y_0, y_1, y_2, T_0, p_0, M_0) \tag{8}$$

where $\hat{F}_{net}$ is the predicted F_{net} value and $f_\theta(\cdot)$ denotes the DNN mapping parameterized by weights and biases θ, and the input variables are defined in Table 1. For notational simplicity, the geometric descriptors can be grouped into a geometry feature vector, defined as $x = (h_3, y_0, y_1, y_2)$. Additionally, the freestream quantities T_0 and p_0 are determined by altitude Z according to the standard-atmosphere relations (NASA 1976). Thus, the DNN mapping of Eq. 8 can be equivalently expressed as Eq. 9:

$$\hat{F}_{net} = f_\theta(x, M_0, Z) \tag{9}$$

In the original optimization framework used for dataset generation (Araújo *et al.* 2024), one goal is to find, for fixed flight conditions (M_0, Z), a geometry vector x^* that maximizes the net thrust predicted by the analytical performance model. This relationship can be written as:

$$x^* = (h_3^*, y_0^*, y_1^*, y_2^*) = \arg\max_{x} F_{net}^{analytical}(x, M_0, Z) \quad (10)$$

where $F_{net}^{analytical}$ denotes the net thrust computed by the analytical model employed in the optimization framework. The components h_3^*, y_0^*, y_1^* and y_2^* represent the optimal geometric parameters obtained under the imposed feasibility constraints for the specified flight condition.

Computational implementation

The development environment for this study was based on Python, leveraging the PyTorch library (Paszke *et al.* 2019) to construct, train, and evaluate the DNN models. PyTorch was selected for its intuitive debugging, computational efficiency, and native support for GPU acceleration. Additional libraries were used to support various tasks: NumPy (Harris *et al.* 2020) and Pandas (McKinney 2010) for data manipulation and numerical operations; Scikit-learn (Pedregosa *et al.* 2011) for statistical tools and ML utilities; and Matplotlib (Hunter 2007) and Plotly Inc. (2015) for data visualization. A complete Python-based pipeline was developed to preprocess the datasets, define and train NN architectures, and carry out performance evaluation under various flight conditions. It is important to note that, as previously described, the synthetic samples used to construct the datasets were generated separately using the STO MH implemented in the LOF-SYSTEM software (Saba 2017), outside the Python environment.

Experiments

Following the described computational implementation, the performance of DNNs in predicting F_{net} was evaluated using a simplified version of the seven-fold cross-validation (7FCV) strategy (Paulino *et al.* 2019; 2020). Instead of considering all 42 possible combinations of training (5 folds), validation (1 fold), and generalization (1 fold) sets among the seven partitions, this study adopts a simplified approach, hence called "Simplified 7FCV" (S7FCV). Before partitioning, all samples were randomly shuffled to ensure statistical independence across folds. One of the seven folds is fixed as the generalization set for all cross-validation iterations. The remaining six folds are used to generate all combinations of five folds for training and one for validation, resulting in six iterations ($C_5^6 \cdot C_1^1 = 6$). This configuration allows for model selection and evaluation under a consistent generalization scenario while reducing computational cost compared to the full combinatorial approach. A depiction of the S7FCV scheme is shown in Table 4.

Table 4. Cross-validation folds and data partitioning.

Iteration	Fold A	Fold B	Fold C	Fold D	Fold E	Fold F	Fold G
1	Validation	Training	Training	Training	Training	Training	Generalization
2	Training	Validation	Training	Training	Training	Training	Generalization
3	Training	Training	Validation	Training	Training	Training	Generalization
4	Training	Training	Training	Validation	Training	Training	Generalization
5	Training	Training	Training	Training	Validation	Training	Generalization
6	Training	Training	Training	Training	Training	Validation	Generalization

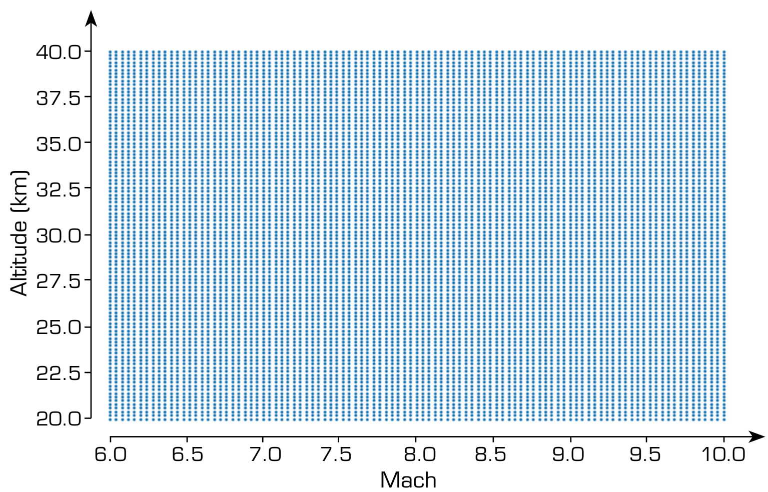
Source: Elaborated by the authors.

All input features and the output $\hat{F}_{net}$ were normalized to the range [0,1] using min-max scaling with the same normalization parameters applied consistently across all models and evaluation scenarios. For Dataset 1, each validation fold contained 33,799 samples and each training set contained 168,995 samples (5 folds), with any remaining samples from the non-divisible total distributed by the random partitioning procedure. Both validation and generalization folds contained 33,799 samples. For Dataset 2, each fold contained 14,143 samples, yielding 70,715 training samples and 14,143 validation samples per iteration, with another 14,143 for generalization.

During training, each combination in the S7FCV procedure generated a distinct DNN instance, characterized by unique weight and bias configurations. Moreover, each data split in the S7FCV was tested with four different values of the "*lr*" hyperparameter, further expanding the total number of trained models. Among the ensemble of networks trained on Dataset 1, the model yielding the lowest RMSE on the fixed generalization fold of the S7FCV protocol was designated Model 1. Similarly, among the ensemble trained on Dataset 2, the model yielding the lowest RMSE under the same evaluation protocol was designated Model 2. Thus, each selected model corresponds to the configuration that achieved the highest overall performance within its respective training dataset under the adopted protocol. Both models are subsequently evaluated under identical flight-condition scenarios to ensure comparability.

Dense-grid evaluation

After model selection, the resulting trained networks are additionally evaluated over a dense grid of freestream operating conditions comprising 101 Mach values ($M_0 \in [6, 10]$) and 101 altitude values ($Z \in [20, 40]$) (totaling 10,201 combinations). In this dense-grid evaluation, the geometric input variables are held fixed to a single reference geometry vector, denoted by $\mathbf{x}_{ref}$, while the freestream operating conditions are varied over (M_0, Z), with Z encoded through the corresponding (T_0, p_0) pairs, enabling consistent Mach-altitude trend comparisons between the trained models. The dense grid of evaluation combination points is depicted in Fig. 12.



Source: Elaborated by the authors.

Figure 12. Dense evaluation grid consisting of 10,201 combinations (points) of freestream operating conditions (101 × 101 Mach and altitude values) used for model trend comparisons.

Physics-based thrust envelope and physics-consistency metric

The physics-based thrust envelope adopted in this work is grounded on the compressible-flow trends embedded in the analytical scramjet performance model employed for dataset generation and optimization, as described in Araújo *et al.* (2024). In that framework, the net thrust F_{net} of a scramjet operating at a fixed altitude and Mach number is obtained from a consistent set of analytical relations that account for inlet compression, combustion, nozzle expansion, and drag contributions, under the feasibility constraints imposed during the optimization process.

For a fixed altitude Z, variations in M_0 affect the freestream conditions (e.g., density and dynamic pressure), the captured mass flow rate, and the inlet compression strength Araújo *et al.* (2021). These compressible-flow trends yield thrust responses that are monotonic in M_0 over the operational range considered in this study. Based on this expected behavior, a physics-based thrust envelope $F_{\text{env}}^{\text{phys}}(M_0, Z)$ was defined for each altitude Z as a smooth reference curve reflecting the monotonic thrust response observed in the analytical model. Consistent with these observations, the envelope satisfies Eq. 11:

$$\frac{dF_{\text{env}}^{\text{phys}}(M_0, Z)}{dM_0} \geq 0, \quad \forall M_0 \in [M_{0,min}, M_{0,max}] \tag{11}$$

where $[M_{0,min}, M_{0,max}]$ denotes the Mach interval considered in the analysis. In practice, $F_{\mathrm{env}}^{\mathrm{phys}}(M_0, Z)$ is represented by a spline curve constructed from thousands of direct evaluations of the analytical F_{net} model at fixed altitude, which consistently exhibit a monotonic increase of thrust with M_0 under the imposed physical feasibility constraints. For consistency with the DNN outputs, the envelope is expressed in the same normalized scale adopted in the evaluation scenario. Additionally, it is important to emphasize that $F_{\mathrm{env}}^{\mathrm{phys}}(M_0, Z)$ is not intended to represent an exact analytical solution for a specific geometry, nor a CFD-derived reference value, although the model was validated with CFD simulations in Araújo *et al.* (2024). Instead, it provides a physically consistent baseline, derived from the analytical model under the imposed feasibility constraints, against which deviations in the DNN-predicted thrust trends can be assessed.

Beyond conventional statistical indicators (RMSE and R^2), the physics-consistency evaluation was performed by comparing $\hat{\mathrm{F}}_{\mathrm{net}}$ trends against $F_{\mathrm{env}}^{\mathrm{phys}}(M_0, Z)$. In this context, the analytical model constitutes the reference representation of physically admissible thrust behavior under the imposed compressible-flow assumptions and feasibility constraints, whereas the DNN predictions are data-driven approximations learned from sampled configurations.

Although the DNN is trained using the full input vector in Table 1, including both freestream variables and geometric descriptors, the physics-consistency analyses in the Results and discussions section evaluate the trained models under a fixed reference geometry, $\mathbf{x}_{\mathrm{ref}}$. Specifically, a single optimized geometry vector obtained from the optimization framework described in Araújo *et al.* (2024) was selected and held constant, while the freestream operating conditions were varied over the dense Mach-altitude grid defined in the Dense-grid evaluation subsection. The optimized geometry was defined as $\mathbf{x}^* = (h_3^*, y_0^*, y_1^*, y_2^*)$, with the "*" denoting the optimality for each parameter. Thus, for each operating condition (M_0, Z), the trained DNNs are queried for this fixed reference scramjet geometry, denoted by $\mathbf{x}_{\mathrm{ref}} = \mathbf{x}^* = (h_3^*, y_0^*, y_1^*, y_2^*)$, and produce a single normalized $\hat{\mathrm{F}}_{\mathrm{net}}$ value, using Eq. 9. The altitude-conditioned predicted thrust response is thus defined as Eq. 12:

$$F_{\mathrm{ref}}^{\mathrm{pred}}(M_0, Z) = F_{\mathrm{net}}^{\mathrm{pred}}(M_0, Z|\mathbf{x}_{\mathrm{ref}}) \tag{12}$$

where $\mathbf{x}_{\mathrm{ref}}$ denotes the fixed reference geometry adopted in the evaluations, and $F_{\mathrm{net}}^{\mathrm{pred}}(M_0, Z|\boldsymbol{x}_{\mathrm{ref}})$ is the normalized $\hat{\mathrm{F}}_{\mathrm{net}}$ under the freestream operating condition (M_0, Z) for that specific geometry. In this formulation, no optimization over geometric configurations is performed, because the geometry is held constant and only (M_0, Z) varies. For notational clarity, when a fixed altitude Z is considered, $F_{\mathrm{env}}^{\mathrm{phys}}(M_0, Z)$ is written as $F_{\mathrm{ref}}^{\mathrm{pred}}(M_0)$, and $F_{\mathrm{env}}^{\mathrm{phys}}(M_0, Z)$ is written as $F_{\mathrm{env}}^{\mathrm{phys}}(M_0)$. Thus, for a given fixed altitude, the discrepancy between the DNN-predicted response and the physics-based envelope was defined as Eq. 13:

$$\Delta F(M_0) = F_{\mathrm{ref}}^{\mathrm{pred}}(M_0) - F_{\mathrm{env}}^{\mathrm{phys}}(M_0) \tag{13}$$

which quantifies the signed deviation between the DNN-predicted normalized thrust response for the fixed reference geometry and the physics-based thrust envelope $F_{\mathrm{ref}}^{\mathrm{pred}}(M_0)$ at each Mach number. Thus, nonzero values of $\Delta F(M_0)$ quantify departures from the baseline monotonic trend represented by $F_{\mathrm{env}}^{\mathrm{phys}}(M_0)$. Because the comparison is performed at a fixed altitude, both $F_{\mathrm{ref}}^{\mathrm{pred}}(M_0)$ and $F_{\mathrm{env}}^{\mathrm{phys}}(M_0)$ are defined for the same Z and expressed in the same normalized scale. The overall magnitude of this deviation at a fixed altitude is summarized through the absolute envelope error (Eq. 14):

$$\mathcal{E}_{abs}(Z) = \int_{M_{0,min}}^{M_{0,max}} |\Delta F(M_0)| \; dM_0 \tag{14}$$

where $[M_{0,min}, M_{0,max}]$ denotes the Mach interval common to both envelopes at the fixed altitude Z. In practice, $\mathcal{E}_{abs}(Z)$ is evaluated numerically by quadrature of the spline representation of $\Delta F(M_0)$. By definition, $\mathcal{E}_{abs}(Z)$ accounts for deviations both above and below the physics-based reference envelope. Thus, lower values of $\mathcal{E}_{abs}(Z)$ indicate closer agreement between the DNN-predicted thrust trends and the baseline behavior implied by the analytical model under the imposed constraints.

Predictions comparison

This methodological framework was designed to investigate the impact of non-uniform coverage across operating conditions on the predictive behavior of DNN models for hypersonic aerodynamic performance prediction. To this end, two models with identical architecture, training protocol, and hyperparameter selection were trained under different data-coverage conditions: Model 1, trained using Dataset 1– generated through optimization-driven sampling with enforced non-uniform coverage across Mach-altitude operating-condition cells – and Model 2, trained using Dataset 2 – explicitly constructed to ensure uniform coverage over the Mach-altitude grid. This controlled setup allows the analysis to isolate how differences in data coverage propagate into differences in model generalization behavior and baseline consistency, rather than reflecting architectural or optimization effects. The results and comparative analyses of their predictive behaviors are presented and discussed next.

RESULTS AND DISCUSSIONS

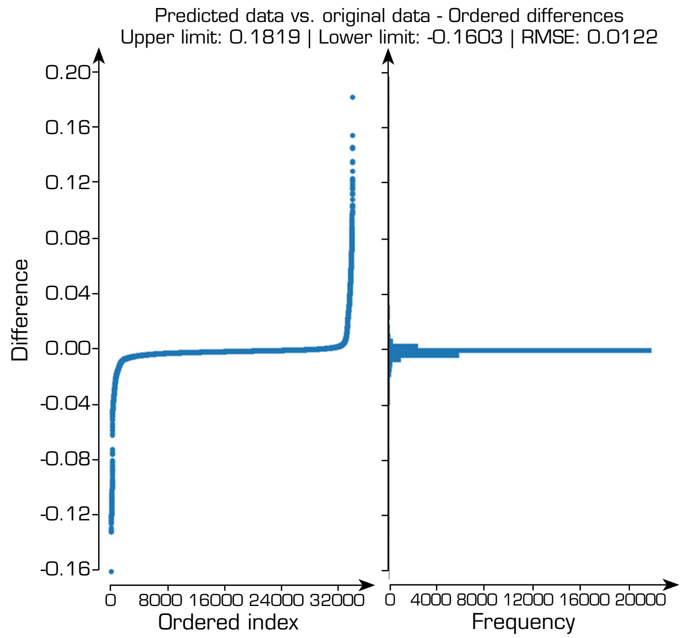
The results presented in this section contrast the predictive behavior of two DNN models – Model 1 and Model 2 – trained on datasets with distinct coverage and sampling-density properties across Mach-altitude operating-condition cells, while sharing the same architecture and training configuration. The analysis emphasizes two complementary dimensions: (i) statistical performance, quantified through cross-validation using the S7FCV scheme, and (ii) physical consistency, evaluated by examining trends across the Mach-altitude envelope using the physics-based metrics defined in the Methodology section. Together, these clarify how non-uniform coverage in the training distribution affects both numerical performance and baseline consistency of aerodynamic predictions. As shown in the following subsections, Dataset 1, which exhibits non-uniform Mach-altitude coverage, yields competitive global error metrics while producing thrust trends that depart from the baseline monotonic behavior implied by the analytical model. In contrast, Dataset 2, constructed to ensure uniform coverage across Mach-altitude operating-condition cells, preserves the expected trend behavior despite its smaller size.

Generalization performance

To assess generalization capability, the predictive performances of the selected models were compared using the statistical metrics defined in the Methodology section. Both models achieved high predictive performance, with R^2 (Eq. 7) exceeding 0.99. Model 1 exhibited a slightly lower RMSE (0.0122, according to Eq. 6) than Model 2 (0.0142), corresponding to an approximately 14% lower statistical error. This marginal advantage is expected given the larger training set size of Dataset 1, which contains approximately 2.4 times more samples than Dataset 2. The prediction errors' upper and lower bounds are also comparable: [–0.1603, 0.1819] for Model 1 and [–0.1760, 0.1882] for Model 2 (Figs. 13 and 14). In both cases, the distributions of ordered normalized differences exhibit a strong concentration near 0, indicating comparable aggregate predictive accuracy on the evaluated samples. Nevertheless, these indicators alone do not characterize whether the learned mapping preserves baseline trend behavior across operating conditions. As shown next, Model 1's predictions deviate from the expected monotonic thrust trends when examined across the Mach-altitude envelope. This highlights that a sheer lower numerical error achieved in training does not guarantee physical consistency, underscoring the importance of representative Mach-altitude coverage across operating-condition cells, beyond dataset size.

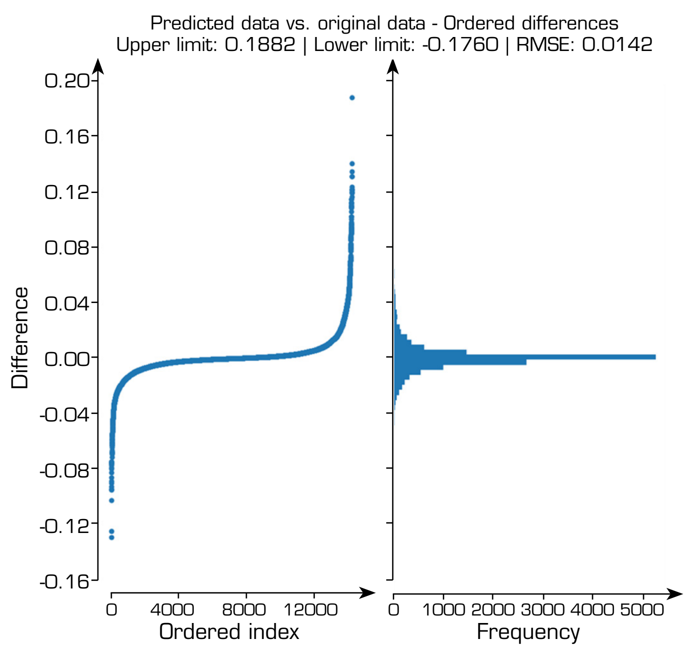
Evaluation of aerodynamic performance across Mach-altitude operating conditions

Figures 15 and 16 illustrate the values of normalized $\hat{F}_{net}$ across the Mach-altitude operating grid, as produced by Model 2 and Model 1, respectively. In these figures, each model is evaluated over 10,201 Mach-altitude combinations (see Dense-grid evaluation subsection). Consistent with the evaluation protocol defined in the Methodology section, the geometric inputs are held fixed to the reference geometry vector $\mathbf{x}_{ref}$, and only the freestream operating conditions vary over (M_0, Z). For each altitude value, the corresponding freestream inputs (T_0, p_0) were computed from the standard-atmosphere relations used in the dataset generation process, and then provided to the DNN together with the remaining geometric parameters (Eq. 9). Thus, these thrust maps reflect the learned DNN responses over the defined Mach-altitude envelope.



Source: Elaborated by the authors.

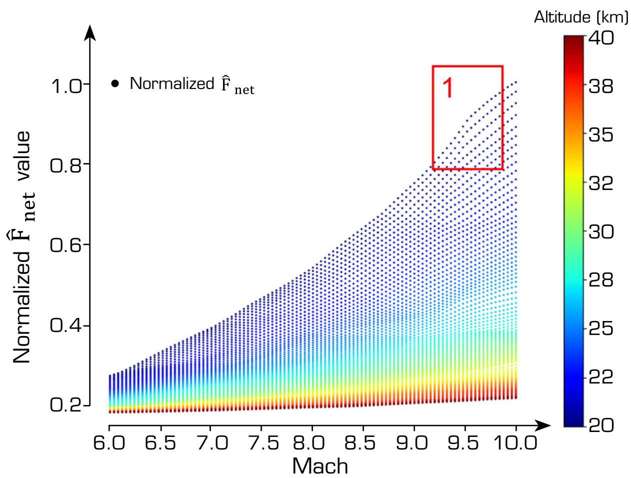
Figure 13. Ordered normalized differences between predictions and reference values for Model 1.



Source: Elaborated by the authors.

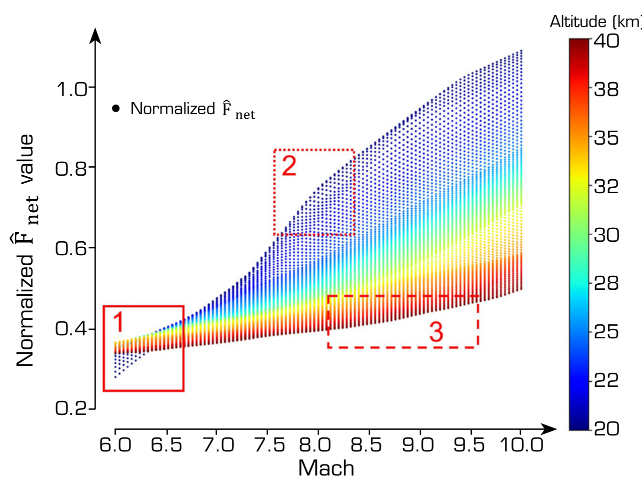
Figure 14. Ordered normalized differences between predictions and reference values for Model 2.

The analysis begins with Model 2 (Fig. 15), which was trained on Dataset 2 with uniform coverage across Mach-altitude operating-condition cells. $\hat{F}_{net}$ increases with Mach number and decreases with altitude over the envelope, yielding a smooth thrust map consistent with the baseline trends embedded in the analytical performance model under the imposed feasibility constraints. Because Dataset 2 provides uniform Mach-altitude coverage by construction, Model 2 is evaluated predominantly under interpolation conditions in the (M_0, Z) space. Specifically, the dense grid includes intermediate (M_0, Z) values between the discrete cells used during dataset construction, while remaining within the bounded Mach and altitude ranges spanned by Dataset 2.



Source: Elaborated by the authors.

Figure 15. Normalized $\hat{F}_{net}$ across a dense Mach-altitude grid (101 × 101 combinations) for Model 2. Region (1) highlights a minor local deviation at $Z = 20$ km for $M_0 > 9.5$.



Source: Elaborated by the authors.

Figure 16. Normalized $\hat{F}_{net}$ across a dense Mach-altitude grid (101 × 101 combinations) for Model 1. Highlighted regions (1-3) indicate physically inconsistent thrust trends.

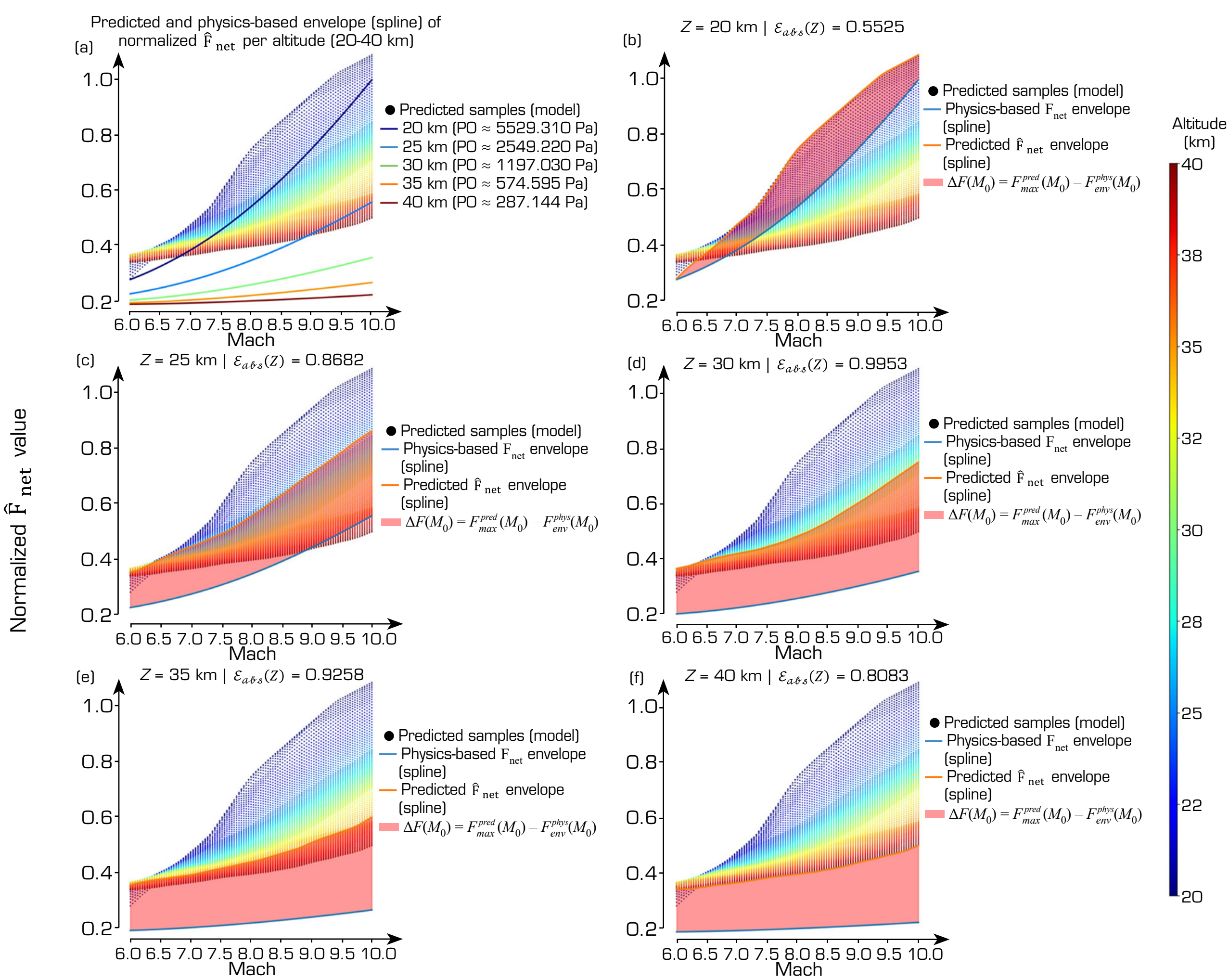
Although Model 2 largely adheres to physical expectations, a minor deviation is observed at high Mach numbers ($M_0 > 9.5$) and low altitude (20 km), where the $\hat{F}_{net}$ curve exhibits a slight flattening (Fig. 15, region 1). This localized deviation cannot be attributed to non-uniform Mach- altitude coverage, which is controlled in Dataset 2 by construction. It may be associated with representational limitations of the adopted NN architecture or with the non-uniform distribution of geometric configurations within cells, but this investigation is beyond the scope of this work.

In contrast, Model 1 (Fig. 16), trained on Dataset 1 with non-uniform Mach-altitude coverage, exhibits several trend departures. At Mach 6, the model predicts lower thrust at lower altitudes (Fig. 16, region 1), which contradicts the baseline trend described in the Methodology section for the analytical model under the imposed constraints. A second inconsistency appears along the $Z = 20$ km curve (Fig. 16, region 2), which increases with Mach number at lower Mach values but then exhibits pronounced flattening at higher Mach values, departing from the expected monotonic thrust trend at fixed altitude. Furthermore, for Mach numbers above approximately 8.8 (Fig. 16, region 3), the predicted thrust at high altitude (40 km) displays an anomalous increase in slope relative to adjacent regions, diverging from the more regular gradients observed for Model 2. These deviations are consistent with the fact that Model 1 is evaluated under both interpolation and extrapolation conditions in the Mach-altitude envelope, given the non-uniform Mach-altitude coverage of Dataset 1. In particular, the dense-grid evaluation includes altitudes (20 km and 40 km) that are outside the altitude interval used to generate Dataset 1, and it also probes Mach-altitude combinations corresponding to cells that are absent in Dataset 1.

Envelope-based physics-consistency analysis

To further quantify the qualitative trends observed in Figs. 15 and 16, an envelope-based physics-consistency analysis was performed following the methodology defined in the Methodology section. For each fixed altitude $Z \in \{20, 25, 30, 35, 40\}$ km, the DNN was queried along the Mach interval using the fixed reference geometry $\mathbf{x}_{\text{ref}}$. This yields the altitude-conditioned predicted response $F_{\text{ref}}^{\text{pred}}(M_0)$ defined in Eq. 12. For comparison with the physics-based reference, both $F_{\text{ref}}^{\text{pred}}(M_0)$ and $F_{\text{env}}^{\text{phys}}(M_0, Z)$ were represented as smooth spline curves over the Mach interval. The signed discrepancy between both envelopes was quantified by Eq. 13 and summarized through the absolute envelope error $\mathcal{E}_{abs}(Z)$ (Eq. 14).

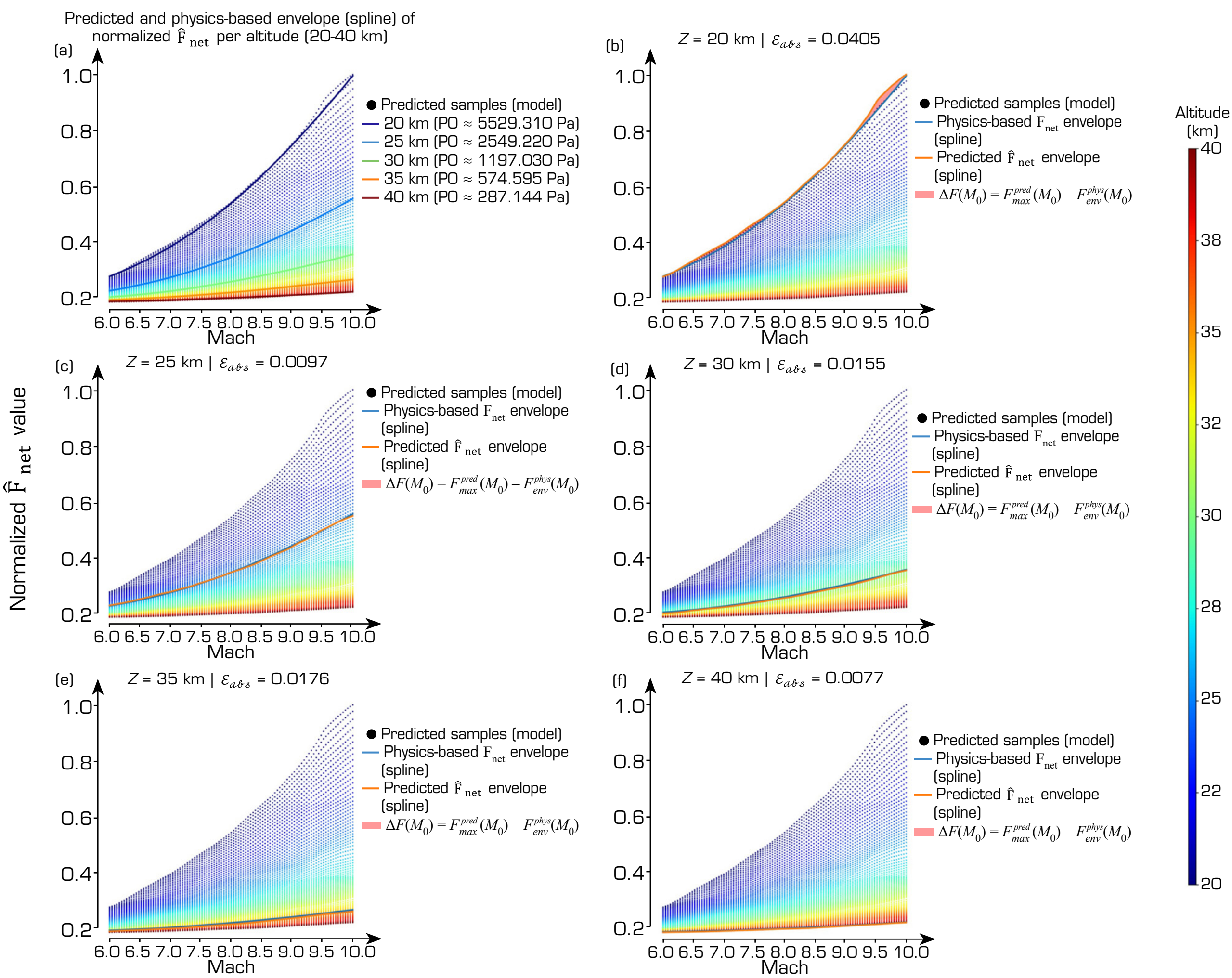
Figure 17 shows that Model 1 violates the physics-based envelope over a large portion of the Mach range at all altitudes considered. Large regions of positive $\Delta F(M_0)$ are observed, resulting in consistently high values of $\mathcal{E}_{abs}$. This behavior provides direct evidence that, despite competitive global error metrics, Model 1 fails to preserve physically admissible thrust trends when



Source: Elaborated by the authors.

Figure 17. Altitude-conditioned thrust envelopes for Model 1 compared with the physics-based envelope. The shaded regions represent $\Delta F(M_0)$ and $\mathcal{E}_{abs}(Z)$ summarizes the envelope discrepancy at each altitude.

evaluated across the full Mach-altitude envelope. In contrast, Fig. 18 indicates that Model 2 produces predicted envelopes that remain closely aligned with the physics-based reference across all altitudes. The resulting $\Delta F(M_0)$ values are small, and $\mathcal{E}_{abs}$ remains low, confirming that uniform Mach-altitude coverage during dataset construction substantially improves the physical consistency of DNN predictions.



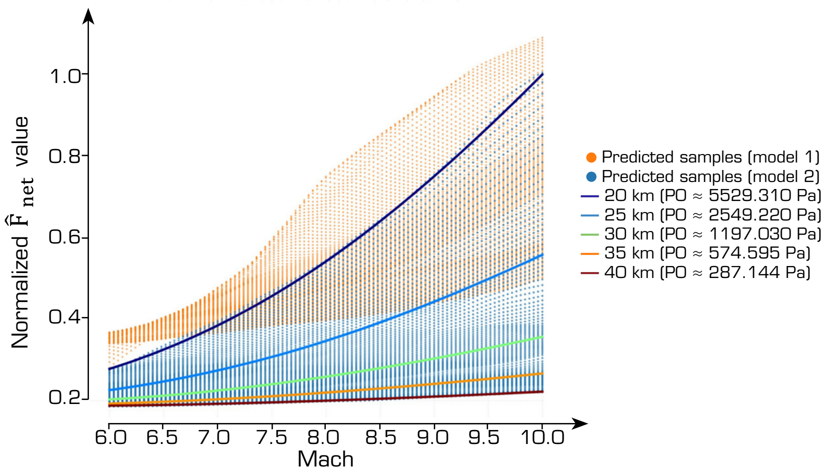
Source: Elaborated by the authors.

Figure 18. Same envelope-based analysis as in Fig. 17, now for Model 2. The predicted envelopes remain closely aligned with the physics-based reference across all altitudes.

Finally, Fig. 19 provides a consolidated visual comparison of the predicted F_{net} values from Model 1 and Model 2 against the corresponding physics-based envelope $F_{env}^{phys}(M_0)$ for each altitude. This overlaid representation clearly indicates the unphysical thrust trends exhibited by Model 1, manifested as envelope violations and departures from the expected monotonic increase of thrust with M_0 at fixed altitude, as previously quantified through $\Delta F(M_0)$ and $\mathcal{E}_{abs}$.

Discussion on non-uniform coverage

Model 1 and Model 2 share identical neural architecture, loss function, optimizer, and training protocol. The sole distinction between them lies in the distributions of the training data. Although conventional statistical metrics suggest similar predictive accuracy, the envelope-based analysis reveals substantial differences in physical consistency. As discussed in the Methodology



Source: Elaborated by the authors.

Figure 19. Normalized $\hat{F}_{net}$ values from Model 1 and Model 2 as a function of M_0. Scatter points show all predicted samples (color-coded by model), and solid curves denote the physics-based thrust envelope $F_{env}^{phys}(M_0, Z)$ for discrete altitudes $Z \in \{20, 25, 30, 35, 40\}$ km obtained from analytical models' splines.

section, Dataset 1 has several operating-condition cell gaps. These gaps correspond to non-uniform Mach–altitude coverage of the training distribution, constraining the regions in which DNNs can reliably learn baseline trend behavior across operating conditions.

The results demonstrate that the consequences of uneven sampling and non-uniform coverage extend beyond minor statistical deviations. Using identical network architectures, Model 1 achieved a lower RMSE yet produced thrust envelopes that violate first-principles expectations, whereas Model 2 preserved physically-consistent thrust trends despite being trained on fewer samples and showing a slightly greater statistical error. By holding the network architecture and training protocol fixed while varying only the distribution of Mach-altitude operating-condition cells, we demonstrated that data coverage is a primary determinant of physical consistency in DL-based aerodynamic modeling of scramjets.

These findings emphasize that, in physics-governed domains, training data do not merely populate an input space but encode admissible trend structure implied by the modeling assumptions and feasibility constraints used to generate the data. Ensuring adequate Mach-altitude coverage across all relevant operating-condition cells constitutes a methodological requirement for developing DNN models that are not only numerically accurate but also physically consistent and reliable across the operational envelope.

CONCLUSION

Juxtaposing the analytical performance model under the imposed feasibility constraints as the baseline reference, this study demonstrated that non-uniform Mach-altitude coverage across operating-condition cells can substantially degrade the physical consistency of DNN predictions for scramjet thrust across broad operational regimes. Although the present case study focuses on datasets generated through optimization-driven sampling, the observed failure modes arise from fundamental properties of the training distribution – namely coverage and sampling density – and are expected to occur whenever these properties are not adequately controlled. While statistical metrics may look promising, the absence of adequate Mach-altitude coverage undermines the reliability of learned mappings when evaluated across operating conditions. By comparing two datasets, the results showed that uneven sampling density and non-uniform Mach-altitude coverage may yield statistically competitive models while producing physically inconsistent thrust trends across operating conditions. Although Dataset 1 contained 2.4 times more training samples than Dataset 2, Model 1 failed to preserve physical consistency across the Mach-altitude envelope, whereas Model 2 maintained thrust trends aligned with the physics-based thrust envelope adopted in this work.

Evidence indicates that Model 1 tended to produce physically inconsistent aerodynamic predictions, such as reduced net thrust at lower altitudes or non-monotonic thrust trends contradicting the monotonic compressible-flow response embedded in the analytical performance model. These inconsistencies indicate that the model response was influenced by artifacts associated with uneven sampling density and non-uniform Mach-altitude coverage across operating-condition cells, leading to physically implausible outputs. Within the scope of this study, the results further indicate that statistical performance alone is insufficient to certify physical plausibility. Taken together, the statistical results and the envelope-based physics-consistency analyses support the formulated hypothesis.

Strengthening DL applications in physics-governed domains, such as hypersonic aerodynamics, requires addressing gaps in Mach-altitude coverage and sampling density throughout the Mach-altitude operating envelope. Beyond the specific optimization-driven sampling strategy examined here, any data-generation process that fails to ensure adequate representation of relevant operating regimes may induce similar failure modes. Accordingly, rigorous data engineering practices that explicitly control sampling coverage are essential for developing DNN surrogates that are not only statistically accurate but also physically reliable. Thus, well-characterized training distributions contribute to credibility in AI-assisted aerodynamic modeling workflows.

Upcoming studies could extend the investigation beyond the current dataset-construction strategy to examine potential deficiencies in coverage within the partitions generated by the S7FCV scheme. Although S7FCV provides a robust framework for model evaluation and selection, random partitioning may still induce fold-specific asymmetries in the representation of Mach-altitude operating-condition cells, thereby affecting performance estimates. Additionally, exploring alternative sampling and data-generation strategies is warranted, aiming at performance comparisons. Variations in network architecture – such as activation functions, layer depth, and neuron count – also merit investigation as potential avenues for improvement.

Strategies for modeling the nonlinear and complex behavior of scramjet systems may benefit from more intricate architectures, such as transformer-based/self-attention networks (Niu *et al.* 2021), Kolmogorov-Arnold networks (Park *et al.* 2025), or interpolating NNs (Liu *et al.* 2025). Moreover, AI methodologies should extend beyond compression ramp analysis to encompass broader aspects of hypersonic vehicle design, such as combustion chamber efficiency, internal geometry, and airframe integration. Ultimately, data engineering that ensures adequate coverage across operating-condition cells should serve as a cornerstone of responsible physics-informed AI deployment in aerospace research, ensuring that future models remain not only computationally effective but also grounded in the physical representativeness that underpins scientific reliability.

CONFLICT OF INTEREST

Nothing to declare.

AUTHORS CONTRIBUTIONS

Conceptualization: Paulino AC; **Methodology**: Paulino AC; **Software:** Paulino AC, Tanaka RY, and Araújo PPB; **Data Curation:** Paulino AC; **Formal Analysis:** Paulino AC and Passaro A; **Validation:** Paulino AC and Passaro A; **Visualization:** Paulino AC; **Writing – Original Draft Preparation:** Paulino AC; **Writing – Review and Editing:** Paulino AC, Passaro A, Araújo PPB, and Tanaka RY; **Supervision:** Passaro A: **Final approval:** Paulino AC.

DATA AVAILABILITY STATEMENT

The data will be available upon reasonable request.

FUNDING

Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
Finance code: 001.

Conselho Nacional de Desenvolvimento Científico e Tecnológico
Grant No: 306639/2025-4.

CAPES and Ministério da Defesa
Grant No: 88881.387753/2019-01.

DECLARATION OF USE OF ARTIFICIAL INTELLIGENCE TOOLS

Artificial intelligence was used for revising the final text.

ACKNOWLEDGEMENTS

"This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001."

REFERENCES

[NASA] National Aeronautics and Space Administration (1976) U.S. standard atmosphere. Washington: NASA. [accessed Mar 18 2021]. https://ntrs.nasa.gov/citations/19770009539

Ai S, Song J, Cai G (2021) A real-time fault diagnosis method for hypersonic air vehicle with sensor fault based on the auto temporal convolutional network. Aerospace Sci Technol 119:107-220. https://doi.org/10.1016/j.ast.2021.107220

Almaiah MA, Alrawashdeh R, Alkhdour T, Al-Ali R, Rjoub G, Aldahyani T (2024) Detecting DDoS attacks using machine learning algorithms and feature selection methods. Int J Data Netw Sci 8(4):2307-2318. https://doi.org/10.5267/j.ijdns.2024.6.001

Anderson JD (2003) Modern compressible flow: with historical perspective. Aeronautical and aerospace engineering series. New York: McGraw-Hill Education.

Anderson JD (2006) Hypersonic and high-temperature gas dynamics. 2nd ed. Reston: American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/4.861956

Araújo PP, Pereira MV, Marinho GS, Martos JF, Toro PG (2021) Optimization of scramjet inlet based on temperature and Mach number of supersonic combustion. Aerospace Sci Technol 116:106. https://doi.org/10.1016/j.ast.2021.106864

Araújo PPB, Tanaka RY, Silva CA, Passaro A, Toro PGP (2024) Multi-objective optimization of a hypersonic air breathing vehicle. Phys Fluids 36(2). https://doi.org/10.1063/5.0181366

Beachy A, Bae H, Boyd I, Grandhi R (2021) Emulator embedded neural networks for multi- fidelity conceptual design exploration of hypersonic vehicles. Paper presented 2021 Scitech 2021 Forum. American Institute of Aeronautics and Astronautics; Reston, USA. https://doi.org/10.2514/6.2021-1241

Bengio Y (2009) Learning deep architectures for AI. Found Trends Mach Learn 2(1):1-127. https://doi.org/10.1561/9781601982957

Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. J Mach Learn Res 13: 281-305.

Bertin JJ, Cummings RM (2003) Fifty years of hypersonics: where we've been, where we're going. Prog Aerosp Sci 39(6-7): 511-536. https://doi.org/10.1016/S0376-0421(03)00079-4

Brahmachary S, Fujio C, Aksay M, Ogawa H (2022) Design optimization and off-design performance analysis of axisymmetric scramjet intakes for ascent flight. Phys Fluids 34(3). https://doi.org/10.1063/5.0080272

Brar AS, Singh K (2024) A multi-objective stacked regression method for distance-based colour measuring device. Sci Rep 14(1):5530. https://doi.org/10.1038/s41598-024-54785-4

Brunton SL, Nathan Kutz J, Manohar K, Aravkin AY, Morgansen K, Klemisch J, Goebel N, Buttrick J, Poskin J, Blom-Schieber AW, *et al*. (2021) Data-driven aerospace engineering: reframing the industry with machine learning. AIAA J 59(8):1-26. https://doi.org/10.2514/1.J060131

Bu X, Hua C, Lv M, Wu Z (2023) Flight control of waverider vehicles with fragility-avoidance prescribed performance. IEEE Trans Aerosp Electron Syst. https://doi.org/10.1109/TAES.2023.3251314

Chai T, Draxler RR (2014) Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature. Geoscientific Model Dev 7(3). https://doi.org/10.5194/gmd-7-1247-2014

Chen W, Yang K, Yu Z, Shi Y, Chen CLP (2024) A survey on imbalanced learning: latest research, applications and future directions. Artif Intell Rev 57(6):137. https://doi.org/10.1007/s10462-024-10759-6

Choi B, Song J (2026) Cross-entropy-based adaptive concurrent importance sampling for probabilistic seismic loss analysis of spatially distributed multistate systems. ASCE-ASME J Risk Uncertain Eng Syst A Civ Eng 12(1). https://doi.org/10.1061/AJRUA6.RUENG-1651

Cui L, Liu A, Yv C, Quan T (2018) Learning algorithm for tracking hypersonic targets in near space. In: Lecture notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering. Cham: Springer. https://doi.org/10.1007/978-3-319-73447-7_24

De Lima Filho GM, Kuroswiski AR, Medeiros FLL, Voskuijl M, Monsuur H, Passaro A (2022) Optimization of unmanned air vehicle tactical formation in war games. IEEE Access 10:21.727-21.741. https://doi.org/10.1109/ACCESS.2022.3152768

Delalleau O, Bengio Y (2011) Shallow vs. deep sum-product networks. In: Weinberger JST, Zemel R, Bartlett P, Pereira F, Weinberger K, editors. Advances in neural information processing systems. Vol. 24. New York: Curran Associates, Inc. p. 666-674.

Doering J, Kizys R, Juan AA, Fitó À, Polat O (2019) Metaheuristics for rich portfolio optimisation and risk management: Current state and future trends. Eur J Oper Res 277(1):1-14. https://doi.org/10.1016/j.orp.2019.100121

Doğan B, Ölmez T (2015) A new metaheuristic for numerical function optimization: Vortex search algorithm. Inf Sci (N Y) 293:125-145. https://doi.org/10.1016/j.ins.2014.08.053

Dong Y, Jiang H, Liu Y, Yi Z (2024) Global wavelet-integrated residual frequency attention regularized network for hypersonic flight vehicle fault diagnosis with imbalanced data. Eng Appl Artif Intell 132:107:968. https://doi.org/10.1016/j.engappai.2024.107968

Drake JH, Kheiri A, Özcan E, Burke EK (2020) Recent advances in selection hyper-heuristics. Eur J Oper Res 2020;285(2): 405-428. https://doi.org/10.1016/j.ejor.2019.07.073

Du KL, Swamy MNS (2016) Search and optimization by metaheuristics techniques and algorithms inspired by nature. Cham: Birkhäuser. https://doi.org/10.1007/978-3-319-41192-7_12

Eyi S, Hanquist KM, Boyd ID (2019) Aerothermodynamic design optimization of hypersonic vehicles. J Thermophys Heat Transf 33(2):392-406. https://doi.org/10.2514/1.T5523

Fujio C, Ogawa H (2022) Deep-learning prediction and uncertainty quantification for scramjet intake flowfields. Aerospace Sci Technol 130:107. https://doi.org/10.1016/j.ast.2022.107931

Garud SS, Karimi IA, Kraft M (2017) Design of computer experiments: a review. Comput Chem Eng 106:71-95. https://doi.org/10.1016/j.compchemeng.2017.05.010

Geraci G, Menhorn F, Huan X, Safta C, Marzouk Y, Najm HN, Eldred MS (2019) Progress in scramjet design optimization under uncertainty using simulations of the HIFIRE direct connect rig. Paper presented 2019 AIAA Scitech 2019 Forum. AIAA; San Diego. USA. https://doi.org/10.2514/6.2019-0725

Gond A, Sengupta A (2025) Data-driven optimal design of premixed hydrogen-air combustion in a microcombustor. AI Therm Fluids 1. https://doi.org/10.1016/j.aitf.2025.100007

Goodfellow I, Bengio Y, Courville A (2016) Deep learning. Cambridge: MIT Press.

Han Z, Yu H, He Q (2025) Optimization of maritime target element resolution strategies for non-uniform sampling based on large language model fine-tuning. J Mar Sci Eng 13(10):1865. https://doi.org/10.3390/jmse13101865

Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, *et al.* (2020) Array programming with NumPy. Nature 585(7825):357-362. https://doi.org/10.1038/s41586-020-2649-2

Haykin SS (2009) Neural networks and learning machines. 3rd ed. Upper Saddle River: Pearson Education.

Heiser WH, Pratt DT, Daley D, Mehta U (1994) Hypersonic airbreathing propulsion. AIAA Education Series. Reston: AIAA. https://doi.org/10.2514/4.470356

Hosseini-Dinani H, Mokhtari AR, Shahrestani S, De Vivo B (2019) Sampling density in regional exploration and environmental geochemical studies: a review. Nat Resour Res 28(3):967-994. https://doi.org/10.1007/s11053-018-9431-2

Houssein EH, Saeed MK, Hu G, Al-Sayed MM (2024) Metaheuristics for solving global and engineering optimization problems: review, applications, open issues and challenges. Arch Comput Methods Eng 31(8):4485-4519. https://doi.org/10.1007/s11831-024-10168-6

Hu X, Chen X, Parks GT, Yao W (2016) Review of improved Monte Carlo methods in uncertainty-based design optimization for aerospace vehicles. Prog Aerosp Sci 86:20-27. https://doi.org/10.1016/j.paerosci.2016.07.004

Hunter JD (2007) Matplotlib: a 2D graphics environment. Comput Sci Eng 9(3):90-95. https://doi.org/10.1109/MCSE.2007.55

Idris A, Saad M, Rahman M, Hashim F, Kontis K (2019) Experimental validation of artificial neural network (ANN) model for scramjet inlet monitoring and control. International Journal of Recent Technology and Engineering 7(5):558-563. https://www.ijrte.org/wp-content/uploads/papers/v7i5s4/E11180275S419.pdf

Ispir AC, Saracoglu BH, Magin T, Coussement A (2023) A methodology for estimating hypersonic engine performance by coupling supersonic reactive flow simulations with machine learning techniques. Aerospace Sci Technol 140:108501. https://doi.org/10.1016/j.ast.2023.108501

Kennedy J, Eberhart R (1995) Particle swarm optimization. ICNN'95 – International Conference on Neural Networks. IEEE 4:1942-1948. https://doi.org/10.1109/ICNN.1995.488968

Kontolati K, Loukrezis D, Giovanis DG, Vandanapu L, Shields MD (2022) A survey of unsupervised learning methods for high-dimensional uncertainty quantification in black-box-type problems. J Comput Phys 464:111313. https://doi.org/10.1016/j.jcp.2022.111313

Korkegi RH (1975) Comparison of shock-induced two- and three-dimensional incipient turbulent separation. AIAA J 13(4):534-535. https://doi.org/10.2514/3.49750

Krishna A, Joseph VR, Ba S, Brenneman WA, Myers WR (2022) Robust experimental designs for model calibration. J Qual Technol 54(4):441-452. https://doi.org/10.1080/00224065.2021.1930618

Le Clainche S, Ferrer E, Gibson S, Cross E, Parente A, Vinuesa R (2023) Improving aircraft performance using machine learning: a review. Aerospace Sci Technol 138(8):108354. https://doi.org/10.1016/j.ast.2023.108354

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436-444. https://doi.org/10.1038/nature14539

Ling J, Kurzawski A, Templeton J (2016) Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. J Fluid Mech 807:155-166. https://doi.org/10.1017/jfm.2016.615

Liu Z, Wang Y, Vaidya S, Ruehle F, Halverson J, Soljačić M, Hou TY, Tegmark M (2025) KAN: Kolmogorov-Arnold Networks. Paper presented 2025 13th International Conference on Learning Representations. ICLR; Singapore, Malaysia. https://proceedings.iclr.cc/paper_files/paper/2025/file/afaed89642ea100935e39d39a4da602c-Paper-Conference.pdf

Long TW, Hanzevack EL (1995) Hierarchical competitive net architecture. In: Kenneth J Hunt, George R, Irwin KW, editors. Neural network engineering in dynamic control systems. London: Springer; pp 255-275. https://doi.org/10.1007/978-1-4471-3066-6_13

Loshchilov I, Hutter F (2019) Decoupled weight decay regularization. Paper presented 2019 7th International Conference on Learning Representations. ICLR; New Orleans, USA. https://doi.org/10.48550/arXiv.1711.05101

Lv J, Wang C, Kao Y (2023) Adaptive fixed-time quantized fault-tolerant attitude control for hypersonic reentry vehicle. Neurocomputing 520:386-399. https://doi.org/10.1016/j.neucom.2022.11.057

Mao Z, Lu L, Marxen O, Zaki T, Karniadakis G (2021) DeepM&Mnet for hypersonics: predicting the coupled flow and finite-rate chemistry behind a normal shock using neural-network approximation of operators. J Comput Phys 447. https://doi.org/10.1016/j.jcp.2021.110698

McCall T, Seyed Alavi K, Rana L, Chudoba B (2018) Artificial intelligent research assistant for aerospace design synthesis-solution logic. Paper presented 2018 22nd AIAA International Space Planes and Hypersonics Systems and Technologies Conference. AIAA; Reston, USA. https://doi.org/10.2514/6.2018-5387

McCulloch W, Pitts W (2021) A logical calculus of the ideas immanent in nervous activity. In: *Ideas that created the future*. Cambridge: MIT Press. p. 79-88. https://doi.org/10.7551/mitpress/12274.003.0011

McKinney W (2010) Data structures for statistical computing in python. Paper presented 2010 9th Python in Science Conference. SciPy; Austin, USA. https://doi.org/10.25080/Majora-92bf1922-00a

Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller M, Fidjeland AK, Ostrovski G, *et al.* (2015) Human-level control through deep reinforcement learning. Nature 518(7540):529-533. https://doi.org/10.1038/nature14236

Niu Z, Zhong G, Yu H (2021) A review on the attention mechanism of deep learning. Neuro computing 452:48-62. https://doi.org/10.1016/j.neucom.2021.03.091

O'Brien TF, Starkey RP, Lewis MJ (2001) Quasi-one-dimensional high-speed engine model with finite-rate chemistry. J Propuls Power 17(6):1366-1374. https://doi.org/10.2514/2.5889

Ogawa H, Boyce RR (2012a) Multi-objective design optimization of fuel injection for mixing enhancement in scramjets by using surrogate-assisted evolutionary algorithms. Paper presented 2012 18th AIAA/3AF International Space Planes and Hypersonic Systems and Technologies Conference. AIAA; Tours, France. https://doi.org/10.2514/6.2012-5815

Ogawa H, Boyce RR (2012b) Physical insight into scramjet inlet behavior via multi-objective design optimization. AIAA J 50(8):1773-1783. https://doi.org/10.2514/1.J051644

Oliveira PM, Solteiro Pires EJ, Boaventura-Cunha J, Pinho TM (2020) Review of nature and biologically inspired metaheuristics for greenhouse environment control. Trans Inst Meas Control 2020;42(12):2338-2358. https://doi.org/10.1177/0142331220909010

Ozbenli E, Vedula P, Vogiatzis K, Josyula E (2020) Numerical solution of hypersonic flows via artificial neural networks. Paper presented 2020 AIAA Scitech Forum. AIAA: Orlando, USA. https://doi.org/10.2514/6.2020-1233

Park C, Saha S, Guo J, Zhang H, Xie X, Bessa MA, Qian D, Chen W, Wanger GJ, Cao J, *et al.* (2025) Unifying machine learning and interpolation theory via interpolating neural networks. Nat Commun 16(1):8753. https://doi.org/10.1038/s41467-025-63790-8

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, *et al.* (2019) PyTorch: an imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, Alché-Buc F, Fox E, Garnett R, editors. Advances in neural information processing systems. Vol. 32. New York: Curran Associates, Inc. p. 8026-8037.

Paulino ÂdC, Guimarães LNF, Shiguemori EH (2019) Hybrid adaptive computational intelligence-based multisensor data fusion applied to real-time UAV autonomous navigation. Intelig Artif 22(63):162-195. https://doi.org/10.4114/intartif.vol22iss63pp162-195

Paulino ÂdC, Guimaraes LNF, Shiguemori EH (2020) Assessment of noise impact on hybrid adaptive computational intelligence multisensor data fusion applied to real-time UAV autonomous navigation. IEEE Lat Am Trans 18(02):295-302. https://doi.org/10.1109/TLA.2020.9085283

Paulino ÂdC, Passaro A (2023) Investigação cientométrica sobre aplicações de redes neurais em hipersônica. Paper presented 2023 XXV Simpósio de Aplicações Operacionais em Áreas de Defesa. Instituto Tecnológico de Aeronáutica; São José dos Campos, Brasil. https://www.sige.ita.br/edicoes-anteriores/2023/st/235039_1.pdf?x99667

Paulino ÂdC, Passaro A (2024) Estudo do desempenho aerodinâmico de scramjets usando redes neurais profundas. Paper presented 2024 XXVI Simpósio de Aplicações Operacionais em Áreas de Defesa. Instituto Tecnológico de Aeronáutica; São José dos Campos, Brasil. https://www.sige.ita.br/edicoes-anteriores/2024/st/243329_1.pdf

Pedersen CP, Prestel A, Teilum K (2021) Software for reconstruction of nonuniformly sampled NMR data. Magn Reson Chem 59(3):315-323. https://doi.org/10.1002/mrc.5060

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, *et al.* (2011) Scikit-learn: machine learning in python. Journal of Machine Learning Research 12:2825-2830

Ren J, Wu X, Liu Y, Ni F, Bo Y, Jiang C (2023) Long-term trajectory prediction of hypersonic glide vehicle based on physics-informed transformer. IEEE Trans Aerosp Electron Syst 59(6):9551-9561. https://doi.org/10.1109/TAES.2023.3322977

Russo G, Voto C, Savino R (2021) S4 – A demonstrator of HYPLANE, a single stage suborbital spaceplane and a hypersonic business jet. Acta Astronaut 183:244-254. https://doi.org/10.1016/j.actaastro.2021.03.025

Saba WB (2017) Framework de otimização baseada em metaheurísticas (masters thesis). São José dos Campos: Instituto Tecnológico de Aeronáutica. In Portuguese. http://www.bdita.bibl.ita.br/tesesdigitais/lista_resumo.php?num_tese=73428

Santana N, Santana N, Caldeira A, Passaro A, Guimaraes L (2022) Optimization of a microreactor core's dimensions using metaheuristic methods. Braz J Radiat Sci 10(3A Suppl.). https://doi.org/10.15392/2319-0612.2022.1903

Scarlatella G, Guadagnini J, Paniccià L, De Zaiacomo G, Tajmar M, Bach C, Pastrone D (2024) Assessment of mission capabilities of a reusable heavy-lift launch vehicle concept with aerospike engine. Paper presented 2024 AIAA SCITECH Forum. American Institute of Aeronautics and Astronautics; Reston, USA. https://doi.org/10.2514/6.2024-2122

Schmidhuber J (2015) Deep learning in neural networks: an overview. Neural Netw 61:85-117. https://doi.org/10.1016/j.neunet.2014.09.003

Shi J, Wang J, Su L, Ma Z, Chen H (2022) A neural network warm-started indirect trajectory optimization method. Aerospace 9(8):435. https://doi.org/10.3390/aerospace9080435

Shou Y, Yan T, Xu B, Sun F (2023) Integrated guidance and control of hypersonic flight vehicle with coordinated mission requirement and input constraint. International Journal of Robust and Nonlinear Control 33(7). https://doi.org/10.1002/rnc.6607

Silva B, Paulino ÂdC, Rêgo IdS, de Oliveira ÉJ, Passaro A (2021) Optimal preliminary design of hypersonic waverider using multiple metaheuristics. Paper presented at 72th International Astronautical Congress. International Astronautical Federation; Dubai, United Arab Emirates. https://iafastro.directory/iac/paper/id/63673/summary/

Smart MK (1999) Optimization of two-dimensional scramjet inlets. J Aircr 36(2):430. https://doi.org/10.2514/2.2448

Soares LHSM, Araújo PPB, Facchin RBC, Carneiro R, Toro PGP, Passaro A (2022) An optimization approach for conceptual and preliminary design of scramjet engines. Paper presented 2022 HiSST: 2nd International Conference on High-Speed Vehicle Science Technology. Council of European Aerospace Societies (CEAS); Brussels, Belgium. https://aerospacerepository.org/wp-content/uploads/2023/10/HiSST-2022-245.pdf

Sörensen K (2015) Metaheuristics – The metaphor exposed. Int Trans Oper Res 22(1):3-18. https://doi.org/10.1111/itor.12001

Speier R, Nacouzi G, Lee C, Moore R (2017) Hypersonic missile nonproliferation: hin- dering the spread of a new class of weapons. Santa Monica: RAND Corporation. https://doi.org/10.7249/RR2137

Staerk C, Klinkhammer H, Wistuba T, Maj C, Mayr A (2024) Generalizability of polygenic prediction models: how is the R2 defined on test data? BMC Med Genomics 17(1):132. https://doi.org/10.1186/s12920-024-01905-8

Sziroczak D, Smith H (2016) A review of design issues specific to hypersonic flight vehicles. Prog Aerosp Sci 84:1-28. https://doi.org/10.1016/j.paerosci.2016.04.001

Takashima N, Lewis MJ (1999) Optimization of wave rider-based hypersonic cruise vehicles with off-design considerations. J Aircr 36(1):235-245. https://doi.org/10.2514/2.2430

Talbi EG (2009) Metaheuristics: from design to implementation. Hoboken: John Wiley & Sons. https://doi.org/10.1002/9780470496916

Thuerey N, Weißenow K, Prantl L, Hu X (2020) Deep learning methods for Reynolds- averaged navier-stokes simulations of airfoil flows. AIAA J 58(1):25-36. https://doi.org/10.2514/1.J058291

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, editors. *Advances in neural information processing systems*. Vol. 30. Red Hook: Curran Associates, Inc. p. 5998-6008.

Wang J, Wu Y, Liu M, Yang M, Liang H (2022) A real-time trajectory optimization method for hypersonic vehicles based on a deep neural network. Aerospace 9(4). https://doi.org/10.3390/aerospace9040188

Wang W, Ma J (2024) A review: applications of machine learning and deep learning in aerospace engineering and aero-engine engineering. Adv Eng Innov 6(1):54-72. https://doi.org/10.54254/2977-3903/6/2024060

Wang X, Tang K, Zhai J, Wan X, Yang C (2024) Deep adaptive sampling for surrogate modeling without labeled data. J Sci Comput 101(3):77. https://doi.org/10.1007/s10915-024-02711-1

Yuchi HS, Joseph RV, Wu CFJ (2023) Design and analysis of multifidelity finite element simulations. J Mech Des 145(6). https://doi.org/10.1115/1.4056874

Zhang D, Yang H, Gao J, Li X (2024) Imbalanced flight test sensor temporal data anomaly detection. IEEE Trans Aerosp Electron Syst (1):1-11. https://ieeexplore.ieee.org/document/10707114